



# **GlassFish v3 Application Server Application Deployment Guide**

Technology Preview 2



Sun Microsystems, Inc.  
4150 Network Circle  
Santa Clara, CA 95054  
U.S.A.

Part No: 820-4502-06  
May 2008

Copyright 2008 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more U.S. patents or pending patent applications in the U.S. and in other countries.

U.S. Government Rights – Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

This distribution may include materials developed by third parties.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, the Solaris logo, the Java Coffee Cup logo, docs.sun.com, Java, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

Products covered by and information contained in this publication are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical or biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

---

Copyright 2008 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. Tous droits réservés.

Sun Microsystems, Inc. détient les droits de propriété intellectuelle relatifs à la technologie incorporée dans le produit qui est décrit dans ce document. En particulier, et ce sans limitation, ces droits de propriété intellectuelle peuvent inclure un ou plusieurs brevets américains ou des applications de brevet en attente aux États-Unis et dans d'autres pays.

Cette distribution peut comprendre des composants développés par des tierces personnes.

Certains composants de ce produit peuvent être dérivées du logiciel Berkeley BSD, licenciés par l'Université de Californie. UNIX est une marque déposée aux États-Unis et dans d'autres pays; elle est licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, le logo Solaris, le logo Java Coffee Cup, docs.sun.com, Java et Solaris sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux États-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux États-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui, en outre, se conforment aux licences écrites de Sun.

Les produits qui font l'objet de cette publication et les informations qu'il contient sont régis par la législation américaine en matière de contrôle des exportations et peuvent être soumis au droit d'autres pays dans le domaine des exportations et importations. Les utilisations finales, ou utilisateurs finaux, pour des armes nucléaires, des missiles, des armes chimiques ou biologiques ou pour le nucléaire maritime, directement ou indirectement, sont strictement interdites. Les exportations ou réexportations vers des pays sous embargo des États-Unis, ou vers des entités figurant sur les listes d'exclusion d'exportation américaines, y compris, mais de manière non exclusive, la liste de personnes qui font objet d'un ordre de ne pas participer, d'une façon directe ou indirecte, aux exportations des produits ou des services qui sont régis par la législation américaine en matière de contrôle des exportations et la liste de ressortissants spécifiquement désignés, sont rigoureusement interdites.

LA DOCUMENTATION EST FOURNIE "EN L'ETAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFACON.

# Contents

---

<b>Preface</b> .....	9
<b>1 Assembling and Deploying Web Applications</b> .....	15
Assembling Web Modules .....	15
About Web Modules .....	15
Java EE Standard Annotation .....	16
Java EE Standard Descriptor .....	16
Application Server Descriptor .....	17
Naming Standards .....	17
Directory Structure .....	17
The NetBeans IDE .....	17
Deploying Web Modules .....	18
Deployment Errors .....	18
The Deployment Life Cycle .....	18
Deployment for Development .....	19
Tools for Deployment .....	21
Deploying a WAR Module .....	22
<b>A Deployment Descriptor Files</b> .....	23
Sun Java System Application Server Descriptors .....	23
The sun-web.xml File .....	24
The sun-ejb-jar.xml File .....	28
A .....	32
activation-config .....	32
activation-config-property .....	33
activation-config-property-name .....	33
activation-config-property-value .....	34

as-context .....	34
auth-method .....	34
B .....	35
bean-cache .....	35
bean-pool .....	36
C .....	37
cache .....	37
cache-helper .....	39
cache-helper-ref .....	39
cache-idle-timeout-in-seconds .....	40
cache-mapping .....	40
call-property .....	41
caller-propagation .....	41
checkpoint-at-end-of-method .....	42
checkpointed-methods .....	42
class-loader .....	42
cmp .....	44
cmp-resource .....	44
cmt-timeout-in-seconds .....	45
commit-option .....	45
confidentiality .....	46
constraint-field .....	46
constraint-field-value .....	47
context-root .....	48
cookie-properties .....	48
create-tables-at-deploy .....	49
D .....	50
database-vendor-name .....	50
debugging-enabled .....	50
default-helper .....	50
default-resource-principal .....	51
description .....	52
dispatcher .....	52
drop-tables-at-undeploy .....	52
E .....	53
ejb .....	53

---

ejb-name .....	56
ejb-ref .....	56
ejb-ref-name .....	57
endpoint-address-uri .....	57
enterprise-beans .....	58
establish-trust-in-client .....	59
establish-trust-in-target .....	59
F .....	60
finder .....	60
flush-at-end-of-method .....	60
G .....	61
gen-classes .....	61
group-name .....	61
H .....	62
http-method .....	62
I .....	62
idempotent-url-pattern .....	62
integrity .....	63
ior-security-config .....	63
is-cache-overflow-allowed .....	63
is-one-one-cmp .....	64
is-read-only-bean .....	64
J .....	64
java-method .....	64
jms-durable-subscription-name .....	65
jms-max-messages-load .....	65
jndi-name .....	65
jsp-config .....	66
K .....	69
key-field .....	69
L .....	70
local-home-impl .....	70
local-impl .....	70
locale-charset-info .....	71
locale-charset-map .....	71
localpart .....	72

---

login-config .....	73
M .....	73
manager-properties .....	73
mapping-properties .....	75
max-cache-size .....	75
max-pool-size .....	76
max-wait-time-in-millis .....	76
mdb-connection-factory .....	76
mdb-resource-adapter .....	77
message .....	77
message-destination .....	78
message-destination-name .....	78
message-destination-ref .....	79
message-destination-ref-name .....	79
message-security .....	79
message-security-binding .....	80
method .....	81
method-intf .....	81
method-name .....	82
method-param .....	82
method-params .....	82
N .....	83
name .....	83
namespaceURI .....	83
O .....	84
one-one-finders .....	84
operation-name .....	84
P .....	84
parameter-encoding .....	84
pass-by-reference .....	85
password .....	86
pm-descriptors .....	87
pool-idle-timeout-in-seconds .....	87
port-component-name .....	87
port-info .....	88
prefetch-disabled .....	88

---

principal .....	89
principal-name .....	89
property (with attributes) .....	90
property (with subelements) .....	91
Q .....	91
query-filter .....	91
query-method .....	92
query-ordering .....	92
query-params .....	92
query-variables .....	93
R .....	93
realm .....	93
refresh-field .....	93
refresh-period-in-seconds .....	94
removal-timeout-in-seconds .....	94
remote-home-impl .....	95
remote-impl .....	95
request-protection .....	95
required .....	96
res-ref-name .....	96
resize-quantity .....	97
resource-adapter-mid .....	97
resource-env-ref .....	98
resource-env-ref-name .....	98
resource-ref .....	99
response-protection .....	99
role-name .....	100
S .....	100
sas-context .....	100
schema-generator-properties .....	101
security-role-mapping .....	102
service-endpoint-interface .....	103
service-impl-class .....	103
service-qname .....	104
service-ref .....	104
service-ref-name .....	105

servlet .....	105
servlet-impl-class .....	106
servlet-name .....	106
session-config .....	106
session-manager .....	107
session-properties .....	107
steady-pool-size .....	108
store-properties .....	108
stub-property .....	110
sun-ejb-jar .....	111
sun-web-app .....	111
T .....	115
tie-class .....	115
timeout .....	115
transport-config .....	116
transport-guarantee .....	117
U .....	117
unique-id .....	117
url-pattern .....	117
use-thread-pool-id .....	118
V .....	118
value .....	118
victim-selection-policy .....	118
W .....	119
webservice-description .....	119
webservice-description-name .....	120
webservice-endpoint .....	120
wsdl-override .....	121
wsdl-port .....	121
wsdl-publish-location .....	122
<b>Index</b> .....	<b>123</b>



# Preface

---

This *Application Deployment Guide* describes deployment of applications and application components to the Application Server, and includes information about deployment descriptors.

This preface contains information about and conventions for the entire GlassFish™ Application Server documentation set.

## Application Server Documentation Set

The Application Server documentation set describes deployment planning and system installation. The Uniform Resource Locator (URL) for Application Server documentation is <http://docs.sun.com/coll/1343.7>. For an introduction to Application Server, refer to the books in the order in which they are listed in the following table.

TABLE P-1 Books in the Application Server Documentation Set

Book Title	Description
<i>Release Notes</i>	Provides late-breaking information about the software and the documentation. Includes a comprehensive, table-based summary of the supported hardware, operating system, Java™ Development Kit (JDK™), and database drivers.
<i>Quick Start Guide</i>	Explains how to get started with the Application Server product.
<i>Installation Guide</i>	Explains how to install the software and its components.
<i>Application Deployment Guide</i>	Explains how to assemble and deploy applications to the Application Server and provides information about deployment descriptors.
<i>Developer's Guide</i>	Explains how to create and implement Java Platform, Enterprise Edition (Java EE platform) applications that are intended to run on the Application Server. These applications follow the open Java standards model for Java EE components and APIs. This guide provides information about developer tools, security, debugging, and creating lifecycle modules.

TABLE P-1 Books in the Application Server Documentation Set (Continued)

Book Title	Description
<i>Java EE 5 Tutorial</i>	Explains how to use Java EE 5 platform technologies and APIs to develop Java EE applications.
<i>Java WSIT Tutorial</i>	Explains how to develop web applications by using the Web Service Interoperability Technologies (WSIT). The tutorial focuses on developing web service endpoints and clients that can interoperate with Windows Communication Foundation (WCF) endpoints and clients.
<i>Administration Guide</i>	Explains how to configure and manage Application Server subsystems and components from the command line by using the <code>asadmin(1M)</code> utility. Instructions for performing these tasks from the Admin Console are provided in the Admin Console online help.
<i>RESTful Web Services Developer's Guide</i>	Explains how to develop Representational State Transfer (RESTful) web services for Application Server.
<i>Getting Started With JRuby on Rails for the GlassFish Application Server</i>	Explains how to develop Ruby on Rails applications for deployment to Application Server.
<i>Getting Started With Project jMaki for the GlassFish Application Server</i>	Explains how to use the jMaki framework to develop Ajax-enabled web applications that are centered on JavaScript™ technology for deployment to Application Server.
<i>Reference Manual</i>	Provides reference information in man page format for Application Server administration commands, utility commands, and related concepts.

## Related Documentation

A Javadoc™ tool reference for packages that are provided with the Application Server is located at <http://glassfish.dev.java.net/nonav/javaee5/api/index.html>. Additionally, the following resources might be useful:

- The Java EE 5 Specifications (<http://java.sun.com/javaee/5/javatech.html>)
- The Java EE Blueprints (<http://java.sun.com/reference/blueprints/index.html>)

For information about creating enterprise applications in the NetBeans™ Integrated Development Environment (IDE), see <http://www.netbeans.org/kb/60/index.html>.

For information about the Java DB database for use with the Application Server, see <http://developers.sun.com/javadb/>.

The GlassFish Samples project is a collection of sample applications that demonstrate a broad range of Java EE technologies. The GlassFish Samples are bundled with the Java EE Software Development Kit (SDK), and are also available from the GlassFish Samples project page at <https://glassfish-samples.dev.java.net/>.

## Default Paths and File Names

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

TABLE P-2 Default Paths and File Names

Placeholder	Description	Default Value
<i>as-install</i>	Represents the base installation directory for Application Server.	Installations on the Solaris™ operating system and Linux operating system: <i>user's-home-directory/glassfish-v3tp2/glassfish</i> Windows, all installations: <i>SystemDrive:\Program Files\glassfish-v3tp2\glassfish</i>
<i>domain-root-dir</i>	Represents the directory containing all domains.	<i>as-install/domains/</i>
<i>domain-dir</i>	Represents the directory for a domain. In configuration files, you might see <i>domain-dir</i> represented as follows: <code>\${com.sun.aas.instanceRoot}</code>	<i>domain-root-dir/domain-name</i>
<i>instance-dir</i>	Represents the directory for a server instance.	<i>domain-dir/instance-name</i>

## Typographic Conventions

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

TABLE P-3 Typographic Conventions

Typeface	Meaning	Example
<b>AaBbCc123</b>	The names of commands, files, and directories, and onscreen computer output	Edit your <code>.login</code> file. Use <code>ls -a</code> to list all files. <code>machine_name% you have mail.</code>
<b>AaBbCc123</b>	What you type, contrasted with onscreen computer output	<code>machine_name% su</code> Password:
<i>AaBbCc123</i>	A placeholder to be replaced with a real name or value	The command to remove a file is <code>rm filename</code> .

TABLE P-3 Typographic Conventions (Continued)

Typeface	Meaning	Example
<i>AaBbCc123</i>	Book titles, new terms, and terms to be emphasized (note that some emphasized items appear bold online)	Read Chapter 6 in the <i>User's Guide</i> . A <i>cache</i> is a copy that is stored locally. Do <i>not</i> save the file.

## Symbol Conventions

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

TABLE P-4 Symbol Conventions

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

## Documentation, Support, and Training

The Sun web site provides information about the following additional resources:

- Documentation (<http://www.sun.com/documentation/>)
- Support (<http://www.sun.com/support/>)
- Training (<http://www.sun.com/training/>)

## Searching Sun Product Documentation

Besides searching Sun product documentation from the docs.sun.com<sup>SM</sup> web site, you can use a search engine by typing the following syntax in the search field:

```
search-term site:docs.sun.com
```

For example, to search for “broker,” type the following:

```
broker site:docs.sun.com
```

To include other Sun web sites in your search (for example, [java.sun.com](http://java.sun.com), [www.sun.com](http://www.sun.com), and [developers.sun.com](http://developers.sun.com)), use sun.com in place of docs.sun.com in the search field.

## Third-Party Web Site References

Third-party URLs are referenced in this document and provide additional, related information.

---

**Note** – Sun is not responsible for the availability of third-party web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused or alleged to be caused by or in connection with use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

---

## Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. To share your comments, go to <http://docs.sun.com> and click Send Comments. In the online form, provide the full document title and part number. The part number is a 7-digit or 9-digit number that can be found on the book's title page or in the document's URL. For example, the part number of this book is 820-4502.



# Assembling and Deploying Web Applications

---

This chapter describes Sun Java™ System Application Server web modules and how these modules are assembled. This chapter also describes tools for assembly and deployment.

The Application Server web modules include Java Platform, Enterprise Edition (Java EE platform) standard features and Application Server specific features. Only Application Server specific features are described in detail in this chapter.

The following topics are presented in this chapter:

- “Assembling Web Modules” on page 15
- “Deploying Web Modules” on page 18

## Assembling Web Modules

Web application assembly (also known as packaging) is the process of combining discrete components of a web application into a single unit that can be deployed to a Java-EE-compliant application server. Assembly is described in the following topics:

- “About Web Modules” on page 15
- “Java EE Standard Annotation” on page 16
- “Java EE Standard Descriptor” on page 16
- “Application Server Descriptor” on page 17
- “Naming Standards” on page 17
- “Directory Structure” on page 17
- “The NetBeans IDE” on page 17

## About Web Modules

A web application or web module is a collection of servlets, HTML pages, classes, and other resources that can be bundled and deployed to several Java EE application servers. A WAR file can consist of the following items: servlets, JavaServer Pages™ (JSP™) files, JSP tag libraries,

utility classes, static pages, client-side applets, beans, bean classes, and annotations or deployment descriptors. In a web application, one descriptor (`web.xml`) is Java EE standard, the other (`sun-web.xml`) is optional and Application Server specific. Annotations can be used instead of Java EE standard descriptors.

Package definitions must be used in the source code of web modules so the class loader can properly locate the classes after the modules have been deployed.

Because the information in a deployment descriptor is declarative, it can be changed without requiring modifications to source code. At run time, the Java EE server reads this information and acts accordingly.

## Java EE Standard Annotation

The Application Server supports web modules annotated according to the following specifications:

- JSR 250 Common Annotation Specification (<http://www.jcp.org/en/jsr/detail?id=250>)
- EJB 3.0 Specification, which includes the Java Persistence API (<http://www.jcp.org/en/jsr/detail?id=220>)

The following annotation and deployment descriptor combinations are supported:

- Web modules can be packaged with full Java EE 5.0 compliant standard and runtime deployment descriptors. If the standard deployment descriptors have specified the attribute `metadata-complete`, annotations in the application or module are ignored.
- Web modules can be fully annotated with metadata defined by the listed specifications. Annotation eliminates the need for Java EE standard deployment descriptors. In most cases, the Application Server deployment descriptors are also optional.
- Web modules can be partially annotated with some deployment information in standard deployment descriptors. In case of conflicts, deployment descriptor values supersede the annotated metadata, but a warning message is logged.

## Java EE Standard Descriptor

Java EE standard deployment descriptors are described in the Java EE specification, v5. You can find the specification at <http://java.sun.com/products/>. Information about the XML schemas that define Java EE standard deployment descriptors is available at <http://java.sun.com/xml/ns/javaee/>. The Java EE standard deployment descriptor for web modules is `web.xml`.



## Application Server Descriptor

The Application Server uses an additional, optional deployment descriptor, `sun-web.xml`, for configuring web module features specific to the Application Server.

For a complete descriptions of `sun-web.xml`, see [Appendix A, “Deployment Descriptor Files.”](#)

## Naming Standards

Names of web applications must be unique within an Application Server domain.

If you do not explicitly specify a name, the default name is the first portion of the file name (without the `.war` extension). This is the case when you use the Admin Console or the `asadmin` command. See [“Tools for Deployment” on page 21.](#)

You can specify a name in one of these ways:

- If deploying using the Admin Console, specify the name in the Application Name field.
- If deploying using the `asadmin deploy` command, the default name of the web module is the prefix of the WAR file that you are deploying. For example, for the `hello.war` file, the Web application name is `hello`. To override the default name, specify the `--name` option.

Make sure your package and file names do not contain spaces or characters that are illegal for your operating system.

Using a Java package-like naming scheme is recommended for module filenames. The use of this package-like naming scheme ensures that name collisions do not occur. The benefits of this naming practice apply not only to the Application Server, but to other Java EE application servers as well.

## Directory Structure

Web module directory structure follows the structure outlined in the Java EE specification.

## The NetBeans IDE

You can use the NetBeans™ Integrated Development Environment (IDE) to assemble web modules. The GlassFish edition of the Application Server is bundled with the NetBeans 5.5 IDE. For more information about using the NetBeans IDE, see <http://www.netbeans.org>.

# Deploying Web Modules

This section describes the different ways to deploy web modules to the Application Server. It covers the following topics:

- [“Deployment Errors” on page 18](#)
- [“The Deployment Life Cycle” on page 18](#)
- [“Deployment for Development” on page 19](#)
- [“Tools for Deployment” on page 21](#)
- [“Deploying a WAR Module” on page 22](#)

## Deployment Errors

If an error occurs during deployment, the web module is not deployed. This prevents a partial deployment that could leave the server in an inconsistent state.

In addition, certain warning conditions allow a web application to be deployed but return a warning message to the deployment client.

## The Deployment Life Cycle

After installing the Application Server and starting a domain, you can deploy (install) web modules. During deployment and as the application is changed, a web module can go through the following stages:

### 1. Initial Deployment

Before deploying a web module, start the domain.

Deploy (install) a web module. Because web modules are packaged in archive files, specify the archive file name during deployment.

You can also deploy using the `asadmin deploy` command or the Admin Console.

Deployment is *dynamic*: you don't need to restart the server instance after deploying for web applications to be available. If you do restart, all deployed web modules are still deployed and available.

### 2. Enabling or Disabling

By default, a web module is enabled, which means that it is runnable and can be accessed by clients if it has been deployed to an accessible server instance or cluster. To prevent access, disable the web module. A disabled web module is not uninstalled from the domain and can be easily enabled after deployment. For more information, see [“Disabling a Deployed Application or Module” on page 19](#).

### 3. Redeployment

To replace a deployed web module, redeploy it. Redeploying automatically undeploys the previously deployed web module and replaces it with the new one.

#### 4. Undeployment

To uninstall an application or module, undeploy it.

## Deployment for Development

This section covers the following topics related to deployment for development:

- “Dynamic Deployment” on page 19
- “Disabling a Deployed Application or Module” on page 19
- “Automatic Deployment” on page 20

---

**Note** – You can overwrite a previously deployed web application by using the `--force` option of `asadmin deploy` or by selecting the Redeploy button in the Admin Console. However, you must remove a preconfigured resource before you can update it.

---

### Dynamic Deployment

You can deploy, redeploy, and undeploy a web module without restarting the server instances. This is called dynamic deployment. Although primarily for developers, dynamic deployment can be used in operational environments to bring new web modules online without requiring a server restart.

Whenever a redeployment is done, the sessions at that transit time become invalid. The client must restart the session.

