



Sun Java™ System

Application Server Enterprise Edition

8.1

Administration Reference

2005Q1

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

Part No: 819-0221

Copyright © 2004 - 2005 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 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 of the U.S. patents listed at <http://www.sun.com/patents> and one or more additional patents or pending patent applications in the U.S. and in other countries.

THIS PRODUCT CONTAINS CONFIDENTIAL INFORMATION AND TRADE SECRETS OF SUN MICROSYSTEMS, INC. USE, DISCLOSURE OR REPRODUCTION IS PROHIBITED WITHOUT THE PRIOR EXPRESS WRITTEN PERMISSION OF SUN MICROSYSTEMS, INC.

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.

Use is subject to license terms.

This distribution may include materials developed by third parties.

Sun, Sun Microsystems, the Sun logo, Java, and the Java Coffee Cup logo 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 architecture developed by Sun Microsystems, Inc.

This product is covered and controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical 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 © 2004 - 2005 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, Etats-Unis. 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 plus des brevets américains listés à l'adresse <http://www.sun.com/patents> et un ou les brevets supplémentaires ou les applications de brevet en attente aux Etats - Unis et dans les autres pays.

CE PRODUIT CONTIENT DES INFORMATIONS CONFIDENTIELLES ET DES SECRETS COMMERCIAUX DE SUN MICROSYSTEMS, INC. SON UTILISATION, SA DIVULGATION ET SA REPRODUCTION SONT INTERDITES SANS L'AUTORISATION EXPRESSE, ECRITE ET PREALABLE DE SUN MICROSYSTEMS, INC.

L'utilisation est soumise aux termes de la License.

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

Sun, Sun Microsystems, le logo Sun, Java, et le logo Java Coffee Cup sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays.

Ce produit est soumis à la législation américaine en matière de contrôle des exportations et peut être soumis à la réglementation en vigueur dans d'autres pays dans le domaine des exportations et importations. Les utilisations, ou utilisateurs finaux, pour des armes nucléaires, des missiles, des armes biologiques et chimiques ou du nucléaire maritime, directement ou indirectement, sont strictement interdites. Les exportations ou réexportations vers les pays sous embargo américain, ou vers des entités figurant sur les listes d'exclusion d'exportation américaines, y compris, mais de manière non exhaustive, 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

Preface	7
Who Should Use This Book	7
Before You Read This Book	7
How This Book Is Organized	8
Conventions Used in This Book	8
Typographic Conventions	8
Symbols	8
Default Paths and File Names	9
Shell Prompts	10
Related Documentation	11
Books in This Documentation Set	11
Other Server Documentation	12
Accessing Sun Resources Online	12
Contacting Sun Technical Support	13
Related Third-Party Web Site References	13
Sun Welcomes Your Comments	13
Chapter 1 The domain.xml File	15
About the domain.xml File	15
The sun-domain_1_1.dtd File	16
Default Values	16
Variables	17
Element Referencing	18
Element Hierarchy	18
Alphabetical List of Elements	21
access-log	21
admin-object-resource	22
admin-service	23
alert-service	24
alert-subscription	25

appliant-module	25
application-ref	26
applications	27
audit-module	27
auth-realm	28
availability-service	30
backend-principal	31
cluster	32
cluster-ref	33
clusters	33
config	34
configs	35
connection-pool	35
connector-connection-pool	36
connector-module	39
connector-resource	40
connector-service	41
custom-resource	42
das-config	43
description	45
domain	45
ejb-container	46
ejb-container-availability	48
ejb-module	50
ejb-timer-service	51
external-jndi-resource	52
filter-config	53
health-checker	54
http-access-log	54
http-file-cache	55
http-listener	56
http-protocol	57
http-service	58
iiop-listener	60
iiop-service	61
j2ee-application	61
jacc-provider	62
java-config	63
jdbc-connection-pool	65
jdbc-resource	68
jms-host	69
jms-service	70
jmx-connector	72

jvm-options	73
keep-alive	74
lb-config	74
lb-configs	75
lifecycle-