### Disabling a Deployed Application or Module

You can disable a deployed web module without removing it from the server. Disabling a web application makes it inaccessible to clients.

To disable a web module using the `asadmin disable` command, see the *GlassFish v3 Application Server Reference Manual*.

#### ▼ To Disable a Web Module in the Admin Console

- 1 Open the Applications component.
- 2 Go to the Web Applications page.
- 3 Click on the box to the left of the name of each web module you wish to disable.
- 4 Click on the Disable button.

**See Also** For details, click the Help button in the Admin Console.

## Automatic Deployment

Automatic deployment, also called *autodeployment*, involves copying a module file (WAR) into a special directory, where it is automatically deployed by the Application Server. To undeploy an automatically deployed web module, simply remove its file from the special autodeployment directory. This is useful in a development environment, because it allows new code to be tested quickly. This feature is available only on the default server instance.

Autodeployment is enabled by default.

### ▼ To Enable and Configure or to Disable Autodeployment

#### 1 To access autodeployment features:

- In the developer profile:
  - a. Select the Application Server component.
  - b. Select the Advanced tab.
  - c. Select the Applications Configuration tab.
- In the cluster profile:
  - a. Select the Stand-Alone Instances component.
  - b. Select the instance named `server` in the table.  
This is the Admin Server.
  - c. Select the Advanced tab.

#### 2 Check the Auto Deploy Enabled box to enable autodeployment, or uncheck this box to disable autodeployment.

#### 3 You can change the Auto Deploy Directory.

You can enter an absolute or relative path. A relative path is relative to *domain-dir*. The default is *domain-dir/autodeploy*.

#### 4 Check the Precompile Enabled box to precompile any JSP files.

**See Also** For details, click the Help button in the Admin Console.

---

## Tools for Deployment

This section discusses the various tools that can be used to deploy web applications. The deployment tools include:

- [“The asadmin Deployment Commands” on page 21](#)
- [“The Admin Console Deployment Pages” on page 21](#)

### The asadmin Deployment Commands

You can use the following asadmin commands to deploy or undeploy web modules on local servers.

- `deploy` — Deploys a web module. If the module is already deployed or already exists, you can force redeployment by setting the `--force` option to `true`.
- `redeploy` — Redeploys a web module that is already deployed. Whenever a redeployment is done, the sessions at that transit time become invalid. The client must restart the session.
- `undeploy` — Undeploys a web module.
- `disable` — Immediately disables a web module. Disabling a web module makes it inaccessible to clients.
- `enable` — Immediately enables a web module.
- `list-components` — Lists all deployed web modules.

---

**Note** – The `deploydir` command is deprecated.

---

For details, see the *GlassFish v3 Application Server Reference Manual*.

### The Admin Console Deployment Pages

You can use the Admin Console to deploy modules and applications to both local and remote Application Server sites.

#### ▼ To Use the Admin Console for Deployment

**1 Open the Applications component.**

**2 Go to the Web Applications page.**

You can undeploy, enable, or disable an application or module from the table on this page.

**3 Click on the Deploy button.**

On this page, you type the path to the WAR file.

**See Also** For details, click the Help button in the Admin Console.

## Deploying a WAR Module

You deploy a WAR module as described in [“Tools for Deployment” on page 21](#). If you do not specify a context root, the default is the name of the WAR file without the extension.

If a web application accesses a `DataSource` that is not specified in a `resource-ref` in `sun-web.xml`, or there is no `sun-web.xml` file, the `resource-ref-name` defined in `web.xml` is used. A warning message is logged recording the JNDI name used to look up the resource.

You can keep the generated source for JSP files by adding the `-keepgenerated` flag to the `jsp-config` element in `sun-web.xml`. If you include this property when you deploy the WAR module, the generated source is kept in `domain-dir/generated/jsp/app-name/module-name`, if it is in an application, or `domain-dir/generated/jsp/module-name`, if it is in an individually deployed web module.

For more information about JSP precompilation, see [“jsp-config” on page 66](#).

Web module context roots must be unique within a server instance.

## Deployment Descriptor Files

---

This chapter describes deployment descriptor files specific to the GlassFish Application Server in the following sections:

- “Application Server Descriptor” on page 17
- “The sun-web.xml File” on page 24
- “The sun-ejb-jar.xml File” on page 28

---

**Note** – For GlassFish v3 Technology Preview 2, web services are not supported unless the optional Metro module is downloaded from the Update Center. Without the Metro module, a servlet or EJB component cannot be a web service endpoint, and the sun-web.xml and sun-ejb-jar.xml elements related to web services are ignored.

EJB modules are not supported unless the optional EJB container module is downloaded from the Update Center. For GlassFish v3 Technology Preview 2, only stateless session beans with local interfaces and entity beans that use the Java Persistence API are supported. Stateful, message-driven, and EJB 2.0 and 2.1 entity beans are not supported. Remote interfaces and remote business interfaces for any of the bean types are not supported. The sun-ejb-jar.xml elements related to these features are ignored.

For information about the Update Center, see the *GlassFish v3 Application Server Quick Start Guide*.

---

## Sun Java System Application Server Descriptors

GlassFish Application Server uses optional deployment descriptors in addition to the Java EE standard descriptors for configuring features specific to the Application Server.

---

**Note** – Settings in the Application Server deployment descriptors override corresponding settings in the Java EE deployment descriptors and in the Application Server's `domain.xml` file unless otherwise stated.

---

Each deployment descriptor (or XML) file has a corresponding DTD file, which defines the elements, data, and attributes that the deployment descriptor file can contain. For example, the `sun-web-app_2_5-0.dtd` file defines the structure of the `sun-web.xml` file. The DTD files for the Application Server deployment descriptors are located in the `as-install/lib/dtds` directory.

---

**Note** – Do not edit the DTD files; their contents change only with new versions of the Application Server.

---

For general information about DTD files and XML, see the XML specification at <http://www.w3.org/TR/REC-xml>.

The following table lists the Application Server deployment descriptors and their DTD files.

TABLE A-1 GlassFish Application Server Descriptors

Deployment Descriptor	DTD File	Description
<code>sun-web.xml</code>	<code>sun-web-app_2_5-0.dtd</code>	Configures a web application (WAR file).
<code>sun-ejb-jar.xml</code>	<code>sun-ejb-jar_3_0-0.dtd</code>	Configures an enterprise bean (EJB JAR file).

---

**Note** – The Application Server deployment descriptors must be readable and writable by the file owners.

In each deployment descriptor file, subelements must be defined in the order in which they are listed under each **Subelements** heading, unless otherwise noted.

---

## The sun-web.xml File

The element hierarchy in the `sun-web.xml` file is as follows:

```
sun-web-app
. context-root
. security-role-mapping
```



---

```

. . role-name
. . principal-name
. . group-name
. servlet
. . servlet-name
. . principal-name
. . webservice-endpoint
. . . port-component-name
. . . endpoint-address-uri
. . . login-config
. . . . auth-method
. . . message-security-binding
. . . . message-security
. . . . . message
. . . . . . java-method
. . . . . . . method-name
. . . . . . . method-params
. . . . . . . . method-param
. . . . . . . operation-name
. . . . . . request-protection
. . . . . . response-protection
. . . transport-guarantee
. . . service-qname
. . . tie-class
. . . servlet-impl-class
. . . debugging-enabled
. . . property (with attributes)
. . . . description
. idempotent-url-pattern
. session-config
. . session-manager
. . . manager-properties
. . . . property (with attributes)
. . . . . description
. . . . store-properties
. . . . . property (with attributes)
. . . . . . description
. . session-properties
. . . property (with attributes)
. . . . description
. . cookie-properties
. . . property (with attributes)
. . . . description
. ejb-ref
. . ejb-ref-name
. . jndi-name
. resource-ref
. . res-ref-name

```

- . . jndi-name
- . . default-resource-principal
  - . . . name
  - . . . password
- . resource-env-ref
  - . . resource-env-ref-name
  - . . jndi-name
- . service-ref
  - . . service-ref-name
  - . . port-info
    - . . . service-endpoint-interface
    - . . . wsdl-port
      - . . . . namespaceURI
      - . . . . localpart
    - . . . stub-property
      - . . . . name
      - . . . . value
    - . . . call-property
      - . . . . name
      - . . . . value
    - . . . message-security-binding
      - . . . . message-security
        - . . . . . message
          - . . . . . . java-method
            - . . . . . . . method-name
            - . . . . . . . method-params
              - . . . . . . . . method-param
            - . . . . . . . operation-name
          - . . . . . . request-protection
          - . . . . . . response-protection
- . . call-property
  - . . . name
  - . . . value
- . . wsdl-override
- . . service-impl-class
- . . service-qname
  - . . . namespaceURI
  - . . . localpart
- . message-destination-ref
  - . . message-destination-ref-name
  - . . jndi-name
- . cache
  - . . cache-helper
    - . . . property (with attributes)
      - . . . . description
  - . . default-helper
    - . . . property (with attributes)
      - . . . . description

- . . property (with attributes)
  - . . . description
- . . cache-mapping
  - . . . servlet-name
  - . . . url-pattern
  - . . . cache-helper-ref
  - . . . dispatcher
  - . . . timeout
  - . . . refresh-field
  - . . . http-method
  - . . . key-field
  - . . . constraint-field
    - . . . . constraint-field-value
- . class-loader
  - . . property (with attributes)
    - . . . description
- . jsp-config
- . locale-charset-info
  - . . locale-charset-map
- . . parameter-encoding
- . parameter-encoding
  - . . property (with attributes)
    - . . . description
- . message-destination
  - . . message-destination-name
  - . . jndi-name
- . webservice-description
  - . . webservice-description-name
  - . . wsdl-publish-location

Here is a sample sun-web.xml file:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sun-web-app PUBLIC "-//Sun Microsystems, Inc.//DTD
Application Server 9.0 Servlet 2.5//EN"
'http://www.sun.com/software/appserver/dtds/sun-web-app_2_5-0.dtd'>
<sun-web-app>
  <session-config>
    <session-manager/>
  </session-config>
  <resource-ref>
    <res-ref-name>mail/Session</res-ref-name>
    <jndi-name>mail/Session</jndi-name>
  </resource-ref>
  <jsp-config/>
</sun-web-app>
```

## The sun-ejb-jar.xml File

The element hierarchy in the sun-ejb-jar.xml file is as follows:

```
sun-ejb-jar
. security-role-mapping
. . role-name
. . principal-name
. . group-name
. enterprise-beans
. . name
. . unique-id
. . ejb
. . . ejb-name
. . . jndi-name
. . . ejb-ref
. . . . ejb-ref-name
. . . . jndi-name
. . . . resource-ref
. . . . . res-ref-name
. . . . . jndi-name
. . . . . default-resource-principal
. . . . . . name
. . . . . . password
. . . . . resource-env-ref
. . . . . . resource-env-ref-name
. . . . . . jndi-name
. . . . . service-ref
. . . . . . service-ref-name
. . . . . . port-info
. . . . . . service-endpoint-interface
. . . . . . . wsdl-port
. . . . . . . . namespaceURI
. . . . . . . . localpart
. . . . . . . stub-property
. . . . . . . . name
. . . . . . . . value
. . . . . . . call-property
. . . . . . . . name
. . . . . . . . value
. . . . . . . message-security-binding
. . . . . . . . message-security
. . . . . . . . . message
. . . . . . . . . . java-method
. . . . . . . . . . . method-name
. . . . . . . . . . . method-params
. . . . . . . . . . . . method-param
. . . . . . . . . . . . . operation-name
```

---

```
. . . . . request-protection
. . . . . response-protection
. . . . call-property
. . . . . name
. . . . . value
. . . . wsdl-override
. . . . service-impl-class
. . . . service-qname
. . . . . namespaceURI
. . . . . localpart
. . . message-destination-ref
. . . . message-destination-ref-name
. . . . jndi-name
. . . pass-by-reference
. . . cmp
. . . . mapping-properties
. . . . . is-one-one-cmp
. . . . . one-one-finders
. . . . . . finder
. . . . . . . method-name
. . . . . . . query-params
. . . . . . . query-filter
. . . . . . . query-variables
. . . . . . . query-ordering
. . . . prefetch-disabled
. . . . . query-method
. . . . . . method-name
. . . . . . . method-params
. . . . . . . . . method-param
. . . principal
. . . . name
. . . mdb-connection-factory
. . . . jndi-name
. . . . default-resource-principal
. . . . . name
. . . . . password
. . . jms-durable-subscription-name
. . . jms-max-messages-load
. . . ior-security-config
. . . . transport-config
. . . . . integrity
. . . . . confidentiality
. . . . . . establish-trust-in-target
. . . . . . establish-trust-in-client
. . . . as-context
. . . . . auth-method
. . . . . realm
. . . . . required
```

```

. . . . sas-context
. . . . . caller-propagation
. . . is-read-only-bean
. . . refresh-period-in-seconds
. . . commit-option
. . . cmt-timeout-in-seconds
. . . use-thread-pool-id
. . . gen-classes
. . . . remote-impl
. . . . local-impl
. . . . remote-home-impl
. . . . local-home-impl
. . . bean-pool
. . . . steady-pool-size
. . . . resize-quantity
. . . . max-pool-size
. . . . pool-idle-timeout-in-seconds
. . . . max-wait-time-in-millis
. . . bean-cache
. . . . max-cache-size
. . . . resize-quantity
. . . . is-cache-overflow-allowed
. . . . cache-idle-timeout-in-seconds
. . . . removal-timeout-in-seconds
. . . . victim-selection-policy
. . . mdb-resource-adapter
. . . . resource-adapter-mid
. . . . activation-config
. . . . . description
. . . . . activation-config-property
. . . . . . activation-config-property-name
. . . . . . activation-config-property-value
. . . webservice-endpoint
. . . . port-component-name
. . . . endpoint-address-uri
. . . . login-config
. . . . . auth-method
. . . . . realm
. . . . message-security-binding
. . . . . message-security
. . . . . . message
. . . . . . . java-method
. . . . . . . . method-name
. . . . . . . . method-params
. . . . . . . . . method-param
. . . . . . . operation-name
. . . . . . request-protection
. . . . . . response-protection

```

---

```
. . . . transport-guarantee
. . . . service-qname
. . . . tie-class
. . . . servlet-impl-class
. . . . debugging-enabled
. . . . property (with subelements)
. . . . . name
. . . . . value
. . . flush-at-end-of-method
. . . . method
. . . . . description
. . . . . ejb-name
. . . . . method-name
. . . . . method-intf
. . . . . method-params
. . . . . . method-param
. . . checkpointed-methods
. . . checkpoint-at-end-of-method
. . . . method
. . . . . description
. . . . . ejb-name
. . . . . method-name
. . . . . method-intf
. . . . . method-params
. . . . . . method-param
. . pm-descriptors
. . cmp-resource
. . . jndi-name
. . . default-resource-principal
. . . . name
. . . . password
. . . property (with subelements)
. . . . name
. . . . value
. . . create-tables-at-deploy
. . . drop-tables-at-undeploy
. . . database-vendor-name
. . . schema-generator-properties
. . . . property (with subelements)
. . . . . name
. . . . . value
. . message-destination
. . . message-destination-name
. . . jndi-name
. . webservice-description
. . . webservice-description-name
. . . wsdl-publish-location
```

---

**Note** – If any configuration information for an enterprise bean is not specified in the `sun-ejb-jar.xml` file, it defaults to a corresponding setting in the EJB container if an equivalency exists.

---

Here is a sample `sun-ejb-jar.xml` file:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sun-ejb-jar PUBLIC "-//Sun Microsystems, Inc.//DTD
Application Server 9.0 EJB 3.0//EN"
'http://www.sun.com/software/appserver/dtds/sun-ejb-jar_3_0-0.dtd'>
<sun-ejb-jar>
<display-name>First Module</display-name>
<enterprise-beans>
  <ejb>
    <ejb-name>CustomerEJB</ejb-name>
    <jndi-name>customer</jndi-name>
    <bean-pool>
      <steady-pool-size>10</steady-pool-size>
      <resize-quantity>10</resize-quantity>
      <max-pool-size>100</max-pool-size>
      <pool-idle-timeout-in-seconds>600</pool-idle-timeout-in-seconds>
    </bean-pool>
    <bean-cache>
      <max-cache-size>100</max-cache-size>
      <resize-quantity>10</resize-quantity>
      <removal-timeout-in-seconds>3600</removal-timeout-in-seconds>
      <victim-selection-policy>LRU</victim-selection-policy>
    </bean-cache>
  </ejb>
  <cmp-resource>
    <jndi-name>jdbc/___default</jndi-name>
    <create-tables-at-deploy>true</create-tables-at-deploy>
    <drop-tables-at-undeploy>true</drop-tables-at-undeploy>
  </cmp-resource>
</enterprise-beans>
</sun-ejb-jar>
```

## A

### activation-config

Specifies an activation configuration, which includes the runtime configuration properties of the message-driven bean in its operational environment. For example, this can include



information about the name of a physical JMS destination. Matches and overrides the `activation-config` element in the `ejb-jar.xml` file.

## Superelements

[“mdb-resource-adapter” on page 77](#) (`sun-ejb-jar.xml`)

## Subelements

The following table describes subelements for the `activation-config` element.

TABLE A-2 `activation-config` subelements

Element	Required	Description
<a href="#">“description” on page 52</a>	zero or one	Specifies a text description of the activation configuration.
<a href="#">“activation-config-property” on page 33</a>	one or more	Specifies an activation configuration property.

## activation-config-property

Specifies the name and value of an activation configuration property.

## Superelements

[“activation-config” on page 32](#) (`sun-ejb-jar.xml`)

## Subelements

The following table describes subelements for the `activation-config-property` element.

TABLE A-3 `activation-config-property` subelements

Element	Required	Description
<a href="#">“activation-config-property-name” on page 33</a>	only one	Specifies the name of an activation configuration property.
<a href="#">“activation-config-property-value” on page 34</a>	only one	Specifies the value of an activation configuration property.

## activation-config-property-name

Specifies the name of an activation configuration property.

## Superelements

[“activation-config-property” on page 33](#) (`sun-ejb-jar.xml`)

## Subelements

none - contains data

## activation-config-property-value

Specifies the value of an activation configuration property.

## Superelements

[“activation-config-property” on page 33](#) (sun-ejb-jar.xml)

## Subelements

none - contains data

## as-context

Specifies the authentication mechanism used to authenticate the client.

## Superelements

[“ior-security-config” on page 63](#) (sun-ejb-jar.xml)

## Subelements

The following table describes subelements for the as - context element.

TABLE A-4 as - context Subelements

Element	Required	Description
<a href="#">“auth-method” on page 34</a>	only one	Specifies the authentication method. The only supported value is USERNAME_PASSWORD.
<a href="#">“realm” on page 93</a>	only one	Specifies the realm in which the user is authenticated.
<a href="#">“required” on page 96</a>	only one	Specifies whether the authentication method specified in the auth - method element must be used for client authentication.

## auth-method

Specifies the authentication method.

If the parent element is [“as-context” on page 34](#), the only supported value is USERNAME\_PASSWORD.

If the parent element is “[login-config](#)” on page 73, specifies the authentication mechanism for the web service endpoint. As a prerequisite to gaining access to any web resources protected by an authorization constraint, a user must be authenticated using the configured mechanism.

## Superelements

“[login-config](#)” on page 73 (`sun-web.xml`), “[as-context](#)” on page 34 (`sun-ejb-jar.xml`)

## Subelements

none - contains data

# B

## bean-cache

Specifies the entity bean cache properties. Used for entity beans and stateful session beans.

## Superelements

“[ejb](#)” on page 53 (`sun-ejb-jar.xml`)

## Subelements

The following table describes subelements for the `bean-cache` element.

TABLE A-5 `bean-cache` Subelements

Element	Required	Description
“ <a href="#">max-cache-size</a> ” on page 75	zero or one	Specifies the maximum number of beans allowable in cache.
“ <a href="#">is-cache-overflow-allowed</a> ” on page 63	zero or one	Deprecated.
“ <a href="#">cache-idle-timeout-in-seconds</a> ” on page 40	zero or one	Specifies the maximum time that a stateful session bean or entity bean is allowed to be idle in cache before being passivated. Default value is 10 minutes (600 seconds).
“ <a href="#">removal-timeout-in-seconds</a> ” on page 94	zero or one	Specifies the amount of time a bean remains before being removed. If <code>removal-timeout-in-seconds</code> is less than <code>idle-timeout</code> , the bean is removed without being passivated.
“ <a href="#">resize-quantity</a> ” on page 97	zero or one	Specifies the number of beans to be created if the pool is empty (subject to the <code>max-pool-size</code> limit). Values are from 0 to <code>MAX_INTEGER</code> .

TABLE A-5 bean-cache Subelements (Continued)

Element	Required	Description
<a href="#">“victim-selection-policy” on page 118</a>	zero or one	Specifies the algorithm that must be used by the container to pick victims. Applies only to stateful session beans.

## Example

```
<bean-cache>
  <max-cache-size>100</max-cache-size>
  <cache-resize-quantity>10</cache-resize-quantity>
  <removal-timeout-in-seconds>3600</removal-timeout-in-seconds>
  <victim-selection-policy>LRU</victim-selection-policy>
  <cache-idle-timeout-in-seconds>600</cache-idle-timeout-in-seconds>
  <removal-timeout-in-seconds>5400</removal-timeout-in-seconds>
</bean-cache>
```

## bean-pool

Specifies the pool properties of stateless session beans, entity beans, and message-driven bean.

## Superelements

[“ejb” on page 53](#) (sun-ejb-jar.xml)

## Subelements

The following table describes subelements for the bean-pool element.

TABLE A-6 bean-pool Subelements

Element	Required	Description
<a href="#">“steady-pool-size” on page 108</a>	zero or one	Specifies the initial and minimum number of beans maintained in the pool. Default is 32.
<a href="#">“resize-quantity” on page 97</a>	zero or one	Specifies the number of beans to be created if the pool is empty (subject to the max-pool-size limit). Values are from 0 to MAX_INTEGER.
<a href="#">“max-pool-size” on page 76</a>	zero or one	Specifies the maximum number of beans in the pool. Values are from 0 to MAX_INTEGER. Default is to the EJB container value or 60.
<a href="#">“max-wait-time-in-millis” on page 76</a>	zero or one	Deprecated.

TABLE A-6 bean-pool Subelements (Continued)

Element	Required	Description
<a href="#">“pool-idle-timeout-in-seconds” on page 87</a>	zero or one	Specifies the maximum time that a bean is allowed to be idle in the pool. After this time, the bean is removed. This is a hint to the server. Default time is 600 seconds (10 minutes).

### Example

```
<bean-pool>
  <steady-pool-size>10</steady-pool-size>
  <resize-quantity>10</resize-quantity>
  <max-pool-size>100</max-pool-size>
  <pool-idle-timeout-in-seconds>600</pool-idle-timeout-in-seconds>
</bean-pool>
```

## C

### cache

Configures caching for web application components.

### Superelements

[“sun-web-app” on page 111](#) (sun-web.xml)

### Subelements

The following table describes subelements for the cache element.

TABLE A-7 cache Subelements

Element	Required	Description
<a href="#">“cache-helper” on page 39</a>	zero or more	Specifies a custom class that implements the CacheHelper interface.
<a href="#">“default-helper” on page 50</a>	zero or one	Allows you to change the properties of the default, built-in <a href="#">“cache-helper” on page 39</a> class.
<a href="#">“property (with attributes)” on page 90</a>	zero or more	Specifies a cache property, which has a name and a value.
<a href="#">“cache-mapping” on page 40</a>	zero or more	Maps a URL pattern or a servlet name to its cacheability constraints.

### Attributes

The following table describes attributes for the cache element.

TABLE A-8 cache Attributes

Attribute	Default	Description
max-entries	4096	(optional) Specifies the maximum number of entries the cache can contain. Must be a positive integer.
timeout-in-seconds	30	(optional) Specifies the maximum amount of time in seconds that an entry can remain in the cache after it is created or refreshed. Can be overridden by a “ <a href="#">timeout</a> ” on <a href="#">page 115</a> element.
enabled	true	(optional) Determines whether servlet and JSP caching is enabled.

## Properties

The following table describes properties for the cache element.

TABLE A-9 cache Properties

Property	Default	Description
cacheClassName	com.sun.appserv.web.cache.LruCache	Specifies the fully qualified name of the class that implements the cache functionality. See “ <a href="#">Cache Class Names</a> ” on <a href="#">page 38</a> for possible values.
MultiLRUSegmentSize	4096	Specifies the number of entries in a segment of the cache table that should have its own LRU (least recently used) list. Applicable only if cacheClassName is set to com.sun.appserv.web.cache.MultiLruCache.
MaxSize	unlimited; Long.MAX_VALUE	Specifies an upper bound on the cache memory size in bytes (KB or MB units). Example values are 32 KB or 2 MB. Applicable only if cacheClassName is set to com.sun.appserv.web.cache.BoundedMultiLruCache.

## Cache Class Names

The following table lists possible values of the cacheClassName property.

TABLE A-10 cacheClassName Values

Value	Description
com.sun.appserv.web.cache.LruCache	A bounded cache with an LRU (least recently used) cache replacement policy.
com.sun.appserv.web.cache.BaseCache	An unbounded cache suitable if the maximum number of entries is known.
com.sun.appserv.web.cache.MultiLruCache	A cache suitable for a large number of entries (>4096). Uses the MultiLRUSegmentSize property.

TABLE A-10 cacheClassName Values (Continued)

Value	Description
com.sun.appserv.web.cache.BoundedMultiLruCache	A cache suitable for limiting the cache size by memory rather than number of entries. Uses the MaxSize property.

## cache-helper

Specifies a class that implements the `com.sun.appserv.web.cache.CacheHelper` interface.

### Superelements

[“cache” on page 37](#) (`sun-web.xml`)

### Subelements

The following table describes subelements for the `cache-helper` element.

TABLE A-11 cache-helper Subelements

Element	Required	Description
<a href="#">“property (with attributes)” on page 90</a>	zero or more	Specifies a property, which has a name and a value.

### Attributes

The following table describes attributes for the `cache-helper` element.

TABLE A-12 cache-helper Attributes

Attribute	Default	Description
name	default	Specifies a unique name for the helper class, which is referenced in the <a href="#">“cache-mapping” on page 40</a> element.
class-name	none	Specifies the fully qualified class name of the cache helper, which must implement the <code>com.sun.appserv.web.CacheHelper</code> interface.

## cache-helper-ref

Specifies the name of the [“cache-helper” on page 39](#) used by the parent [“cache-mapping” on page 40](#) element.

### Superelements

[“cache-mapping” on page 40](#) (`sun-web.xml`)

## Subelements

none - contains data

## cache-idle-timeout-in-seconds

Specifies the maximum time that a bean can remain idle in the cache. After this amount of time, the container can passivate this bean. A value of 0 specifies that beans never become candidates for passivation. Default is 600.

Applies to stateful session beans and entity beans.

## Superelements

[“bean-cache” on page 35](#) (sun-ejb-jar.xml)

## Subelements

none - contains data

## cache-mapping

Maps a URL pattern or a servlet name to its cacheability constraints.

## Superelements

[“cache” on page 37](#) (sun-web.xml)

## Subelements

The following table describes subelements for the cache-mapping element.

TABLE A-13 cache-mapping Subelements

Element	Required	Description
<a href="#">“servlet-name” on page 106</a>	requires one <code>servlet-name</code> or <code>url-pattern</code>	Contains the name of a servlet.
<a href="#">“url-pattern” on page 117</a>	requires one <code>servlet-name</code> or <code>url-pattern</code>	Contains a servlet URL pattern for which caching is enabled.
<a href="#">“cache-helper-ref” on page 39</a>	required if <code>dispatcher</code> , <code>timeout</code> , <code>refresh-field</code> , <code>http-method</code> , <code>key-field</code> , and <code>constraint-field</code> are not used	Contains the name of the <a href="#">“cache-helper” on page 39</a> used by the parent cache-mapping element.



TABLE A-13 cache-mapping Subelements (Continued)

Element	Required	Description
<a href="#">“dispatcher” on page 52</a>	zero or one if cache-helper-ref is not used	Contains a comma-separated list of RequestDispatcher methods for which caching is enabled.
<a href="#">“timeout” on page 115</a>	zero or one if cache-helper-ref is not used	Contains the <a href="#">“cache-mapping” on page 40</a> specific maximum amount of time in seconds that an entry can remain in the cache after it is created or refreshed.
<a href="#">“refresh-field” on page 93</a>	zero or one if cache-helper-ref is not used	Specifies a field that gives the application component a programmatic way to refresh a cached entry.
<a href="#">“http-method” on page 62</a>	zero or more if cache-helper-ref is not used	Contains an HTTP method that is eligible for caching.
<a href="#">“key-field” on page 69</a>	zero or more if cache-helper-ref is not used	Specifies a component of the key used to look up and extract cache entries.
<a href="#">“constraint-field” on page 46</a>	zero or more if cache-helper-ref is not used	Specifies a cacheability constraint for the given url-pattern or servlet-name.

## call-property

Specifies JAX-RPC property values that can be set on a `javax.xml.rpc.Call` object before it is returned to the web service client. The property names can be any properties supported by the JAX-RPC `Call` implementation.

### Superelements

[“port-info” on page 88](#), [“service-ref” on page 104](#) (`sun-web.xml`, `sun-ejb-jar.xml`)

### Subelements

The following table describes subelements for the `call-property` element.

TABLE A-14 call-property subelements

Element	Required	Description
<a href="#">“name” on page 83</a>	only one	Specifies the name of the entity.
<a href="#">“value” on page 118</a>	only one	Specifies the value of the entity.

## caller-propagation

Specifies whether the target accepts propagated caller identities. The values are NONE, SUPPORTED, or REQUIRED.

## Superelements

[“sas-context” on page 100](#) (sun-ejb-jar.xml)

## Subelements

none - contains data

## checkpoint-at-end-of-method

Specifies that the stateful session bean state is checkpointed, or persisted, after the specified methods are executed. The `availability-enabled` attribute of the parent [“ejb” on page 53](#) element must be set to `true`.

## Superelements

[“ejb” on page 53](#) (sun-ejb-jar.xml)

## Subelements

The following table describes subelements for the `checkpoint-at-end-of-method` element.

TABLE A-15 `checkpoint-at-end-of-method` Subelements

Element	Required	Description
<a href="#">“method” on page 81</a>	one or more	Specifies a bean method.

## checkpointed-methods

Deprecated. Supported for backward compatibility. Use [“checkpoint-at-end-of-method” on page 42](#) instead.

## Superelements

[“ejb” on page 53](#) (sun-ejb-jar.xml)

## class-loader

Configures the class loader for the web module.

## Superelements

[“sun-web-app” on page 111](#) (sun-web.xml)

## Subelements

The following table describes subelements for the `class-loader` element.

TABLE A-16 `class-loader` Subelements

Element	Required	Description
“property (with attributes)” on page 90	zero or more	Specifies a property, which has a name and a value.

## Attributes

The following table describes attributes for the `class-loader` element.

TABLE A-17 `class-loader` Attributes

Attribute	Default	Description
<code>extra-class-path</code>	null	(optional) Specifies a colon or semicolon separated list of additional classpaths for this web module. Paths can be absolute or relative to the web module's root, for example: <code>extra-class-path="WEB-INF/lib/extra/extra.jar"</code>
<code>delegate</code>	true	(optional) If true, the web module follows the standard class loader delegation model and delegates to its parent class loader first before looking in the local class loader. You must set this to true for a web application that accesses EJB components or that acts as a web service client or endpoint.  If false, the web module follows the delegation model specified in the Servlet specification and looks in its class loader before looking in the parent class loader. It's safe to set this to false only for a web module that does not interact with any other modules.  <b>Note</b> – For Technology Preview 2, the <code>delegate</code> value is ignored and assumed to be set to true.
<code>dynamic-reload-interval</code>		(optional) Not implemented. Included for backward compatibility with previous GlassFish Web Server versions.

**Note** – If the `delegate` element is set to false, the class loader delegation behavior complies with the Servlet 2.4 specification, section 9.7.2. If set to its default value of true, classes and resources residing in container-wide library JAR files are loaded in preference to classes and resources packaged within the WAR file.

Portable programs that use this element should not be packaged with any classes or interfaces that are a part of the Java EE specification. The behavior of a program that includes such classes or interfaces in its WAR file is undefined.

## Properties

The following table describes properties for the `class-loader` element.

TABLE A-18 `class-loader` Properties

Property	Default	Description
<code>ignoreHiddenJarFiles</code>	false	If true, specifies that all JAR and ZIP files in the <code>WEB-INF/lib</code> directory that start with a period (.) are ignored by the class loader.

## cmp

Describes runtime information for a CMP entity bean object for EJB 1.1 and EJB 2.1 beans.

### Superelements

[“ejb” on page 53](#) (`sun-ejb-jar.xml`)

### Subelements

The following table describes subelements for the `cmp` element.

TABLE A-19 `cmp` Subelements

Element	Required	Description
<a href="#">“mapping-properties” on page 75</a>	zero or one	This element is not implemented.
<a href="#">“is-one-one-cmp” on page 64</a>	zero or one	This element is not implemented.
<a href="#">“one-one-finders” on page 84</a>	zero or one	Describes the finders for CMP 1.1 beans.
<a href="#">“prefetch-disabled” on page 88</a>	zero or one	Disables prefetching of entity bean states for the specified query methods.

## cmp-resource

Specifies the database to be used for storing CMP beans.

### Superelements

[“enterprise-beans” on page 58](#) (`sun-ejb-jar.xml`)

### Subelements

The following table describes subelements for the `cmp-resource` element.

TABLE A-20 cmp - resource Subelements

Element	Required	Description
<a href="#">“jndi-name” on page 65</a>	only one	Specifies the absolute <code>jndi</code> - name of a JDBC resource.
<a href="#">“default-resource-principal” on page 51</a>	zero or one	Specifies the default runtime bindings of a resource reference.
<a href="#">“property (with subelements)” on page 91</a>	zero or more	Specifies a property name and value. Used to configure <code>PersistenceManagerFactory</code> properties.
<a href="#">“create-tables-at-deploy” on page 49</a>	zero or one	If <code>true</code> , specifies that database tables are created for beans that are automatically mapped by the EJB container.
<a href="#">“drop-tables-at-undeploy” on page 52</a>	zero or one	If <code>true</code> , specifies that database tables that were automatically created when the bean(s) were last deployed are dropped when the bean(s) are undeployed.
<a href="#">“database-vendor-name” on page 50</a>	zero or one	Specifies the name of the database vendor for which tables can be created.
<a href="#">“schema-generator-properties” on page 101</a>	zero or one	Specifies field-specific type mappings and allows you to set the <code>use-unique-table-names</code> property.

## cmt-timeout-in-seconds

Overrides the Transaction Timeout setting of the Transaction Service for an individual bean. The default value, `0`, specifies that the default Transaction Service timeout is used. If positive, this value is used for all methods in the bean that start a new container-managed transaction. This value is *not* used if the bean joins a client transaction.

## Superelements

[“ejb” on page 53](#) (`sun-ejb-jar.xml`)

## Subelements

none - contains data

## commit-option

Specifies the commit option used on transaction completion. Valid values for the Application Server are B or C. Default value is B. Applies to entity beans.

---

**Note** – Commit option A is not supported for this Application Server release.

---

## Superelements

“[ejb](#)” on page 53 (`sun-ejb-jar.xml`)

## Subelements

none - contains data

## confidentiality

Specifies if the target supports privacy-protected messages. The values are NONE, SUPPORTED, or REQUIRED.

## Superelements

“[transport-config](#)” on page 116 (`sun-ejb-jar.xml`)

## Subelements

none - contains data

## constraint-field

Specifies a cacheability constraint for the given “[url-pattern](#)” on page 117 or “[servlet-name](#)” on page 106.

All `constraint-field` constraints must pass for a response to be cached. If there are value constraints, at least one of them must pass.

## Superelements

“[cache-mapping](#)” on page 40 (`sun-web.xml`)

## Subelements

The following table describes subelements for the `constraint-field` element.

TABLE A-21 `constraint-field` Subelements

Element	Required	Description
“ <a href="#">constraint-field-value</a> ” on page 47	zero or more	Contains a value to be matched to the input parameter value.

## Attributes

The following table describes attributes for the `constraint-field` element.

TABLE A-22 constraint-field Attributes

Attribute	Default	Description
name	none	Specifies the input parameter name.
scope	request.parameter	(optional) Specifies the scope from which the input parameter is retrieved. Allowed values are context.attribute, request.header, request.parameter, request.cookie, request.attribute, and session.attribute.
cache-on-match	true	(optional) If true, caches the response if matching succeeds. Overrides the same attribute in a “constraint-field-value” on page 47 subelement.
cache-on-match-failure	false	(optional) If true, caches the response if matching fails. Overrides the same attribute in a “constraint-field-value” on page 47 subelement.

## constraint-field-value

Specifies a value to be matched to the input parameter value. The matching is case sensitive. For example:

```
<value match-expr="in-range">1-60</value>
```

### Superelements

“constraint-field” on page 46 (sun-web.xml)

### Subelements

none - contains data

### Attributes

The following table describes attributes for the constraint-field-value element.

TABLE A-23 constraint-field-value Attributes

Attribute	Default	Description
match-expr	equals	(optional) Specifies the type of comparison performed with the value. Allowed values are equals, not-equals, greater, lesser, and in-range.  If match-expr is greater or lesser, the value must be a number. If match-expr is in-range, the value must be of the form $n1-n2$ , where $n1$ and $n2$ are numbers.
cache-on-match	true	(optional) If true, caches the response if matching succeeds.

TABLE A-23 constraint-field-value Attributes (Continued)

Attribute	Default	Description
cache-on-match-failure	false	(optional) If true, caches the response if matching fails.

## context-root

Contains the web context root for the web application. Overrides the corresponding element in the web.xml file.

### Superelements

“sun-web-app” on page 111 (sun-web.xml)

### Subelements

none - contains data

## cookie-properties

Specifies session cookie properties.

### Superelements

“session-config” on page 106 (sun-web.xml)

### Subelements

The following table describes subelements for the cookie-properties element.

TABLE A-24 cookie-properties Subelements

Element	Required	Description
“property (with attributes)” on page 90	zero or more	Specifies a property, which has a name and a value.

## Properties

The following table describes properties for the cookie-properties element.



TABLE A-25 cookie-properties Properties

Property	Default	Description
cookiePath	Context path at which the web module is installed.	Specifies the pathname that is set when the cookie is created. The browser sends the cookie if the pathname for the request contains this pathname. If set to / (slash), the browser sends cookies to all URLs served by the Application Server. You can set the path to a narrower mapping to limit the request URLs to which the browser sends cookies.
cookieMaxAgeSeconds	-1	Specifies the expiration time (in seconds) after which the browser expires the cookie.
cookieDomain	(unset)	Specifies the domain for which the cookie is valid.
cookieComment	GlassFish Application Server Session Tracking Cookie	Specifies the comment that identifies the session tracking cookie in the cookie file. Applications can provide a more specific comment for the cookie.
cookieSecure	dynamic	Sets the Secure attribute of any JSESSION or JSESSIONIDSSO cookies associated with the web application. Allowed values are as follows: <ul style="list-style-type: none"> <li>■ true — Sets Secure to true.</li> <li>■ false — Sets Secure to false.</li> <li>■ dynamic — Sets Secure to the security setting of the request that generated the cookie.</li> </ul>

## create-tables-at-deploy

Specifies whether database tables are created for beans that are automatically mapped by the EJB container. If `true`, creates tables in the database. If `false` (the default if this element is not present), does not create tables.

This element can be overridden during deployment. See *GlassFish v3 Application Server Developer's Guide*.

## Superelements

“`cmp-resource`” on page 44 (`sun-ejb-jar.xml`)

## Subelements

none - contains data

## D

### database-vendor-name

Specifies the name of the database vendor for which tables can be created. Allowed values are `javadb`, `db2`, `mssql`, `oracle`, `postgresql`, `pointbase`, `derby` (also for CloudScape), and `sybase`, case-insensitive.

If no value is specified, a connection is made to the resource specified by the “[jndi-name](#)” on [page 65](#) subelement of the “[cmp-resource](#)” on [page 44](#) element, and the database vendor name is read. If the connection cannot be established, or if the value is not recognized, SQL-92 compliance is presumed.

This element can be overridden during deployment. See *GlassFish v3 Application Server Developer's Guide*.

### Superelements

“[cmp-resource](#)” on [page 44](#) (`sun-ejb-jar.xml`)

### Subelements

none - contains data

### debugging-enabled

Specifies whether the debugging servlet is enabled for this web service endpoint. Allowed values are `true` (the default) and `false`.

### Superelements

“[webservice-endpoint](#)” on [page 120](#) (`sun-web.xml`, `sun-ejb-jar.xml`)

### Subelements

none - contains data

### default-helper

Passes property values to the built-in default “[cache-helper](#)” on [page 39](#) class.

### Superelements

“[cache](#)” on [page 37](#) (`sun-web.xml`)

## Subelements

The following table describes subelements for the `default-helper` element.

TABLE A-26 `default-helper` Subelements

Element	Required	Description
<a href="#">“property (with attributes)” on page 90</a>	zero or more	Specifies a property, which has a name and a value.

## Properties

The following table describes properties for the `default-helper` element.

TABLE A-27 `default-helper` Properties

Property	Default	Description
<code>cacheKeyGeneratorAttrName</code>	Uses the built-in default <a href="#">“cache-helper” on page 39</a> key generation, which concatenates the servlet path with <a href="#">“key-field” on page 69</a> values, if any.	The caching engine looks in the <code>ServletContext</code> for an attribute with a name equal to the value specified for this property to determine whether a customized <code>CacheKeyGenerator</code> implementation is used. An application can provide a customized key generator rather than using the default helper. See <a href="#">“The CacheKeyGenerator Interface” in <i>GlassFish v3 Application Server Developer’s Guide</i></a> .

## default-resource-principal

Specifies the default principal (user) for the resource.

## Superelements

[“resource-ref” on page 99](#) (`sun-web.xml`, `sun-ejb-jar.xml`); [“cmp-resource” on page 44](#), [“mdb-connection-factory” on page 76](#) (`sun-ejb-jar.xml`)

## Subelements

The following table describes subelements for the `default-resource-principal` element.

TABLE A-28 `default-resource-principal` Subelements

Element	Required	Description
<a href="#">“name” on page 83</a>	only one	Specifies the default resource principal name used to sign on to a resource manager.
<a href="#">“password” on page 86</a>	only one	Specifies password of the default resource principal.

## description

Specifies a text description of the containing element.

### Superelements

“[property \(with attributes\)](#)” on page 90 (`sun-web.xml`); “[activation-config](#)” on page 32, “[method](#)” on page 81 (`sun-ejb-jar.xml`)

### Subelements

none - contains data

## dispatcher

Specifies a comma-separated list of `RequestDispatcher` methods for which caching is enabled on the target resource. Valid values are `REQUEST`, `FORWARD`, `INCLUDE`, and `ERROR`. If this element is not specified, the default is `REQUEST`. See SRV.6.2.5 of the Servlet 2.4 specification for more information.

### Superelements

“[cache-mapping](#)” on page 40 (`sun-web.xml`)

### Subelements

none - contains data

## drop-tables-at-undeploy

Specifies whether database tables that were automatically created when the bean(s) were last deployed are dropped when the bean(s) are undeployed. If `true`, drops tables from the database. If `false` (the default if this element is not present), does not drop tables.

This element can be overridden during deployment. See *GlassFish v3 Application Server Developer's Guide*.

### Superelements

“[cmp-resource](#)” on page 44 (`sun-ejb-jar.xml`)

### Subelements

none - contains data

## E

## ejb

Defines runtime properties for a single enterprise bean within the application. The subelements listed below apply to particular enterprise beans as follows:

- All types of beans: `ejb-name`, `ejb-ref`, `resource-ref`, `resource-env-ref`, `ior-security-config`, `gen-classes`, `jndi-name`, `use-thread-pool-id`, `message-destination-ref`, `pass-by-reference`, `service-ref`
- Stateless session beans: `bean-pool`, `webservice-endpoint`
- Stateful session beans: `bean-cache`, `webservice-endpoint`, `checkpoint-at-end-of-method`
- Entity beans: `commit-option`, `bean-cache`, `bean-pool`, `cmp`, `is-read-only-bean`, `refresh-period-in-seconds`, `flush-at-end-of-method`
- Message-driven beans: `mdb-resource-adapter`, `mdb-connection-factory`, `jms-durable-subscription-name`, `jms-max-messages-load`, `bean-pool`

## Superelements

[“enterprise-beans” on page 58](#) (`sun-ejb-jar.xml`)

## Subelements

The following table describes subelements for the `ejb` element.

TABLE A-29 `ejb` Subelements

Element	Required	Description
<a href="#">“ejb-name” on page 56</a>	only one	Matches the <code>ejb-name</code> in the corresponding <code>ejb-jar.xml</code> file.
<a href="#">“jndi-name” on page 65</a>	zero or more	Specifies the absolute <code>jndi-name</code> .
<a href="#">“ejb-ref” on page 56</a>	zero or more	Maps the absolute JNDI name to the <code>ejb-ref</code> element in the corresponding Java EE XML file.
<a href="#">“resource-ref” on page 99</a>	zero or more	Maps the absolute JNDI name to the <code>resource-ref</code> in the corresponding Java EE XML file.
<a href="#">“resource-env-ref” on page 98</a>	zero or more	Maps the absolute JNDI name to the <code>resource-env-ref</code> in the corresponding Java EE XML file.
<a href="#">“service-ref” on page 104</a>	zero or more	Specifies runtime settings for a web service reference.

TABLE A-29 ejb Subelements (Continued)

Element	Required	Description
“message-destination-ref” on page 79	zero or more	Specifies the name of a physical message destination.
“pass-by-reference” on page 85	zero or one	Specifies the passing method used by an enterprise bean calling a remote interface method in another bean that is colocated within the same process.
“cmp” on page 44	zero or one	Specifies runtime information for a container-managed persistence (CMP) entity bean for EJB 1.1 and EJB 2.1 beans.
“principal” on page 89	zero or one	Specifies the principal (user) name in an enterprise bean that has the run-as role specified.
“mdb-connection-factory” on page 76	zero or one	Specifies the connection factory associated with a message-driven bean.
“jms-durable-subscription-name” on page 65	zero or one	Specifies the durable subscription associated with a message-driven bean.
“jms-max-messages-load” on page 65	zero or one	Specifies the maximum number of messages to load into a Java Message Service session at one time for a message-driven bean to serve. The default is 1.
“ior-security-config” on page 63	zero or one	Specifies the security information for the IOR.
“is-read-only-bean” on page 64	zero or one	Specifies that this entity bean is read-only.
“refresh-period-in-seconds” on page 94	zero or one	Specifies the rate at which a read-only-bean must be refreshed from the data source.
“commit-option” on page 45	zero or one	Has valid values of B or C. Default value is B.
“cmt-timeout-in-seconds” on page 45	zero or one	Overrides the Transaction Timeout setting of the Transaction Service for an individual bean.
“use-thread-pool-id” on page 118	zero or one	Specifies the thread pool from which threads are selected for remote invocations of this bean.
“gen-classes” on page 61	zero or one	Specifies all the generated class names for a bean.
“bean-pool” on page 36	zero or one	Specifies the bean pool properties. Used for stateless session beans, entity beans, and message-driven beans.
“bean-cache” on page 35	zero or one	Specifies the bean cache properties. Used only for stateful session beans and entity beans.
“mdb-resource-adapter” on page 77	zero or one	Specifies runtime configuration information for a message-driven bean.
“webservice-endpoint” on page 120	zero or more	Specifies information about a web service endpoint.
“flush-at-end-of-method” on page 60	zero or one	Specifies the methods that force a database flush after execution. Used for entity beans.

TABLE A-29 `ejb` Subelements (Continued)

Element	Required	Description
<a href="#">“checkpointed-methods” on page 42</a>	zero or one	Deprecated. Supported for backward compatibility. Use <a href="#">“checkpoint-at-end-of-method” on page 42</a> instead.
<a href="#">“checkpoint-at-end-of-method” on page 42</a>	zero or one	Specifies that the stateful session bean state is checkpointed, or persisted, after the specified methods are executed. The <code>availability-enabled</code> attribute must be set to <code>true</code> .

## Attributes

The following table describes attributes for the `ejb` element.

TABLE A-30 `ejb` Attributes

Attribute	Default	Description
<code>availability-enabled</code>	<code>false</code>	(optional) If set to <code>true</code> , and if availability is enabled in the EJB container, high-availability features apply to this bean if it is a stateful session bean.

## Example

```

<ejb>
  <ejb-name>CustomerEJB</ejb-name>
  <jndi-name>customer</jndi-name>
  <resource-ref>
    <res-ref-name>jdbc/SimpleBank</res-ref-name>
    <jndi-name>jdbc/__default</jndi-name>
  </resource-ref>
  <is-read-only-bean>false</is-read-only-bean>
  <commit-option>B</commit-option>
  <bean-pool>
    <steady-pool-size>10</steady-pool-size>
    <resize-quantity>10</resize-quantity>
    <max-pool-size>100</max-pool-size>
    <pool-idle-timeout-in-seconds>600</pool-idle-timeout-in-seconds>
  </bean-pool>
  <bean-cache>
    <max-cache-size>100</max-cache-size>
    <resize-quantity>10</resize-quantity>
    <removal-timeout-in-seconds>3600</removal-timeout-in-seconds>
    <victim-selection-policy>LRU</victim-selection-policy>
  </bean-cache>
</ejb>

```

## ejb-name

In the `sun-ejb-jar.xml` file, matches the `ejb-name` in the corresponding `ejb-jar.xml` file. The name must be unique among the names of the enterprise beans in the same EJB JAR file.

There is no architected relationship between the `ejb-name` in the deployment descriptor and the JNDI name that the deployer assigns to the EJB component's home.

In the `sun-cmp-mappings.xml` file, specifies the `ejb-name` of the entity bean in the `ejb-jar.xml` file to which the container-managed persistence (CMP) bean corresponds.

## Superelements

[“ejb” on page 53](#), [“method” on page 81](#) (`sun-ejb-jar.xml`)

## Subelements

none - contains data

## ejb-ref

Maps the `ejb-ref-name` in the corresponding Java EE deployment descriptor file `ejb-ref` entry to the absolute `jndi-name` of a resource.

The `ejb-ref` element is used for the declaration of a reference to an EJB's home. Applies to session beans or entity beans.

## Superelements

[“sun-web-app” on page 111](#) (`sun-web.xml`), [“ejb” on page 53](#) (`sun-ejb-jar.xml`)

## Subelements

The following table describes subelements for the `ejb-ref` element.

TABLE A-31 `ejb-ref` Subelements

Element	Required	Description
<a href="#">“ejb-ref-name” on page 57</a>	only one	Specifies the <code>ejb-ref-name</code> in the corresponding Java EE deployment descriptor file <code>ejb-ref</code> entry.
<a href="#">“jndi-name” on page 65</a>	only one	Specifies the absolute <code>jndi-name</code> of a resource.



---

## ejb-ref-name

Specifies the `ejb-ref-name` in the corresponding Java EE deployment descriptor file `ejb-ref` entry.

### Superelements

[“ejb-ref” on page 56](#) (`sun-web.xml`, `sun-ejb-jar.xml`)

### Subelements

none - contains data

## endpoint-address-uri

Specifies the relative path combined with the web server root to form the fully qualified endpoint address for a web service endpoint. This is a required element for EJB endpoints and an optional element for servlet endpoints.

For servlet endpoints, this value is relative to the web application context root. For EJB endpoints, the URI is relative to root of the web server (the first portion of the URI is a context root). The context root portion must not conflict with the context root of any web application deployed to the same web server.

In all cases, this value must be a fixed pattern (no `*` allowed).

If the web service endpoint is a servlet that implements only a single endpoint and has only one `url-pattern`, it is not necessary to set this value, because the web container derives it from the `web.xml` file.

### Superelements

[“webservice-endpoint” on page 120](#) (`sun-web.xml`, `sun-ejb-jar.xml`)

### Subelements

none - contains data

### Example

If the web server is listening at `http://localhost:8080`, the following `endpoint-address-uri`:

```
<endpoint-address-uri>StockQuoteService/StockQuotePort</endpoint-address-uri>
```

results in the following target endpoint address:

<http://localhost:8080/StockQuoteService/StockQuotePort>

## enterprise-beans

Specifies all the runtime properties for an EJB JAR file in the application.

### Superelements

“sun-ejb-jar” on page 111 ([sun-ejb-jar.xml](#))

### Subelements

The following table describes subelements for the `enterprise-beans` element.

TABLE A-32 enterprise-beans Subelements

Element	Required	Description
“name” on page 83	zero or one	Specifies the name string.
“unique-id” on page 117	zero or one	Specifies a unique system identifier. This data is automatically generated and updated at deployment/redeployment. Do not specify or edit this value.
“ejb” on page 53	zero or more	Defines runtime properties for a single enterprise bean within the application.
“pm-descriptors” on page 87	zero or one	Deprecated.
“cmp-resource” on page 44	zero or one	Specifies the database to be used for storing container-managed persistence (CMP) beans in an EJB JAR file.
“message-destination” on page 78	zero or more	Specifies the name of a logical message destination.
“webservice-description” on page 119	zero or more	Specifies a name and optional publish location for a web service.

### Example

```
<enterprise-beans>
  <ejb>
    <ejb-name>CustomerEJB</ejb-name>
    <jndi-name>customer</jndi-name>
    <resource-ref>
      <res-ref-name>jdbc/SimpleBank</res-ref-name>
      <jndi-name>jdbc/__default</jndi-name>
    </resource-ref>
    <is-read-only-bean>false</is-read-only-bean>
    <commit-option>B</commit-option>
    <bean-pool>
```

```

    <steady-pool-size>10</steady-pool-size>
    <resize-quantity>10</resize-quantity>
    <max-pool-size>100</max-pool-size>
    <pool-idle-timeout-in-seconds>600</pool-idle-timeout-in-seconds>
  </bean-pool>
  <bean-cache>
    <max-cache-size>100</max-cache-size>
    <resize-quantity>10</resize-quantity>
    <removal-timeout-in-seconds>3600</removal-timeout-in-seconds>
    <victim-selection-policy>LRU</victim-selection-policy>
  </bean-cache>
</ejb>
</enterprise-beans>

```

## establish-trust-in-client

Specifies if the target is capable of authenticating a client. The values are NONE, SUPPORTED, or REQUIRED.

### Superelements

[“transport-config” on page 116](#) (sun-ejb-jar.xml)

### Subelements

none - contains data

## establish-trust-in-target

Specifies if the target is capable of authenticating *to* a client. The values are NONE, SUPPORTED, or REQUIRED.

### Superelements

[“transport-config” on page 116](#) (sun-ejb-jar.xml)

### Subelements

none - contains data

# F

## finder

Describes the finders for CMP 1.1 with a method name and query.

### Superelements

“one-one-finders” on page 84 (sun-ejb-jar.xml)

### Subelements

The following table describes subelements for the `finder` element.

TABLE A-33 `finder` Subelements

Element	Required	Description
“method-name” on page 82	only one	Specifies the method name for the finder.
“query-params” on page 92	zero or one	Specifies the query parameters for the CMP 1.1 finder.
“query-filter” on page 91	zero or one	Specifies the query filter for the CMP 1.1 finder.
“query-variables” on page 93	zero or one	Specifies variables in query expression for the CMP 1.1 finder.
“query-ordering” on page 92	zero or one	Specifies the query ordering for the CMP 1.1 finder.

## flush-at-end-of-method

Specifies the methods that force a database flush after execution. Applicable to entity beans.

### Superelements

“ejb” on page 53 (sun-ejb-jar.xml)

### Subelements

The following table describes subelements for the `flush-at-end-of-method` element.

TABLE A-34 `flush-at-end-of-method` Subelements

Element	Required	Description
“method” on page 81	one or more	Specifies a bean method.

## G

## gen-classes

Specifies all the generated class names for a bean.

---

**Note** – This value is automatically generated by the server at deployment or redeployment time. Do not specify it or change it after deployment.

---

### Superelements

“[ejb](#)” on page 53 (`sun-ejb-jar.xml`)

### Subelements

The following table describes subelements for the `gen-class` element.

TABLE A-35 `gen-classes` Subelements

Element	Required	Description
“ <a href="#">remote-impl</a> ” on page 95	zero or one	Specifies the fully-qualified class name of the generated <code>EJBObject</code> impl class.
“ <a href="#">local-impl</a> ” on page 70	zero or one	Specifies the fully-qualified class name of the generated <code>EJBLocalObject</code> impl class.
“ <a href="#">remote-home-impl</a> ” on page 95	zero or one	Specifies the fully-qualified class name of the generated <code>EJBHome</code> impl class.
“ <a href="#">local-home-impl</a> ” on page 70	zero or one	Specifies the fully-qualified class name of the generated <code>EJBLocalHome</code> impl class.

## group-name

Specifies a group name in the current realm.

### Superelements

“[security-role-mapping](#)” on page 102 (`sun-web.xml`, `sun-ejb-jar.xml`)

### Subelements

none - contains data

## H

## http-method

Specifies an HTTP method that is eligible for caching. The default is GET.

### Superelements

“[cache-mapping](#)” on page 40 (`sun-web.xml`)

### Subelements

none - contains data

## I

## idempotent-url-pattern

Specifies a URL pattern for idempotent requests.

### Superelements

“[sun-web-app](#)” on page 111 (`sun-web.xml`)

### Subelements

none

### Attributes

The following table describes attributes for the `idempotent-url-pattern` element.

TABLE A-36 `idempotent-url-pattern` Attributes

Attribute	Default	Description
<code>url-pattern</code>	none	Specifies a URL pattern, which can contain wildcards. The URL pattern must conform to the mappings specified in section SRV 11.2 of the Servlet 2.4 specification.
<code>no-of-retries</code>	-1	(optional) Specifies the number of times the load balancer retries an idempotent request. A value of -1 indicates infinite retries.

## Example

The following example specifies that all requests for the URI `sun-java/*` are idempotent.

```
<idempotent-url-pattern url-pattern="sun_java/*" no-of-retries="10"/>
```

## integrity

Specifies if the target supports integrity-protected messages. The values are NONE, SUPPORTED, or REQUIRED.

## Superelements

[“transport-config” on page 116](#) (`sun-ejb-jar.xml`)

## Subelements

none - contains data

## ior-security-config

Specifies the security information for the input-output redirection (IOR).

## Superelements

[“ejb” on page 53](#) (`sun-ejb-jar.xml`)

## Subelements

The following table describes subelements for the `ior-security-config` element.

TABLE A-37 `ior-security-config` Subelements

Element	Required	Description
<a href="#">“transport-config” on page 116</a>	zero or one	Specifies the security information for transport.
<a href="#">“as-context” on page 34</a>	zero or one	Specifies the authentication mechanism used to authenticate the client. If specified, it is USERNAME_PASSWORD.
<a href="#">“sas-context” on page 100</a>	zero or one	Describes the sas-context fields.

## is-cache-overflow-allowed

This element is deprecated. Do not use.

## Superelements

[“bean-cache” on page 35](#) (sun-ejb-jar.xml)

## is-one-one-cmp

This element is not used.

## Superelements

[“cmp” on page 44](#) (sun-ejb-jar.xml)

## is-read-only-bean

Specifies that this entity bean is a read-only bean if `true`. If this element is absent, the default value of `false` is used.

## Superelements

[“ejb” on page 53](#) (sun-ejb-jar.xml)

## Subelements

none - contains data

## J

## java-method

Specifies a method.

## Superelements

[“message” on page 77](#) (sun-web.xml, sun-ejb-jar.xml)

## Subelements

The following table describes subelements for the `java-method` element.



TABLE A-38 java-method Subelements

Element	Required	Description
<a href="#">“method-name” on page 82</a>	only one	Specifies a method name.
<a href="#">“method-params” on page 82</a>	zero or one	Specifies fully qualified Java type names of method parameters.

## jms-durable-subscription-name

Specifies the durable subscription associated with a message-driven bean class. Only applies to the Java Message Service Topic Destination type, and only when the message-driven bean deployment descriptor subscription durability is Durable.

### Superelements

[“ejb” on page 53](#) (sun-ejb-jar.xml)

### Subelements

none - contains data

## jms-max-messages-load

Specifies the maximum number of messages to load into a Java Message Service session at one time for a message-driven bean to serve. The default is 1.

### Superelements

[“ejb” on page 53](#) (sun-ejb-jar.xml)

### Subelements

none - contains data

## jndi-name

Specifies the absolute jndi - name of a URL resource or a resource.

For entity beans and session beans, this value specifies the global JNDI name of the EJBHome object. It is only needed if the entity or session bean exposes a remote view.

## Superelements

“[ejb-ref](#)” on page 56, “[message-destination](#)” on page 78, “[resource-env-ref](#)” on page 98, “[resource-ref](#)” on page 99 ([sun-web.xml](#), [sun-ejb-jar.xml](#)); “[cmp-resource](#)” on page 44, “[ejb](#)” on page 53, “[mdb-connection-factory](#)” on page 76 ([sun-ejb-jar.xml](#))

## Subelements

none - contains data

## jsp-config

Specifies JSP configuration information.

## Superelements

“[sun-web-app](#)” on page 111 ([sun-web.xml](#))

## Subelements

The following table describes subelements for the `jsp-config` element.

TABLE A-39 `jsp-config` Subelements

Element	Required	Description
“ <a href="#">property (with attributes)</a> ” on page 90	zero or more	Specifies a property, which has a name and a value.

## Properties

The default property values are tuned for development of JSP files at the cost of performance. To maximize performance, set `jsp-config` properties to these non-default values:

- `development` - `false` (as an alternative, set to `true` and give `modificationTestInterval` a large value)
- `mappedfile` - `false`
- `trimSpaces` - `true`
- `suppressSmap` - `true`
- `fork` - `false` (on Solaris)
- `classdebuginfo` - `false`

The following table describes properties for the `jsp-config` element.

TABLE A-40 jsp-config Properties

Property	Default	Description
checkInterval	0	If development is set to false and checkInterval is greater than zero, background compilations are enabled. The checkInterval is the time in seconds between checks to see if a JSP file needs to be recompiled.
classdebuginfo	true	Specifies whether the generated Java servlets are compiled with the debug option set (-g for javac).
classpath	created dynamically based on the current web application	Specifies the classpath to use when compiling generated servlets.
compiler	javac	Specifies the compiler Ant uses to compile JSP files. See the Ant documentation for more information: <a href="http://antinstaller.sourceforge.net/manual/manual/">http://antinstaller.sourceforge.net/manual/manual/</a>
compilerSourceVM	Depends on the Application Server's Java runtime	Specifies the JDK release with which source compatibility of the generated servlets is provided. Same as the -source release option of javac. For more information, see <a href="http://java.sun.com/j2se/1.5.0/docs/tooldocs/solaris/javac.html#options">http://java.sun.com/j2se/1.5.0/docs/tooldocs/solaris/javac.html#options</a> .
compilerTargetVM	Depends on the Application Server's Java runtime	Specifies the JVM version for which the servlet class files are generated. Same as the -target release option of javac. For more information, see <a href="http://java.sun.com/j2se/1.5.0/docs/tooldocs/solaris/javac.html#options">http://java.sun.com/j2se/1.5.0/docs/tooldocs/solaris/javac.html#options</a> .
defaultBufferNone	false	If true, the default for the buffer attribute of the page directive is none.
development	true	If set to true, enables development mode, which allows JSP files to be checked for modification. Specify the frequency at which JSPs are checked using the modificationTestInterval property.
dumpSmap	false	If set to true, dumps SMAP information for JSR 45 debugging to a file. Set to false if suppressSmap is true.
enablePooling	true	If set to true, tag handler pooling is enabled.
enableTldValidation	false	If set to true, all Tag Library Descriptor (TLD) files referenced by the web application are validated against their underlying schema or DTD file.
errorOnUseBeanInvalidClassAttribute	false	If set to true, issues an error when the value of the class attribute in a useBean action is not a valid bean class.
fork	true	Specifies that Ant forks the compiling of JSP files, using a JVM separate from the one in which Tomcat is running.

TABLE A-40 jsp-config Properties (Continued)

Property	Default	Description
genStrAsByteArray	true	If true, text strings are generated as bytes (encoded with the page encoding), if the page is not buffered.
genStrAsCharArray	false	If set to true, generates text strings as char arrays, which improves performance in some cases.
httpMethods	* for all methods	Specifies a comma separated list of HTTP methods supported by the JspServlet.
ieClassId	clsid:8AD9C840-044E-11D1-B3E9-00805F499D93	Specifies the Java plug-in COM class ID for Internet Explorer. Used by the <jsp:plugin> tags.
ignoreJspFragmentErrors	false	If set to true, instructs the compiler to ignore any JSP precompilation errors pertaining to statically included JSP segments that, despite not being top level JSP files, use the .jsp or .jspx extension (instead of the recommended .jspxf).
initialCapacity	32	Specifies the initial capacity of the HashMap that maps JSP files to their corresponding servlets.
javaEncoding	UTF8	Specifies the encoding for the generated Java servlet. This encoding is passed to the Java compiler that is used to compile the servlet as well. By default, the web container tries to use UTF8. If that fails, it tries to use the javaEncoding value.  For encodings, see: <a href="http://java.sun.com/j2se/1.5.0/docs/guide/intl/encoding.doc.html">http://java.sun.com/j2se/1.5.0/docs/guide/intl/encoding.doc.html</a>
keepgenerated	true with JDK 5 and before and for jspc, otherwise false	If set to true, keeps the generated Java files. If false, deletes the Java files.
mappedfile	true	If set to true, generates static content with one print statement per input line, to ease debugging.
modificationTestInterval	0	Specifies the frequency in seconds at which JSPs are checked for modification. A value of 0 causes the JSP to be checked on every access. Used only if development is set to true.
reload-interval	0	Specifies the frequency in seconds at which JSP files are checked for modifications. Setting this value to 0 checks JSP files for modifications on every request. Setting this value to -1 disables checks for JSP modifications and JSP recompilation.
saveBytecode	true for jspc, otherwise false	If true, generated byte code is saved to .class files? This option is meaningful only when the Java compiler API, JSR 199 (available with and used as the default on Java 6) is used for javac compilations.
scratchdir	The default work directory for the web application	Specifies the working directory created for storing all the generated code.

TABLE A-40 jsp-config Properties (Continued)

Property	Default	Description
suppressSmap	false	If set to true, generation of SMAP information for JSR 45 debugging is suppressed.
trimSpaces	false	If set to true, trims white spaces in template text between actions or directives.
usePrecompiled	false	If set to true, an accessed JSP file is not compiled. Its precompiled servlet class is used instead.  It is assumed that JSP files have been precompiled, and their corresponding servlet classes have been bundled in the web application's WEB-INF/lib or WEB-INF/classes directory.
xpoweredBy	true	If set to true, the X-Powered-By response header is added by the generated servlet.

## K

### key-field

Specifies a component of the key used to look up and extract cache entries. The web container looks for the named parameter, or field, in the specified scope.

If this element is not present, the web container uses the Servlet Path (the path section that corresponds to the servlet mapping that activated the current request). See the Servlet 2.4 specification, section SRV 4.4, for details on the Servlet Path.

### Superelements

[“cache-mapping” on page 40](#) (sun-web.xml)

### Subelements

none

### Attributes

The following table describes attributes for the key-field element.

TABLE A-41 key-field Attributes

Attribute	Default	Description
name	none	Specifies the input parameter name.
scope	request.parameter	(optional) Specifies the scope from which the input parameter is retrieved. Allowed values are context.attribute, request.header, request.parameter, request.cookie, session.id, and session.attribute.

## L

## local-home-impl

Specifies the fully-qualified class name of the generated `EJBLocalHome impl` class.

---

**Note** – This value is automatically generated by the server at deployment or redeployment time. Do not specify it or change it after deployment.

---

### Superelements

[“gen-classes” on page 61](#) (sun-ejb-jar.xml)

### Subelements

none - contains data

## local-impl

Specifies the fully-qualified class name of the generated `EJBLocalObject impl` class.

---

**Note** – This value is automatically generated by the server at deployment or redeployment time. Do not specify it or change it after deployment.

---

### Superelements

[“gen-classes” on page 61](#) (sun-ejb-jar.xml)

### Subelements

none - contains data

## locale-charset-info

Deprecated. For backward compatibility only. Use the “parameter-encoding” on page 84 subelement of “sun-web-app” on page 111 instead. Specifies information about the application’s internationalization settings.

### Superelements

“sun-web-app” on page 111 (sun-web.xml)

### Subelements

The following table describes subelements for the locale-charset-info element.

TABLE A-42 locale-charset-info Subelements

Element	Required	Description
“locale-charset-map” on page 71	one or more	Maps a locale and an agent to a character encoding. Provided for backward compatibility. Used only for request processing, and only if no parameter-encoding is defined.
“parameter-encoding” on page 84	zero or one	Determines the default request character encoding and how the web container decodes parameters from forms according to a hidden field value.

### Attributes

The following table describes attributes for the locale-charset-info element.

TABLE A-43 locale-charset-info Attributes

Attribute	Default	Description
default-locale	none	Although a value is required, the value is ignored. Use the default-charset attribute of the “parameter-encoding” on page 84 element.

## locale-charset-map

Maps locales and agents to character encodings. Provided for backward compatibility. Used only for request processing. Used only if the character encoding is not specified in the request and cannot be derived from the optional “parameter-encoding” on page 84 element. For encodings, see <http://java.sun.com/j2se/1.5.0/docs/guide/intl/encoding.doc.html>.

### Superelements

“locale-charset-info” on page 71 (sun-web.xml)

## Subelements

The following table describes subelements for the `locale-charset-map` element.

TABLE A-44 `locale-charset-map` Subelements

Element	Required	Description
<a href="#">“description” on page 52</a>	zero or one	Specifies an optional text description of a mapping.

## Attributes

The following table describes attributes for the `locale-charset-map` element.

TABLE A-45 `locale-charset-map` Attributes

Attribute	Default	Description
<code>locale</code>	none	Specifies the locale name.
<code>agent</code>	none	(optional) Specifies the type of client that interacts with the application server. For a given locale, different agents can have different preferred character encodings. The value of this attribute must exactly match the value of the <code>user-agent</code> HTTP request header sent by the client. See <a href="#">Table A-46</a> for more information.
<code>charset</code>	none	Specifies the character encoding to which the locale maps.

## Example Agents

The following table specifies example agent attribute values.

TABLE A-46 Example agent Attribute Values

Agent	user-agent Header and agent Attribute Value
Internet Explorer 5.00 for Windows 2000	Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)
Netscape 4.7.7 for Windows 2000	Mozilla/4.77 [en] (Windows NT 5.0; U)
Netscape 4.7 for Solaris	Mozilla/4.7 [en] (X11; u; Sun OS 5.6 sun4u)

## localpart

Specifies the local part of a QName.

## Superelements

[“service-qname” on page 104](#), [“wsdl-port” on page 121](#) (`sun-web.xml`, `sun-ejb-jar.xml`)



## Subelements

none - contains data

## login-config

Specifies the authentication configuration for an EJB web service endpoint. Not needed for servlet web service endpoints. A servlet's security configuration is contained in the `web.xml` file.

## Superelements

[“webservice-endpoint” on page 120](#) (`sun-web.xml`, `sun-ejb-jar.xml`)

## Subelements

The following table describes subelements for the `login-config` element.

TABLE A-47 `login-config` subelements

Element	Required	Description
<a href="#">“auth-method” on page 34</a>	only one	Specifies the authentication method.
<a href="#">“realm” on page 93</a>	zero or one	Specifies the name of the realm used to process all authentication requests.

## M

## manager-properties

Specifies session manager properties.

## Superelements

[“session-manager” on page 107](#) (`sun-web.xml`)

## Subelements

The following table describes subelements for the `manager-properties` element.

TABLE A-48 manager-properties Subelements

Element	Required	Description
<a href="#">“property (with attributes)” on page 90</a>	zero or more	Specifies a property, which has a name and a value.

## Properties

The following table describes properties for the `manager-properties` element.

TABLE A-49 manager-properties Properties

Property	Default	Description
<code>reapIntervalSeconds</code>	60	<p>Specifies the number of seconds between checks for expired sessions. This is also the interval at which sessions are passivated if <code>maxSessions</code> is exceeded.</p> <p>If <code>persistenceFrequency</code> is set to <code>time-based</code>, active sessions are stored at this interval.</p> <p>To prevent data inconsistency, set this value lower than the frequency at which session data changes. For example, this value should be as low as possible (1 second) for a hit counter servlet on a frequently accessed web site, or the last few hits might be lost each time the server is restarted.</p> <p>Applicable only if the <code>persistence-type</code> attribute of the parent <a href="#">“session-manager” on page 107</a> element is <code>file</code> or <code>replicated</code>.</p>
<code>maxSessions</code>	-1	<p>Specifies the maximum number of sessions that are permitted in the cache, or -1 for no limit. After this, an attempt to create a new session causes an <code>IllegalStateException</code> to be thrown.</p> <p>If the <code>persistence-type</code> attribute of the parent <a href="#">“session-manager” on page 107</a> element is <code>file</code> or <code>replicated</code>, the session manager passivates sessions to the persistent store when this maximum is reached.</p>
<code>sessionFilename</code>	One of the following: <i>domain-dir/generated/jsp/            module-name/context-path            _SESSIONS.ser</i>  <i>domain-dir/generated/jsp/            app-name/module-name/            context-path_SESSIONS.ser</i>	<p>Specifies the absolute or relative path to the directory in which the session state is preserved between application restarts, if preserving the state is possible. A relative path is relative to the temporary directory for this web application. To disable preservation of the session state, set this property's value to an empty string.</p> <p>Applicable only if the <code>persistence-type</code> attribute of the parent <a href="#">“session-manager” on page 107</a> element is <code>memory</code>.</p>

TABLE A-49 manager-properties Properties (Continued)

Property	Default	Description
persistenceFrequency	web-method	<p>Specifies how often the session state is stored. Allowed values are as follows:</p> <ul style="list-style-type: none"> <li>■ <code>web-method</code> - The session state is stored at the end of each web request prior to sending a response back to the client. This mode provides the best guarantee that the session state is fully updated in case of failure.</li> <li>■ <code>time-based</code> - The session state is stored in the background at the frequency set by <code>reapIntervalSeconds</code>. This mode provides less of a guarantee that the session state is fully updated. However, it can provide a significant performance improvement because the state is not stored after each request.</li> </ul> <p>Applicable only if the <code>persistence-type</code> attribute of the parent <code>session-manager</code> on page 107 element is <code>replicated</code>.</p>

## mapping-properties

This element is not implemented.

### Superelements

[“cmp” on page 44](#) (`sun-ejb-jar.xml`)

## max-cache-size

Specifies the maximum number of beans allowable in cache. A value of zero indicates an unbounded cache. In reality, there is no hard limit. The `max-cache-size` limit is just a hint to the cache implementation. Default is 512.

Applies to stateful session beans and entity beans.

### Superelements

[“bean-cache” on page 35](#) (`sun-ejb-jar.xml`)

### Subelements

none - contains data

## max-pool-size

Specifies the maximum number of bean instances in the pool. Values are from 0 (1 for message-driven bean) to MAX\_INTEGER. A value of 0 means the pool is unbounded. Default is 64.

Applies to all beans.

### Superelements

[“bean-pool” on page 36](#) (sun-ejb-jar.xml)

### Subelements

none - contains data

## max-wait-time-in-millis

This element is deprecated. Do not use.

### Superelements

[“bean-pool” on page 36](#) (sun-ejb-jar.xml)

## mdb-connection-factory

Specifies the connection factory associated with a message-driven bean. Queue or Topic type must be consistent with the Java Message Service Destination type associated with the message-driven bean class.

### Superelements

[“ejb” on page 53](#) (sun-ejb-jar.xml)

### Subelements

The following table describes subelements for the `mdb-connection-factory` element.

TABLE A-50 `mdb-connection-factory` Subelements

Element	Required	Description
<a href="#">“jndi-name” on page 65</a>	only one	Specifies the absolute jndi-name.

TABLE A-50 mdb-connection-factory Subelements (Continued)

Element	Required	Description
<a href="#">“default-resource-principal” on page 51</a>	zero or one	Specifies the default sign-on (name/password) to the resource manager.

## mdb-resource-adapter

Specifies runtime configuration information for a message-driven bean.

### Superelements

[“ejb” on page 53](#) (sun-ejb-jar.xml)

### Subelements

The following table describes subelements for the mdb-resource-adapter element.

TABLE A-51 mdb-resource-adapter subelements

Element	Required	Description
<a href="#">“resource-adapter-mid” on page 97</a>	zero or one	Specifies a resource adapter module ID.
<a href="#">“activation-config” on page 32</a>	one or more	Specifies an activation configuration.

## message

Specifies the methods or operations to which message security requirements apply.

### Superelements

[“message-security” on page 79](#) (sun-web.xml, sun-ejb-jar.xml)

### Subelements

The following table describes subelements for the message element.

TABLE A-52 message Subelements

Element	Required	Description
<a href="#">“java-method” on page 64</a>	zero or one	Specifies the methods or operations to which message security requirements apply.

TABLE A-52 message Subelements *(Continued)*

Element	Required	Description
<a href="#">“operation-name” on page 84</a>	zero or one	Specifies the WSDL name of an operation of a web service.

## message-destination

Specifies the name of a logical message-destination defined within an application. The message-destination-name matches the corresponding message-destination-name in the corresponding Java EE deployment descriptor file. Use when the message destination reference in the corresponding Java EE deployment descriptor file specifies a message-destination-link to a logical message-destination.

### Superelements

[“sun-web-app” on page 111](#) (sun-web.xml), [“enterprise-beans” on page 58](#) (sun-ejb-jar.xml)

### Subelements

The following table describes subelements for the message-destination element.

TABLE A-53 message-destination subelements

Element	Required	Description
<a href="#">“message-destination-name” on page 78</a>	only one	Specifies the name of a logical message destination defined within the corresponding Java EE deployment descriptor file.
<a href="#">“jndi-name” on page 65</a>	only one	Specifies the jndi-name of the associated entity.

## message-destination-name

Specifies the name of a logical message destination defined within the corresponding Java EE deployment descriptor file.

### Superelements

[“message-destination” on page 78](#) (sun-web.xml, sun-ejb-jar.xml)

### Subelements

none - contains data

## message-destination-ref

Directly binds a message destination reference to the JNDI name of a `Queue`, `Topic`, or other physical destination. Use only when the message destination reference in the corresponding Java EE deployment descriptor file does *not* specify a `message-destination-link` to a logical `message-destination`.

### Superelements

“[sun-web-app](#)” on page 111 (`sun-web.xml`), “[ejb](#)” on page 53 (`sun-ejb-jar.xml`)

### Subelements

The following table describes subelements for the `message-destination-ref` element.

TABLE A-54 message-destination-ref subelements

Element	Required	Description
“ <a href="#">message-destination-ref-name</a> ” on page 79	only one	Specifies the name of a physical message destination defined within the corresponding Java EE deployment descriptor file.
“ <a href="#">jndi-name</a> ” on page 65	only one	Specifies the <code>jndi-name</code> of the associated entity.

## message-destination-ref-name

Specifies the name of a physical message destination defined within the corresponding Java EE deployment descriptor file.

### Superelements

“[message-destination-ref](#)” on page 79 (`sun-web.xml`, `sun-ejb-jar.xml`)

### Subelements

none - contains data

## message-security

Specifies message security requirements.

- If the grandparent element is “[webservice-endpoint](#)” on page 120, these requirements pertain to request and response messages of the endpoint.
- If the grandparent element is “[port-info](#)” on page 88, these requirements pertain to the port of the referenced service.

## Superelements

“[message-security-binding](#)” on page 80 (sun-web.xml, sun-ejb-jar.xml)

## Subelements

The following table describes subelements for the message-security element.

TABLE A-55 message-security Subelements

Element	Required	Description
“ <a href="#">message</a> ” on page 77	one or more	Specifies the methods or operations to which message security requirements apply.
“ <a href="#">request-protection</a> ” on page 95	zero or one	Defines the authentication policy requirements of the application’s request processing.
“ <a href="#">response-protection</a> ” on page 99	zero or one	Defines the authentication policy requirements of the application’s response processing.

## message-security-binding

Specifies a custom authentication provider binding for a parent “[webservice-endpoint](#)” on page 120 or “[port-info](#)” on page 88 element in one or both of these ways:

- By binding to a specific provider
- By specifying the message security requirements enforced by the provider

## Superelements

“[webservice-endpoint](#)” on page 120, “[port-info](#)” on page 88 (sun-web.xml, sun-ejb-jar.xml)

## Subelements

The following table describes subelements for the message-security-binding element.

TABLE A-56 message-security-binding Subelements

Element	Required	Description
“ <a href="#">message-security</a> ” on page 79	zero or more	Specifies message security requirements.

## Attributes

The following table describes attributes for the message-security-binding element.



TABLE A-57 message-security-binding Attributes

Attribute	Default	Description
auth-layer	none	Specifies the message layer at which authentication is performed. The value must be SOAP.
provider-id	none	(optional) Specifies the authentication provider used to satisfy application-specific message security requirements.  If this attribute is not specified, a default provider is used, if it is defined for the message layer.  if no default provider is defined, authentication requirements defined in the message-security-binding are not enforced.

## method

Specifies a bean method.

### Superelements

“flush-at-end-of-method” on page 60 (sun-ejb-jar.xml)

### Subelements

The following table describes subelements for the method element.

TABLE A-58 method Subelements

Element	Required	Description
“description” on page 52	zero or one	Specifies an optional text description.
“ejb-name” on page 56	zero or one	Matches the ejb-name in the corresponding ejb-jar.xml file.
“method-name” on page 82	only one	Specifies a method name.
“method-intf” on page 81	zero or one	Specifies the method interface to distinguish between methods with the same name in different interfaces.
“method-params” on page 82	zero or one	Specifies fully qualified Java type names of method parameters.

## method-intf

Specifies the method interface to distinguish between methods with the same name in different interfaces. Allowed values are Home, Remote, LocalHome, and Local.

### Superelements

“method” on page 81 (sun-ejb-jar.xml)

## Subelements

none - contains data

## method-name

Specifies a method name or \* (an asterisk) for all methods. If a method is overloaded, specifies all methods with the same name.

## Superelements

[“java-method” on page 64](#) (sun-web.xml, sun-ejb-jar.xml); [“finder” on page 60](#), [“query-method” on page 92](#), [“method” on page 81](#) (sun-ejb-jar.xml)

## Subelements

none - contains data

## Examples

```
<method-name>findTeammates</method-name>
```

```
<method-name>*</method-name>
```

## method-param

Specifies the fully qualified Java type name of a method parameter.

## Superelements

[“method-params” on page 82](#) (sun-web.xml, sun-ejb-jar.xml)

## Subelements

none - contains data

## method-params

Specifies fully qualified Java type names of method parameters.

## Superelements

[“java-method” on page 64](#) (sun-web.xml, sun-ejb-jar.xml); [“query-method” on page 92](#), [“method” on page 81](#) (sun-ejb-jar.xml)

## Subelements

The following table describes subelements for the `method-params` element.

TABLE A-59 `method-params` Subelements

Element	Required	Description
<a href="#">“method-param” on page 82</a>	zero or more	Specifies the fully qualified Java type name of a method parameter.

## N

### name

Specifies the name of the entity.

### Superelements

[“call-property” on page 41](#), [“default-resource-principal” on page 51](#), [“stub-property” on page 110](#) (`sun-web.xml`, `sun-ejb-jar.xml`); [“enterprise-beans” on page 58](#), [“principal” on page 89](#), [“property \(with subelements\)” on page 91](#) (`sun-ejb-jar.xml`)

### Subelements

none - contains data

### namespaceURI

Specifies the namespace URI.

### Superelements

[“service-qname” on page 104](#), [“wsdl-port” on page 121](#) (`sun-web.xml`, `sun-ejb-jar.xml`)

### Subelements

none - contains data

## O

## one-one-finders

Describes the finders for CMP 1.1 beans.

### Superelements

“[cmp](#)” on page 44 ([sun-ejb-jar.xml](#))

### Subelements

The following table describes subelements for the one-one-finders element.

TABLE A-60 one-one-finders Subelements

Element	Required	Description
“ <a href="#">finder</a> ” on page 60	one or more	Describes the finders for CMP 1.1 with a method name and query.

## operation-name

Specifies the WSDL name of an operation of a web service.

### Superelements

“[message](#)” on page 77 ([sun-web.xml](#), [sun-ejb-jar.xml](#))

### Subelements

none - contains data

## P

## parameter-encoding

Specifies the default request character encoding and how the web container decodes parameters from forms according to a hidden field value.

If both the “[sun-web-app](#)” on page 111 and “[locale-charset-info](#)” on page 71 elements have parameter-encoding subelements, the subelement of sun-web-app takes precedence. For encodings, see <http://java.sun.com/j2se/1.5.0/docs/guide/intl/encoding.doc.html>.

## Superelements

“[locale-charset-info](#)” on page 71, “[sun-web-app](#)” on page 111 (`sun-web.xml`)

## Subelements

none

## Attributes

The following table describes attributes for the `parameter-encoding` element.

TABLE A-61 `parameter-encoding` Attributes

Attribute	Default	Description
<code>form-hint-field</code>	none	(optional) The name of the hidden field in the form. This field specifies the character encoding the web container uses for <code>request.getParameter</code> and <code>request.getReader</code> calls when the charset is not set in the request's <code>content-type</code> header.
<code>default-charset</code>	ISO-8859-1	(optional) The default request character encoding.

## pass-by-reference

Specifies the passing method used by a servlet or enterprise bean calling a remote interface method in another bean that is colocated within the same process.

- If `false` (the default if this element is not present), this application uses pass-by-value semantics.
- If `true`, this application uses pass-by-reference semantics.

---

**Note** – The `pass-by-reference` element only applies to remote calls. As defined in the EJB 2.1 specification, section 5.4, calls to local interfaces use pass-by-reference semantics.

If the `pass-by-reference` element is set to its default value of `false`, the passing semantics for calls to remote interfaces comply with the EJB 2.1 specification, section 5.4. If set to `true`, remote calls involve pass-by-reference semantics instead of pass-by-value semantics, contrary to this specification.

Portable programs cannot assume that a copy of the object is made during such a call, and thus that it's safe to modify the original. Nor can they assume that a copy is not made, and thus that changes to the object are visible to both caller and callee. When this element is set to `true`, parameters and return values should be considered read-only. The behavior of a program that modifies such parameters or return values is undefined.

---

When a servlet or enterprise bean calls a remote interface method in another bean that is colocated within the same process, by default the Application Server makes copies of all the call parameters in order to preserve the pass-by-value semantics. This increases the call overhead and decreases performance.

However, if the calling method does not change the object being passed as a parameter, it is safe to pass the object itself without making a copy of it. To do this, set the `pass-by-reference` value to `true`.

The setting of this element in the `sun-application.xml` file applies to all EJB modules in the application. For an individually deployed EJB module, you can set the same element in the `sun-ejb-jar.xml` file. If `pass-by-reference` is used at both the bean and application level, the bean level takes precedence.

## Superelements

[“ejb” on page 53](#) (`sun-ejb-jar.xml`)

## Subelements

`none` - contains data

## password

Specifies the password for the principal.

## Superelements

[“default-resource-principal” on page 51](#) (`sun-web.xml`, `sun-ejb-jar.xml`)

---

## Subelements

none - contains data

## pm-descriptors

This element and its subelements are deprecated. Do not use.

## Superelements

[“enterprise-beans” on page 58](#) (sun-ejb-jar.xml)

## pool-idle-timeout-in-seconds

Specifies the maximum time, in seconds, that a bean instance is allowed to remain idle in the pool. When this timeout expires, the bean instance in a pool becomes a candidate for passivation or deletion. This is a hint to the server. A value of 0 specifies that idle beans remain in the pool indefinitely. Default value is 600.

Applies to stateless session beans, entity beans, and message-driven beans.

---

**Note** – For a stateless session bean or a message-driven bean, the bean is removed (garbage collected) when the timeout expires.

---

## Superelements

[“bean-pool” on page 36](#) (sun-ejb-jar.xml)

## Subelements

none - contains data

## port-component-name

Specifies a unique name for a port component within a web or EJB module.

## Superelements

[“webservice-endpoint” on page 120](#) (sun-web.xml, sun-ejb-jar.xml)

## Subelements

none - contains data

## port-info

Specifies information for a port within a web service reference.

Either a `service-endpoint-interface` or a `wSDL-port` or both must be specified. If both are specified, `wSDL-port` specifies the port that the container chooses for container-managed port selection.

The same `wSDL-port` value must not appear in more than one `port-info` element within the same `service-ref`.

If a `service-endpoint-interface` is using container-managed port selection, its value must not appear in more than one `port-info` element within the same `service-ref`.

### Superelements

“`service-ref`” on page 104 (`sun-web.xml`, `sun-ejb-jar.xml`)

### Subelements

The following table describes subelements for the `port-info` element.

TABLE A-62 `port-info` subelements

Element	Required	Description
“ <code>service-endpoint-interface</code> ” on page 103	zero or one	Specifies the web service reference name relative to <code>java:comp/env</code> .
“ <code>wSDL-port</code> ” on page 121	zero or one	Specifies the WSDL port.
“ <code>stub-property</code> ” on page 110	zero or more	Specifies JAX-RPC property values that are set on a <code>javax.xml.rpc.Stub</code> object before it is returned to the web service client.
“ <code>call-property</code> ” on page 41	zero or more	Specifies JAX-RPC property values that are set on a <code>javax.xml.rpc.Call</code> object before it is returned to the web service client.
“ <code>message-security-binding</code> ” on page 80	zero or one	Specifies a custom authentication provider binding.

## prefetch-disabled

Disables prefetching of entity bean states for the specified query methods.

### Superelements

“`cmp`” on page 44 (`sun-ejb-jar.xml`)



## Subelements

The following table describes subelements for the `prefetch-disabled` element.

TABLE A-63 `prefetch-disabled` Subelements

Element	Required	Description
<a href="#">“query-method” on page 92</a>	one or more	Specifies a query method.

## principal

Defines a node that specifies a user name on the platform.

## Superelements

[“ejb” on page 53](#) (`sun-ejb-jar.xml`)

## Subelements

The following table describes subelements for the `principal` element.

TABLE A-64 `principal` Subelements

Element	Required	Description
<a href="#">“name” on page 83</a>	only one	Specifies the name of the user.

## principal-name

Contains the principal (user) name.

In an enterprise bean, specifies the principal (user) name that has the `run-as` role specified.

## Superelements

[“security-role-mapping” on page 102](#) (`sun-web.xml`, `sun-ejb-jar.xml`), [“servlet” on page 105](#) (`sun-web.xml`)

## Subelements

none - contains data

## Attributes

The following table describes attributes for the `principal-name` element.

TABLE A-65 principal-name Attributes

Attribute	Default	Description
class-name	com.sun.enterprise.deployment. PrincipalImpl	(optional) Specifies the custom principal implementation class corresponding to the named principal.

## property (with attributes)

Specifies the name and value of a property. A property adds configuration information to its parent element that is one or both of the following:

- Optional with respect to Application Server
- Needed by a system or object that Application Server doesn't have knowledge of, such as an LDAP server or a Java class

### Superelements

“cache” on page 37, “cache-helper” on page 39, “class-loader” on page 42, “cookie-properties” on page 48, “default-helper” on page 50, “manager-properties” on page 73, “session-properties” on page 107, “store-properties” on page 108, “sun-web-app” on page 111, “webservice-endpoint” on page 120 (sun-web.xml)

### Subelements

The following table describes subelements for the property element.

TABLE A-66 property Subelements

Element	Required	Description
“description” on page 52	zero or one	Specifies an optional text description of a property.

### Attributes

The following table describes attributes for the property element.

TABLE A-67 property Attributes

Attribute	Default	Description
name	none	Specifies the name of the property.
value	none	Specifies the value of the property.

### Example

```
<property name="reapIntervalSeconds" value="20" />
```

## property (with subelements)

Specifies the name and value of a property. A property adds configuration information to its parent element that is one or both of the following:

- Optional with respect to Application Server
- Needed by a system or object that Application Server doesn't have knowledge of, such as an LDAP server or a Java class

### Superelements

“[cmp-resource](#)” on page 44, “[schema-generator-properties](#)” on page 101, “[webservice-endpoint](#)” on page 120 (`sun-ejb-jar.xml`)

### Subelements

The following table describes subelements for the property element.

TABLE A-68 property subelements

Element	Required	Description
“ <a href="#">name</a> ” on page 83	only one	Specifies the name of the property.
“ <a href="#">value</a> ” on page 118	only one	Specifies the value of the property.

### Example

```
<property>
  <name>use-unique-table-names</name>
  <value>>true</value>
</property>
```

## Q

## query-filter

Specifies the query filter for the CMP 1.1 finder.

### Superelements

“[finder](#)” on page 60 (`sun-ejb-jar.xml`)

### Subelements

none - contains data

## query-method

Specifies a query method.

### Superelements

[“prefetch-disabled” on page 88](#) (sun-ejb-jar.xml)

### Subelements

The following table describes subelements for the query-method element.

TABLE A-69 query-method Subelements

Element	Required	Description
<a href="#">“method-name” on page 82</a>	only one	Specifies a method name.
<a href="#">“method-params” on page 82</a>	only one	Specifies the fully qualified Java type names of method parameters.

## query-ordering

Specifies the query ordering for the CMP 1.1 finder.

### Superelements

[“finder” on page 60](#) (sun-ejb-jar.xml)

### Subelements

none - contains data

## query-params

Specifies the query parameters for the CMP 1.1 finder.

### Superelements

[“finder” on page 60](#) (sun-ejb-jar.xml)

### Subelements

none - contains data

---

## query-variables

Specifies variables in the query expression for the CMP 1.1 finder.

### Superelements

[“finder”](#) on page 60 (sun-ejb-jar.xml)

### Subelements

none - contains data

## R

## realm

Specifies the name of the realm used to process all authentication requests associated with this application. If this element is not specified or does not match the name of a configured realm, the default realm is used. For more information about realms, see “Realm Configuration” in *GlassFish v3 Application Server Developer’s Guide*.

### Superelements

[“as-context”](#) on page 34, [“login-config”](#) on page 73 (sun-ejb-jar.xml)

### Subelements

none - contains data

## refresh-field

Specifies a field that gives the application component a programmatic way to refresh a cached entry.

### Superelements

[“cache-mapping”](#) on page 40 (sun-web.xml)

### Subelements

none

## Attributes

The following table describes attributes for the `refresh-field` element.

TABLE A-70 `refresh-field` Attributes

Attribute	Default	Description
<code>name</code>	<code>none</code>	Specifies the input parameter name.
<code>scope</code>	<code>request.parameter</code>	(optional) Specifies the scope from which the input parameter is retrieved. Allowed values are <code>context.attribute</code> , <code>request.header</code> , <code>request.parameter</code> , <code>request.cookie</code> , <code>session.id</code> , and <code>session.attribute</code> .

## refresh-period-in-seconds

Specifies the rate at which a read-only-bean must be refreshed from the data source. If the value is less than or equal to zero, the bean is never refreshed; if the value is greater than zero, the bean instances are refreshed at the specified interval. This rate is just a hint to the container. Default is 0 (no refresh).

### Superelements

[“ejb” on page 53](#) (`sun-ejb-jar.xml`)

### Subelements

`none` - contains data

## removal-timeout-in-seconds

Specifies the amount of time a bean instance can remain idle in the container before it is removed (timeout). A value of 0 specifies that the container does not remove inactive beans automatically. The default value is 5400.

If `removal-timeout-in-seconds` is less than or equal to `cache-idle-timeout-in-seconds`, beans are removed immediately without being passivated.

Applies to stateful session beans.

For related information, see [“cache-idle-timeout-in-seconds” on page 40](#).

### Superelements

[“bean-cache” on page 35](#) (`sun-ejb-jar.xml`)

---

## Subelements

none - contains data

## remote-home-impl

Specifies the fully-qualified class name of the generated EJBHome impl class.

---

**Note** – This value is automatically generated by the server at deployment or redeployment time. Do not specify it or change it after deployment.

---

## Superelements

[“gen-classes” on page 61](#) (sun-ejb-jar.xml)

## Subelements

none - contains data

## remote-impl

Specifies the fully-qualified class name of the generated EJBObject impl class.

---

**Note** – This value is automatically generated by the server at deployment or redeployment time. Do not specify it or change it after deployment.

---

## Superelements

[“gen-classes” on page 61](#) (sun-ejb-jar.xml)

## Subelements

none - contains data

## request-protection

Defines the authentication policy requirements of the application’s request processing.

## Superelements

[“message-security” on page 79](#) (sun-web.xml, sun-ejb-jar.xml)

## Subelements

none

## Attributes

The following table describes attributes for the `request-protection` element.

TABLE A-71 `request-protection` Attributes

Attribute	Default	Description
<code>auth-source</code>	none	Specifies the type of required authentication, either sender (user name and password) or content (digital signature).
<code>auth-recipient</code>	none	Specifies whether recipient authentication occurs before or after content authentication. Allowed values are <code>before-content</code> and <code>after-content</code> .

## required

Specifies whether the authentication method specified in the [“auth-method” on page 34](#) element must be used for client authentication. The value is `true` or `false` (the default).

## Superelements

[“as-context” on page 34](#) (`sun-ejb-jar.xml`)

## Subelements

none - contains data

## res-ref-name

Specifies the `res-ref-name` in the corresponding Java EE deployment descriptor file resource-ref entry. The `res-ref-name` element specifies the name of a resource manager connection factory reference. The name must be unique within an enterprise bean.

## Superelements

[“resource-ref” on page 99](#) (`sun-web.xml`, `sun-ejb-jar.xml`)

## Subelements

none - contains data



## resize-quantity

Specifies the number of bean instances to be:

- Created, if a request arrives when the pool has less than “[steady-pool-size](#)” on page 108 quantity of beans (applies to pools only for creation). If the pool has more than `steady-pool-size` minus “[resize-quantity](#)” on page 97 of beans, then `resize-quantity` is still created.
- Removed, when the “[pool-idle-timeout-in-seconds](#)” on page 87 timer expires and a cleaner thread removes any unused instances.
  - For caches, when “[max-cache-size](#)” on page 75 is reached, `resize-quantity` beans are selected for passivation using the “[victim-selection-policy](#)” on page 118. In addition, the “[cache-idle-timeout-in-seconds](#)” on page 40 or “[removal-timeout-in-seconds](#)” on page 94 timers passivate beans from the cache.
  - For pools, when the “[max-pool-size](#)” on page 76 is reached, `resize-quantity` beans are selected for removal. In addition, the “[pool-idle-timeout-in-seconds](#)” on page 87 timer removes beans until `steady-pool-size` is reached.

Values are from 0 to `MAX_INTEGER`. The pool is not resized below the `steady-pool-size`. Default is 16.

Applies to stateless session beans, entity beans, and message-driven beans.

For EJB pools, the value can be defined in the EJB container. Default is 16.

For EJB caches, the value can be defined in the EJB container. Default is 32.

For message-driven beans, the value can be defined in the EJB container. Default is 2.

### Superelements

“[bean-cache](#)” on page 35, “[bean-pool](#)” on page 36 (`sun-ejb-jar.xml`)

### Subelements

none - contains data

## resource-adapter-mid

Specifies the module ID of the resource adapter that is responsible for delivering messages to the message-driven bean.

### Superelements

“[mdb-resource-adapter](#)” on page 77 (`sun-ejb-jar.xml`)

## Subelements

none - contains data

## resource-env-ref

Maps the `res-ref-name` in the corresponding Java EE deployment descriptor file `resource-env-ref` entry to the absolute `jndi-name` of a resource.

## Superelements

[“sun-web-app” on page 111](#) (`sun-web.xml`), [“ejb” on page 53](#) (`sun-ejb-jar.xml`)

## Subelements

The following table describes subelements for the `resource-env-ref` element.

TABLE A-72 `resource-env-ref` Subelements

Element	Required	Description
<a href="#">“resource-env-ref-name” on page 98</a>	only one	Specifies the <code>res-ref-name</code> in the corresponding Java EE deployment descriptor file <code>resource-env-ref</code> entry.
<a href="#">“jndi-name” on page 65</a>	only one	Specifies the absolute <code>jndi-name</code> of a resource.

## Example

```
<resource-env-ref>
  <resource-env-ref-name>jms/StockQueueName</resource-env-ref-name>
  <jndi-name>jms/StockQueue</jndi-name>
</resource-env-ref>
```

## resource-env-ref-name

Specifies the `res-ref-name` in the corresponding Java EE deployment descriptor file `resource-env-ref` entry.

## Superelements

[“resource-env-ref” on page 98](#) (`sun-web.xml`, `sun-ejb-jar.xml`)

## Subelements

none - contains data

## resource-ref

Maps the `res-ref-name` in the corresponding Java EE deployment descriptor file `resource-ref` entry to the absolute `jndi-name` of a resource.

### Superelements

“[sun-web-app](#)” on page 111 (`sun-web.xml`), “[ejb](#)” on page 53 (`sun-ejb-jar.xml`)

### Subelements

The following table describes subelements for the `resource-ref` element.

TABLE A-73 `resource-ref` Subelements

Element	Required	Description
“ <a href="#">res-ref-name</a> ” on page 96	only one	Specifies the <code>res-ref-name</code> in the corresponding Java EE deployment descriptor file <code>resource-ref</code> entry.
“ <a href="#">jndi-name</a> ” on page 65	only one	Specifies the absolute <code>jndi-name</code> of a resource.
“ <a href="#">default-resource-principal</a> ” on page 51	zero or one	Specifies the default principal (user) for the resource.

### Example

```
<resource-ref>
  <res-ref-name>jdbc/EmployeeDBName</res-ref-name>
  <jndi-name>jdbc/EmployeeDB</jndi-name>
</resource-ref>
```

## response-protection

Defines the authentication policy requirements of the application’s response processing.

### Superelements

“[message-security](#)” on page 79 (`sun-web.xml`, `sun-ejb-jar.xml`)

### Subelements

none

### Attributes

The following table describes attributes for the `response-protection` element.

TABLE A-74 response-protection Attributes

Attribute	Default	Description
auth-source	none	Specifies the type of required authentication, either sender (user name and password) or content (digital signature).
auth-recipient	none	Specifies whether recipient authentication occurs before or after content authentication. Allowed values are before-content and after-content.

## role-name

Contains the `role-name` in the `security-role` element of the corresponding Java EE deployment descriptor file.

### Superelements

[“security-role-mapping” on page 102](#) (`sun-web.xml`, `sun-ejb-jar.xml`)

### Subelements

none - contains data

## S

## sas-context

Describes the `sas-context` fields.

### Superelements

[“ior-security-config” on page 63](#) (`sun-ejb-jar.xml`)

### Subelements

The following table describes subelements for the `sas-context` element.

TABLE A-75 sas - context Subelements

Element	Required	Description
<a href="#">“caller-propagation” on page 41</a>	only one	Specifies whether the target accepts propagated caller identities. The values are NONE, SUPPORTED, or REQUIRED.

## schema-generator-properties

Specifies field-specific column attributes in property subelements.

### Superelements

[“cmp-resource” on page 44](#) (sun-ejb-jar.xml)

### Subelements

The following table describes subelements for the `schema-generator-properties` element.

TABLE A-76 schema-generator-properties Subelements

Element	Required	Description
<a href="#">“property (with subelements)” on page 91</a>	zero or more	Specifies a property name and value.

### Properties

The following table describes properties for the `schema-generator-properties` element.

TABLE A-77 schema-generator-properties Properties

Property	Default	Description
<code>use-unique-table-names</code>	false	Specifies that generated table names are unique within each application server domain. This property can be overridden during deployment. See <i>GlassFish v3 Application Server Developer's Guide</i> .
<code>bean-name.field-name.attribute</code>	none	Defines a column attribute. For attribute descriptions, see <a href="#">Table A-78</a> .

The following table lists the column attributes for properties defined in the `schema-generator-properties` element.

TABLE A-78 schema-generator-properties Column Attributes

Attribute	Description
jdbc-type	Specifies the JDBC type of the column created for the CMP field. The actual SQL type generated is based on this JDBC type but is database vendor specific.
jdbc-maximum-length	Specifies the maximum number of characters stored in the column corresponding to the CMP field. Applies only when the actual SQL that is generated for the column requires a length.  For example, a jdbc-maximum-length of 32 on a CMP String field such as firstName normally results in a column definition such as VARCHAR(32). But if the jdbc-type is CLOB and you are deploying on Oracle, the resulting column definition is CLOB. No length is given, because in an Oracle database, a CLOB has no length.
jdbc-precision	Specifies the maximum number of digits stored in a column which represents a numeric type.
jdbc-scale	Specifies the number of digits stored to the right of the decimal point in a column that represents a floating point number.
jdbc-nullable	Specifies whether the column generated for the CMP field allows null values.

## Example

```
<schema-generator-properties>
  <property>
    <name>Employee.firstName.jdbc-type</name>
    <value>char</value>
  </property>
  <property>
    <name>Employee.firstName.jdbc-maximum-length</name>
    <value>25</value>
  </property>
  <property>
    <name>use-unique-table-names</name>
    <value>true</value>
  </property>
</schema-generator-properties>
```

## security-role-mapping

Maps roles to users or groups in the currently active realm. See “Realm Configuration” in *GlassFish v3 Application Server Developer’s Guide*.

The role mapping element maps a role, as specified in the EJB JAR role-name entries, to a environment-specific user or group. If it maps to a user, it must be a concrete user which exists in the current realm, who can log into the server using the current authentication method. If it maps to a group, the realm must support groups and the group must be a concrete group which exists in the current realm. To be useful, there must be at least one user in that realm who belongs to that group.

## Superelements

“[sun-web-app](#)” on page 111 (`sun-web.xml`), “[sun-ejb-jar](#)” on page 111 (`sun-ejb-jar.xml`)

## Subelements

The following table describes subelements for the `security-role-mapping` element.

TABLE A-79 `security-role-mapping` Subelements

Element	Required	Description
“ <a href="#">role-name</a> ” on page 100	only one	Contains the <code>role-name</code> in the <code>security-role</code> element of the corresponding Java EE deployment descriptor file.
“ <a href="#">principal-name</a> ” on page 89	one or more if no <code>group-name</code> , otherwise zero or more	Contains a principal (user) name in the current realm. In an enterprise bean, the principal must have the <code>run-as</code> role specified.
“ <a href="#">group-name</a> ” on page 61	one or more if no <code>principal-name</code> , otherwise zero or more	Contains a group name in the current realm.

## service-endpoint-interface

Specifies the web service reference name relative to `java:comp/env`.

## Superelements

“[port-info](#)” on page 88 (`sun-web.xml`, `sun-ejb-jar.xml`)

## Subelements

none - contains data

## service-impl-class

Specifies the name of the generated service implementation class.

## Superelements

“[service-ref](#)” on page 104 (`sun-web.xml`, `sun-ejb-jar.xml`)

## Subelements

none - contains data

## service-qname

Specifies the WSDL service element that is being referred to.

### Superelements

“[service-ref](#)” on page 104, “[webservice-endpoint](#)” on page 120 (`sun-web.xml`, `sun-ejb-jar.xml`)

### Subelements

The following table describes subelements for the `service-qname` element.

TABLE A-80 `service-qname` subelements

Element	Required	Description
“ <a href="#">namespaceURI</a> ” on page 83	only one	Specifies the namespace URI.
“ <a href="#">localpart</a> ” on page 72	only one	Specifies the local part of a QNAME.

## service-ref

Specifies runtime settings for a web service reference. Runtime information is only needed in the following cases:

- To define the port used to resolve a container-managed port
- To define the default Stub/Call property settings for Stub objects
- To define the URL of a final WSDL document to be used instead of the one associated with the `service-ref` in the standard Java EE deployment descriptor

### Superelements

“[sun-web-app](#)” on page 111 (`sun-web.xml`), “[ejb](#)” on page 53 (`sun-ejb-jar.xml`)

### Subelements

The following table describes subelements for the `service-ref` element.

TABLE A-81 `service-ref` subelements

Element	Required	Description
“ <a href="#">service-ref-name</a> ” on page 105	only one	Specifies the web service reference name relative to <code>java:comp/env</code> .
“ <a href="#">port-info</a> ” on page 88	zero or more	Specifies information for a port within a web service reference.



TABLE A-81 service-ref subelements (Continued)

Element	Required	Description
<a href="#">“call-property” on page 41</a>	zero or more	Specifies JAX-RPC property values that can be set on a <code>javax.xml.rpc.Call</code> object before it is returned to the web service client.
<a href="#">“wsdl-override” on page 121</a>	zero or one	Specifies a valid URL pointing to a final WSDL document.
<a href="#">“service-impl-class” on page 103</a>	zero or one	Specifies the name of the generated service implementation class.
<a href="#">“service-qname” on page 104</a>	zero or one	Specifies the WSDL service element that is being referenced.

## service-ref-name

Specifies the web service reference name relative to `java:comp/env`.

### Superelements

[“service-ref” on page 104](#) (`sun-web.xml`, `sun-ejb-jar.xml`)

### Subelements

none - contains data

## servlet

Specifies a principal name for a servlet. Used for the `run-as` role defined in `web.xml`.

### Superelements

[“sun-web-app” on page 111](#) (`sun-web.xml`)

### Subelements

The following table describes subelements for the `servlet` element.

TABLE A-82 servlet Subelements

Element	Required	Description
<a href="#">“servlet-name” on page 106</a>	only one	Contains the name of a servlet, which is matched to a <code>servlet-name</code> in <code>web.xml</code> .
<a href="#">“principal-name” on page 89</a>	zero or one	Contains a principal (user) name in the current realm.

TABLE A-82 `servlet` Subelements (Continued)

Element	Required	Description
<a href="#">“webservice-endpoint” on page 120</a>	zero or more	Specifies information about a web service endpoint.

## **servlet-impl-class**

Specifies the automatically generated name of the servlet implementation class.

### **Superelements**

[“webservice-endpoint” on page 120](#) (`sun-web.xml`, `sun-ejb-jar.xml`)

### **Subelements**

none - contains data

## **servlet-name**

Specifies the name of a servlet, which is matched to a `servlet-name` in `web.xml`. This name must be present in `web.xml`.

### **Superelements**

[“cache-mapping” on page 40](#), [“servlet” on page 105](#) (`sun-web.xml`)

### **Subelements**

none - contains data

## **session-config**

Specifies session configuration information. Overrides the web container settings for an individual web application.

### **Superelements**

[“sun-web-app” on page 111](#) (`sun-web.xml`)

### **Subelements**

The following table describes subelements for the `session-config` element.

TABLE A-83 session-config Subelements

Element	Required	Description
<a href="#">“session-manager” on page 107</a>	zero or one	Specifies session manager configuration information.
<a href="#">“session-properties” on page 107</a>	zero or one	Specifies session properties.
<a href="#">“cookie-properties” on page 48</a>	zero or one	Specifies session cookie properties.

## session-manager

Specifies session manager information.

### Superelements

[“session-config” on page 106](#) (sun-web.xml)

### Subelements

The following table describes subelements for the session-manager element.

TABLE A-84 session-manager Subelements

Element	Required	Description
<a href="#">“manager-properties” on page 73</a>	zero or one	Specifies session manager properties.
<a href="#">“store-properties” on page 108</a>	zero or one	Specifies session persistence (storage) properties.

### Attributes

The following table describes attributes for the session-manager element.

TABLE A-85 session-manager Attributes

Attribute	Default	Description
persistence-type	memory	(optional) Specifies the session persistence mechanism. Allowed values are memory, file, and replicated.

## session-properties

Specifies session properties.

### Superelements

[“session-config” on page 106](#) (sun-web.xml)

## Subelements

The following table describes subelements for the `session-properties` element.

TABLE A-86 `session-properties` Subelements

Element	Required	Description
<code>"property (with attributes)"</code> on page 90	zero or more	Specifies a property, which has a name and a value.

## Properties

The following table describes properties for the `session-properties` element.

TABLE A-87 `session-properties` Properties

Property	Default	Description
<code>timeoutSeconds</code>	1800	Specifies the default maximum inactive interval (in seconds) for all sessions created in this web module. If set to 0 or less, sessions in this web module never expire.  If a <code>session-timeout</code> element is specified in the <code>web.xml</code> file, the <code>session-timeout</code> value overrides any <code>timeoutSeconds</code> value. If neither <code>session-timeout</code> nor <code>timeoutSeconds</code> is specified, the <code>timeoutSeconds</code> default is used.  Note that the <code>session-timeout</code> element in <code>web.xml</code> is specified in minutes, not seconds.
<code>enableCookies</code>	true	Uses cookies for session tracking if set to true.
<code>enableURLRewriting</code>	true	Enables URL rewriting. This provides session tracking via URL rewriting when the browser does not accept cookies. You must also use an <code>encodeURL</code> or <code>encodeRedirectURL</code> call in the servlet or JSP.

## steady-pool-size

Specifies the initial and minimum number of bean instances that are maintained in the pool. Default is 32. Applies to stateless session beans and message-driven beans.

## Superelements

`"bean-pool"` on page 36 (`sun-ejb-jar.xml`)

## Subelements

none - contains data

## store-properties

Specifies session persistence (storage) properties.

## Superelements

“[session-manager](#)” on page 107 (`sun-web.xml`)

## Subelements

The following table describes subelements for the `store-properties` element.

TABLE A-88 `store-properties` Subelements

Element	Required	Description
“ <a href="#">property (with attributes)</a> ” on page 90	zero or more	Specifies a property, which has a name and a value.

## Properties

The following table describes properties for the `store-properties` element.

TABLE A-89 `store-properties` Properties

Property	Default	Description
<code>directory</code>	<code>domain-dir/generated/jsp/app-name/app-name_war</code>	Specifies the absolute or relative pathname of the directory into which individual session files are written. A relative path is relative to the temporary work directory for this web application.  Applicable only if the <code>persistence-type</code> attribute of the parent “ <a href="#">session-manager</a> ” on page 107 element is <code>file</code> .
<code>persistenceScope</code>	<code>session</code>	Specifies how much of the session state is stored. Allowed values are as follows: <ul style="list-style-type: none"> <li>■ <code>session</code> - The entire session state is stored every time. This mode provides the best guarantee that your session data is correctly stored for any distributable web application.</li> <li>■ <code>modified-session</code> - The entire session state is stored if it has been modified. A session is considered to have been modified if <code>HttpSession.setAttribute()</code> or <code>HttpSession.removeAttribute()</code> was called. You must guarantee that <code>setAttribute()</code> is called every time an attribute is changed. This is not a Java EE specification requirement, but it is required for this mode to work properly.</li> <li>■ <code>modified-attribute</code> - Only modified session attributes are stored. For this mode to work properly, you must follow some guidelines, which are explained immediately following this table.</li> </ul> Applicable only if the <code>persistence-type</code> attribute of the parent “ <a href="#">session-manager</a> ” on page 107 element is <code>replicated</code> .

If the `persistenceScope` store property is set to `modified-attribute`, a web application must follow these guidelines:

- Call `setAttribute()` every time the session state is modified.
- Make sure there are no cross-references between attributes. The object graph under each distinct attribute key is serialized and stored separately. If there are any object cross references between the objects under each separate key, they are not serialized and deserialized correctly.
- Distribute the session state across multiple attributes, or at least between a read-only attribute and a modifiable attribute.

## stub-property

Specifies JAX-RPC property values that are set on a `javax.xml.rpc.Stub` object before it is returned to the web service client. The property names can be any properties supported by the JAX-RPC `Stub` implementation.

### Superelements

[“port-info” on page 88](#) (`sun-web.xml`, `sun-ejb-jar.xml`)

### Subelements

The following table describes subelements for the `stub-property` element.

TABLE A-90 `stub-property` subelements

Element	Required	Description
<a href="#">“name” on page 83</a>	only one	Specifies the name of the entity.
<a href="#">“value” on page 118</a>	only one	Specifies the value of the entity.

### Properties

The following table describes properties for the `stub-property` element.

TABLE A-91 `stub-property` properties

Property	Default	Description
<code>jbi-enabled</code>	<code>true</code>	Determines whether the visibility of this endpoint as a Java™ Business Integration service is enabled or disabled.

### Example

```
<service-ref>
<service-ref-name>service/FooProxy</service-ref-name>
  <port-info>
```

```

<service-endpoint-interface>a.FooPort</service-endpoint-interface>
<wsdl-port>
  <namespaceURI>urn:Foo</namespaceURI>
  <localpart>FooPort</localpart>
</wsdl-port>
<stub-property>
  <name>javax.xml.rpc.service.endpoint.address</name>
  <value>http://localhost:8080/a/Foo</value>
</stub-property>
</port-info>
</service-ref>

```

## sun-ejb-jar

Defines the Application Server specific configuration for an EJB JAR file. This is the root element; there can only be one sun-ejb-jar element in a sun-ejb-jar.xml file. See [“The sun-ejb-jar.xml File” on page 28](#).

### Superelements

none

### Subelements

The following table describes subelements for the sun-ejb-jar element.

TABLE A-92 sun-ejb-jar Subelements

Element	Required	Description
<a href="#">“security-role-mapping” on page 102</a>	zero or more	Maps a role in the corresponding Java EE XML file to a user or group.
<a href="#">“enterprise-beans” on page 58</a>	only one	Describes all the runtime properties for an EJB JAR file in the application.

## sun-web-app

Defines Application Server specific configuration for a web module. This is the root element; there can only be one sun-web-app element in a sun-web.xml file. See [“The sun-web.xml File” on page 24](#).

### Superelements

none

## Subelements

The following table describes subelements for the `sun-web-app` element.

TABLE A-93 sun-web-app Subelements

Element	Required	Description
<a href="#">“context-root” on page 48</a>	zero or one	Contains the web context root for the web application.
<a href="#">“security-role-mapping” on page 102</a>	zero or more	Maps roles to users or groups in the currently active realm.
<a href="#">“servlet” on page 105</a>	zero or more	Specifies a principal name for a servlet, which is used for the <code>run-as</code> role defined in <code>web.xml</code> .
<a href="#">“idempotent-url-pattern” on page 62</a>	zero or more	Specifies a URL pattern for idempotent requests.
<a href="#">“session-config” on page 106</a>	zero or one	Specifies session manager, session cookie, and other session-related information.
<a href="#">“ejb-ref” on page 56</a>	zero or more	Maps the absolute JNDI name to the <code>ejb-ref</code> in the corresponding Java EE XML file.
<a href="#">“resource-ref” on page 99</a>	zero or more	Maps the absolute JNDI name to the <code>resource-ref</code> in the corresponding Java EE XML file.
<a href="#">“resource-env-ref” on page 98</a>	zero or more	Maps the absolute JNDI name to the <code>resource-env-ref</code> in the corresponding Java EE XML file.
<a href="#">“service-ref” on page 104</a>	zero or more	Specifies runtime settings for a web service reference.
<a href="#">“message-destination-ref” on page 79</a>	zero or more	Specifies the name of a physical message destination.
<a href="#">“cache” on page 37</a>	zero or one	Configures caching for web application components.
<a href="#">“class-loader” on page 42</a>	zero or one	Specifies class loader configuration information.
<a href="#">“jsp-config” on page 66</a>	zero or one	Specifies JSP configuration information.
<a href="#">“locale-charset-info” on page 71</a>	zero or one	Deprecated. Use the <code>parameter-encoding</code> subelement of <code>sun-web-app</code> instead.
<a href="#">“parameter-encoding” on page 84</a>	zero or one	Determines the default request character encoding and how the web container decodes parameters from forms according to a hidden field value.
<a href="#">“property (with attributes)” on page 90</a>	zero or more	Specifies a property, which has a name and a value.
<a href="#">“message-destination” on page 78</a>	zero or more	Specifies the name of a logical message destination.



TABLE A-93 sun-web-app Subelements (Continued)

Element	Required	Description
<a href="#">“webservice-description” on page 119</a>	zero or more	Specifies a name and optional publish location for a web service.

## Attributes

The following table describes attributes for the sun-web-app element.

TABLE A-94 sun-web-app Attributes

Attribute	Default	Description
error-url	(blank)	(optional) Not implemented. Do not use.
httpServlet-security-provider	none	(optional) Specifies the HttpServlet message layer provider that the web container's servlet auth-constraint processing calls.

## Properties

The following table describes properties for the sun-web-app element.

TABLE A-95 sun-web-app Properties

Property	Default	Description
allowLinking	false	<p>If true, resources in this web application that are symbolic links are served. You can also define this property for a virtual server. Web applications on the virtual server that do not define this property use the virtual server's value.</p> <p><b>Caution</b> – Setting this property to true on Windows systems exposes JSP source code.</p>

TABLE A-95 sun-web-app Properties (Continued)

Property	Default	Description
alternatedocroot_ <i>n</i>	none	<p>Specifies an alternate document root (docroot), where <i>n</i> is a positive integer that allows specification of more than one. Alternate docroots allow web applications to serve requests for certain resources from outside their own docroot, based on whether those requests match one (or more) of the URI patterns of the web application's alternate docroots.</p> <p>If a request matches an alternate docroot's URI pattern, it is mapped to the alternate docroot by appending the request URI (minus the web application's context root) to the alternate docroot's physical location (directory). If a request matches multiple URI patterns, the alternate docroot is determined according to the following precedence order:</p> <ul style="list-style-type: none"> <li>■ Exact match</li> <li>■ Longest path match</li> <li>■ Extension match</li> </ul> <p>For example, the following properties specify three alternate docroots. The URI pattern of the first alternate docroot uses an exact match, whereas the URI patterns of the second and third alternate docroots use extension and longest path prefix matches, respectively.</p> <pre>&lt;property name="alternatedocroot_1"   value="from=/my.jpg dir=/srv/images/jpg"/&gt; &lt;property name="alternatedocroot_2"   value="from=*.jpg dir=/srv/images/jpg"/&gt; &lt;property name="alternatedocroot_3"   value="from=/jpg/* dir=/src/images"/&gt;</pre> <p>The value of each alternate docroot has two components: The first component, <code>from</code>, specifies the alternate docroot's URI pattern, and the second component, <code>dir</code>, specifies the alternate docroot's physical location (directory). Spaces are allowed in the <code>dir</code> component.</p> <p>You can set this property for all the web applications on a specific virtual server.</p>
crossContextAllowed	true	<p>If <code>true</code>, allows this web application to access the contexts of other web applications using the <code>ServletContext.getContext()</code> method.</p>
relativeRedirectAllowed	false	<p>If <code>true</code>, allows this web application to send a relative URL to the client using <code>HttpServletResponse.sendRedirect()</code>, and instructs the web container not to translate any relative URLs to fully qualified ones.</p>

TABLE A-95 sun-web-app Properties (Continued)

Property	Default	Description
reuseSessionID	false	If true, sessions generated for this web application use the session ID specified in the request.
securePagesWithPragma	true	Set this property to false to ensure that for this web application file downloads using SSL work properly in Internet Explorer.  You can set this property for all the web applications on a specific virtual server.
singleThreadedServletPoolSize	5	Specifies the maximum number of servlet instances allocated for each SingleThreadModel servlet in the web application.
tempdir	<i>domain-dir/generated/app-name</i> or <i>domain-dir/generated/module-name</i>	Specifies a temporary directory for use by this web module. This value is used to construct the value of the <code>javax.servlet.context.tempdir</code> context attribute. Compiled JSP files are also placed in this directory.
useResponseCTForHeaders	false	If true, response headers are encoded using the response's charset instead of the default (UTF-8).

## T

### tie-class

Specifies the automatically generated name of a tie implementation class for a port component.

### Superelements

“[webservice-endpoint](#)” on page 120 (`sun-web.xml`, `sun-ejb-jar.xml`)

### Subelements

none - contains data

### timeout

Specifies the “[cache-mapping](#)” on page 40 specific maximum amount of time in seconds that an entry can remain in the cache after it is created or refreshed. If not specified, the default is the value of the `timeout` attribute of the “[cache](#)” on page 37 element.

### Superelements

“[cache-mapping](#)” on page 40 (`sun-web.xml`)

## Subelements

none - contains data

## Attributes

The following table describes attributes for the `timeout` element.

TABLE A-96 `timeout` Attributes

Attribute	Default	Description
<code>name</code>	<code>none</code>	Specifies the timeout input parameter, whose value is interpreted in seconds. The field's type must be <code>java.lang.Long</code> or <code>java.lang.Integer</code> .
<code>scope</code>	<code>request.attribute</code>	(optional) Specifies the scope from which the input parameter is retrieved. Allowed values are <code>context.attribute</code> , <code>request.header</code> , <code>request.parameter</code> , <code>request.cookie</code> , <code>request.attribute</code> , and <code>session.attribute</code> .

## transport-config

Specifies the security transport information.

## Superelements

[“ior-security-config” on page 63](#) (`sun-ejb-jar.xml`)

## Subelements

The following table describes subelements for the `transport-config` element.

TABLE A-97 `transport-config` Subelements

Element	Required	Description
<a href="#">“integrity” on page 63</a>	only one	Specifies if the target supports integrity-protected messages. The values are NONE, SUPPORTED, or REQUIRED.
<a href="#">“confidentiality” on page 46</a>	only one	Specifies if the target supports privacy-protected messages. The values are NONE, SUPPORTED, or REQUIRED.
<a href="#">“establish-trust-in-target” on page 59</a>	only one	Specifies if the target is capable of authenticating <i>to</i> a client. The values are NONE, SUPPORTED, or REQUIRED.
<a href="#">“establish-trust-in-client” on page 59</a>	only one	Specifies if the target is capable of authenticating a client. The values are NONE, SUPPORTED, or REQUIRED.

---

## transport-guarantee

Specifies that the communication between client and server is NONE, INTEGRAL, or CONFIDENTIAL.

- NONE means the application does not require any transport guarantees.
- INTEGRAL means the application requires that the data sent between client and server be sent in such a way that it can't be changed in transit.
- CONFIDENTIAL means the application requires that the data be transmitted in a fashion that prevents other entities from observing the contents of the transmission.

In most cases, a value of INTEGRAL or CONFIDENTIAL indicates that the use of SSL is required.

### Superelements

[“webservice-endpoint” on page 120](#) (sun-web.xml, sun-ejb-jar.xml)

### Subelements

none - contains data

## U

## unique-id

Contains the unique ID for the application. This value is automatically updated each time the application is deployed or redeployed. Do not edit this value.

### Superelements

[“enterprise-beans” on page 58](#) (sun-ejb-jar.xml)

### Subelements

none - contains data

## url-pattern

Specifies a servlet URL pattern for which caching is enabled. See the Servlet 2.4 specification section SRV. 11.2 for applicable patterns.

### Superelements

[“cache-mapping” on page 40](#) (sun-web.xml)

## Subelements

none - contains data

## use-thread-pool-id

Specifies the thread pool from which threads are selected for remote invocations of this bean.

## Superelements

[“ejb” on page 53](#) (sun-ejb-jar.xml)

## Subelements

none - contains data

# V

## value

Specifies the value of the entity.

## Superelements

[“call-property” on page 41](#), [“stub-property” on page 110](#) (sun-web.xml, sun-ejb-jar.xml);  
[“property \(with subelements\)” on page 91](#) (sun-ejb-jar.xml)

## Subelements

none - contains data

## victim-selection-policy

Specifies how stateful session beans are selected for passivation. Possible values are First In, First Out (FIFO), Least Recently Used (LRU), Not Recently Used (NRU). The default value is NRU, which is actually pseudo-LRU.

---

**Note** – You cannot plug in your own victim selection algorithm.

---

The victims are generally passivated into a backup store (typically a file system or database). This store is cleaned during startup, and also by a periodic background process that removes idle entries as specified by `removal-timeout-in-seconds`. The backup store is monitored by a background thread (or sweeper thread) to remove unwanted entries.

Applies to stateful session beans.

## Superelements

“[bean-cache](#)” on page 35 (`sun-ejb-jar.xml`)

## Subelements

none - contains data

## Example

```
<victim-selection-policy>LRU</victim-selection-policy>
```

If both SSL2 and SSL3 are enabled, the server tries SSL3 encryption first. If that fails, the server tries SSL2 encryption. If both SSL2 and SSL3 are enabled for a virtual server, the server tries SSL3 encryption first. If that fails, the server tries SSL2 encryption.

# W

## webservice-description

Specifies a name and optional publish location for a web service.

## Superelements

“[sun-web-app](#)” on page 111 (`sun-web.xml`), “[enterprise-beans](#)” on page 58 (`sun-ejb-jar.xml`)

## Subelements

The following table describes subelements for the `webservice-description` element.

TABLE A-98 webservice-description subelements

Element	Required	Description
<a href="#">“webservice-description-name” on page 120</a>	only one	Specifies a unique name for the web service within a web or EJB module.
<a href="#">“wsdl-publish-location” on page 122</a>	zero or one	Specifies the URL of a directory to which a web service’s WSDL is published during deployment.

## webservice-description-name

Specifies a unique name for the web service within a web or EJB module.

### Superelements

[“webservice-description” on page 119](#) (sun-web.xml, sun-ejb-jar.xml)

### Subelements

none - contains data

## webservice-endpoint

Specifies information about a web service endpoint.

### Superelements

[“servlet” on page 105](#) (sun-web.xml), [“ejb” on page 53](#) (sun-ejb-jar.xml)

### Subelements

The following table describes subelements for the webservice-endpoint element.

TABLE A-99 webservice-endpoint subelements

Element	Required	Description
<a href="#">“port-component-name” on page 87</a>	only one	Specifies a unique name for a port component within a web or EJB module.
<a href="#">“endpoint-address-uri” on page 57</a>	zero or one	Specifies the automatically generated endpoint address.
<a href="#">“login-config” on page 73</a>	zero or one	Specifies the authentication configuration for an EJB web service endpoint.



TABLE A-99 webservice-endpoint subelements (Continued)

Element	Required	Description
“message-security-binding” on page 80	zero or one	Specifies a custom authentication provider binding.
“transport-guarantee” on page 117	zero or one	Specifies that the communication between client and server is NONE, INTEGRAL, or CONFIDENTIAL.
“service-qname” on page 104	zero or one	Specifies the WSDL service element that is being referenced.
“tie-class” on page 115	zero or one	Specifies the automatically generated name of a tie implementation class for a port component.
“servlet-impl-class” on page 106	zero or one	Specifies the automatically generated name of the generated servlet implementation class.
“debugging-enabled” on page 50	zero or one	Specifies whether the debugging servlet is enabled for this web service endpoint. Allowed values are true and false (the default).
“property (with attributes)” on page 90 (sun-web.xml)	zero or more	Specifies a property, which has a name and a value.
“property (with subelements)” on page 91 (sun-ejb-jar.xml)		

## wSDL-override

Specifies a valid URL pointing to a final WSDL document. If not specified, the WSDL document associated with the `service-ref` in the standard Java EE deployment descriptor is used.

## Superelements

“service-ref” on page 104 (sun-web.xml, sun-ejb-jar.xml)

## Subelements

none - contains data

## Example

```
// available via HTTP
<wSDL-override>http://localhost:8000/myservice/myport?WSDL</wSDL-override>
```

```
// in a file
<wSDL-override>file:/home/user1/myfinalwSDL.wSDL</wSDL-override>
```

## wSDL-port

Specifies the WSDL port.

## Superelements

“[port-info](#)” on page 88 (sun-web.xml, sun-ejb-jar.xml)

## Subelements

The following table describes subelements for the `wSDL-port` element.

TABLE A-100 wSDL-port subelements

Element	Required	Description
“ <a href="#">namespaceURI</a> ” on page 83	only one	Specifies the namespace URI.
“ <a href="#">localpart</a> ” on page 72	only one	Specifies the local part of a QName.

## wSDL-publish-location

Specifies the URL of a directory to which a web service’s WSDL is published during deployment. Any required files are published to this directory, preserving their location relative to the module-specific WSDL directory (`META-INF/wSDL` or `WEB-INF/wSDL`).

## Superelements

“[webservice-description](#)” on page 119 (sun-web.xml, sun-ejb-jar.xml)

## Subelements

none - contains data

## Example

Suppose you have an `ejb.jar` file whose `webservicess.xml` file’s `wSDL-file` element contains the following reference:

```
META-INF/wSDL/a/Foo.wSDL
```

Suppose your `sun-ejb-jar` file contains the following element:

```
<wSDL-publish-location>file:/home/user1/publish</wSDL-publish-location>
```

The final WSDL is stored in `/home/user1/publish/a/Foo.wSDL`.

# Index

---

## A

- activation-config element, 32-33
- activation-config-property element, 33
- activation-config-property-name element, 33-34
- activation-config-property-value element, 34
- Admin Console
  - using for deployment, 21
  - using to disable web applications, 19
- agent attribute, 72
- allowLinking property, 113
- alternatedocroot\_*n* property, 114
- annotation, 16
- as-context element, 34
- asadmin deploy command, 21
  - force option, 19
- asadmin disable command, 21
- asadmin enable command, 21
- asadmin list-components command, 21
- asadmin redeploy command, 21
- asadmin undeploy command, 21
- assembly, overview, 15-17
- auth-layer attribute, 81
- auth-method element, 34
- auth-recipient attribute, 96, 100
- auth-source attribute, 96, 100
- autodeployment, 20
- availability-enabled attribute, 55

## B

- BaseCache cacheClassName value, 38

- bean-cache element, 35
- bean-pool element, 36
- BoundedMultiLruCache cacheClassName value, 39

## C

- cache element, 37-39
- cache-helper element, 39
- cache-helper-ref element, 39-40
- cache-idle-timeout-in-seconds element, 40
- cache-mapping element, 40-41
- cache-on-match attribute, 47
- cache-on-match-failure attribute, 47, 48
- cacheClassName property, 38
- CacheHelper interface, 39
- cacheKeyGeneratorAttrName property, 51
- call-property element, 41
- caller-propagation element, 41
- charset attribute, 72
- checkInterval property, 67
- checkpoint-at-end-of-method element, 42
- checkpointed-methods element, 42
- class loader delegation model, 43
- class-loader element, 42-44
- class-name attribute, 39, 90
- classdebuginfo property, 67
- classpath property, 67
- cmp element, 44
- cmp-resource element, 44-45
- cmt-timeout-in-seconds element, 45
- commit-option element, 45

- compiler property, 67
- compilerSourceVM property, 67
- compilerTargetVM property, 67
- confidentiality element, 46
- constraint-field element, 46-47
- constraint-field-value element, 47-48
- context-root element, 48
- cookie-properties element, 48-49
- cookieComment property, 49
- cookieDomain property, 49
- cookieMaxAgeSeconds property, 49
- cookiePath property, 49
- cookieSecure property, 49
- create-tables-at-deploy element, 49
- crossContextAllowed property, 114

## D

- database-vendor-name element, 50
- debugging-enabled element, 50
- default-charset attribute, 85
- default-helper element, 50-51
- default-locale attribute, 71
- default-resource-principal element, 51-52
- defaultBufferNone property, 67
- delegate attribute, 43
- delegation model for classloaders, 43
- deploydir command (deprecated), 21
- deployment
  - directory deployment (deprecated), 21
  - disabling deployed web modules, 19
  - dynamic, 18, 19
  - errors during, 18
  - forcing, 19
  - life cycle, 18-19
  - of web applications, 22
  - overview, 15-17
  - redeployment, 19
  - standard Java EE descriptor, 16
  - Sun Java System Application Server descriptor, 17
  - Sun Java System Application Server
    - descriptors, 23-24
  - tools for, 21-22
  - undeploying a web module, 21

- deployment (*Continued*)
  - using the Admin Console, 21
- description element, 52
- development property, 67
- directory deployment (deprecated), 21
- directory property, 109
- dispatcher element, 52
- domain, deploying web applications to, 18
- drop-tables-at-undeploy element, 52
- DTD files, 24
  - location of, 24
- dumpSmap property, 67
- dynamic, deployment, 19
- dynamic deployment, 18
- dynamic-reload-interval attribute, 43

## E

- EJB components, elements, 58-59
- ejb element, 53-55
- ejb-name element, 56
- ejb-ref element, 56-57
- ejb-ref-name element, 57
- elements in XML files, 58-59
- enableCookies property, 108
- enabled attribute, 38
- enablePooling property, 67
- enableTldValidation property, 67
- enableURLRewriting property, 108
- encoding, of JSP files, 68
- endpoint-address-uri element, 57-58
- enterprise-beans element, 58
- error-url attribute, 113
- errorOnUseBeanInvalidClassAttribute property, 67
- errors during deployment, 18
- establish-trust-in-client element, 59
- establish-trust-in-target element, 59
- extra-class-path attribute, 43

## F

- finder element, 60
- flush-at-end-of-method element, 60-61

forcing deployment, 19  
fork property, 67  
form-hint-field attribute, 85

## G

genStrAsByteArray property, 68  
genStrAsCharArray property, 68  
getParameter method, 85  
getReader method, 85  
group-name element, 61  
groups in realms, 102

## H

http-method element, 62  
httpMethods property, 68  
httpServlet-security-provider attribute, 113

## I

idempotent-url-pattern element, 62-63  
ieClassId property, 68  
ignoreHiddenJarFiles property, 44  
ignoreJspFragmentErrors property, 68  
initialCapacity property, 68  
integrity element, 63  
ior-security-config element, 63  
is-cache-overflow-allowed element, 63-64  
is-one-one-cmp element, 64  
is-read-only-bean element, 64

## J

Java EE, standard deployment descriptor, 16  
java-method element, 64-65  
javaEncoding property, 68  
jbi-enabled property, 110  
jms-durable-subscription-name element, 65  
jms-max-messages-load, 65  
jndi-name element, 65

jsp-config element, 22, 66-69  
JSP files  
    configuring, 66-69  
    encoding of, 68  
    generated source code, 22

## K

-keepgenerated flag, 22  
keepgenerated property, 68  
key-field element, 69-70

## L

lib directory, DTD file location, 24  
locale attribute, 72  
locale-charset-info element, 71  
locale-charset-map element, 71-72  
localpart element, 72-73  
login-config element, 73  
LruCache cacheClassName value, 38

## M

manager-properties element, 73-75  
mappedfile property, 68  
mapping-properties element, 75  
match-expr attribute, 47  
max-cache-size element, 75  
max-entries attribute, 38  
max-pool-size element, 76  
max-wait-time-in-millis element, 76  
maxSessions property, 74  
MaxSize property, 38  
mdb-connection-factory element, 76  
mdb-resource-adapter element, 77  
message-destination element, 78  
message-destination-name element, 78  
message-destination-ref element, 79  
message-destination-ref-name element, 79  
message element, 77-78  
message-security-binding element, 80-81

message-security element, 79-80  
method element, 81  
method-intf element, 81-82  
method-name element, 82  
method-param element, 82  
method-params element, 82-83  
modificationTestInterval property, 68  
MultiLruCache cacheClassName value, 38  
MultiLRUSegmentSize property, 38

## N

name element, 83  
namespaceURI element, 83  
NetBeans, using for assembly, 17  
no-of-retries attribute, 62

## O

one-one-finders element, 84  
operation-name element, 84

## P

packaging, *See* assembly  
parameter-encoding element, 84-85  
pass-by-reference element, 85-86  
pass-by-value semantics, 85  
password element, 86-87  
persistence-type attribute, 107  
persistenceFrequency property, 75  
persistenceScope property, 109  
plugin tag, 68  
pm-descriptors element, 87  
pool-idle-timeout-in-seconds element, 87  
port-component-name element, 87  
port-info element, 88  
prefetch-disabled element, 88  
principal element, 89  
principal-name element, 89-90  
properties  
    about, 90,91

property element, 90,91  
provider-id attribute, 81

## Q

query-filter element, 91  
query-method element, 92  
query-ordering element, 92  
query-params element, 92  
query-variables element, 93

## R

realm element, 93  
realms, mapping groups and users to, 102  
reapIntervalSeconds property, 74  
redeploying web applications, 18  
redeployment, 19  
refresh-field element, 93-94  
refresh-period-in-seconds element, 94  
relativeRedirectAllowed property, 114  
reload-interval property, 68  
removal-timeout-in-seconds element, 94  
request-protection element, 95-96  
required element, 96  
res-ref-name element, 96  
resize-quantity element, 97  
resource-adapter-mid element, 97-98  
resource-env-ref element, 98  
resource-env-ref-name element, 98  
resource-ref element, 99  
response-protection element, 99-100  
reuseSessionID property, 115  
role-name element, 100

## S

sas-context element, 100  
saveBytecode property, 68  
schema-generator-properties element, 101-102  
scope attribute, 47,70,94,116  
scratchdir property, 68

securePagesWithPragma property, 115  
 security-role-mapping element, 102-103  
 server  
   lib directory of, 24  
   Sun Java System Application Server deployment  
     descriptor, 17  
   Sun Java System Application Server deployment  
     descriptors, 23-24  
 service-endpoint-interface element, 103  
 service-impl-class element, 103  
 service-qname element, 104  
 service-ref element, 104-105  
 service-ref-name element, 105  
 servlet element, 105-106  
 servlet-impl-class element, 106  
 servlet-name element, 106  
 session-config element, 106-107  
 session-manager element, 107  
 session-properties element, 107-108  
 session-timeout element, 108  
 sessionFilename property, 74  
 sessions, and dynamic redeployment, 19  
 singleThreadedServletPoolSize property, 115  
 steady-pool-size element, 108  
 store-properties element, 108-110  
 stub-property element, 110-111  
 sun-ejb-jar\_3\_0-0.dtd file, 24  
 sun-ejb-jar element, 111  
 sun-ejb-jar.xml file, 24  
   elements in, 28-32  
   example of, 32  
 sun-web-app\_2\_5-0.dtd file, 24  
 sun-web-app element, 111-115  
 sun-web.xml file, 22, 24  
   elements in, 24-27  
   example of, 27  
 suppressSmap property, 69

## T

tempdir property, 115  
 tie-class element, 115  
 timeout element, 115-116  
 timeout-in-seconds attribute, 38

timeoutSeconds property, 108  
 tools, for deployment, 21-22  
 transport-config element, 116  
 transport-guarantee element, 117  
 trimSpaces property, 69

## U

unique-id element, 117  
 url-pattern attribute, 62  
 url-pattern element, 117-118  
 use-thread-pool-id element, 118  
 use-unique-table-names property, 101  
 usePrecompiled property, 69  
 useResponseCTForHeaders property, 115  
 users in realms, 102

## V

value attribute, 90  
 value element, 118  
 victim-selection-policy element, 118

## W

web applications  
   deploying, 22  
   directory structure, 17  
   disabling, 19  
   module definition, 15  
   naming, 17  
   redeploying, 18  
 web modules  
   *See also* applications  
   definition, 15-16  
   directory structure, 17  
   disabling, 19  
   naming, 17  
 web services, debugging, 50  
 webservice-description element, 119-120  
 webservice-description-name element, 120  
 webservice-endpoint element, 120-121

wsdl-override element, 121  
wsdl-port element, 121-122  
wsdl-publish-location element, 122

## **X**

XML specification, 24  
xpoweredBy property, 69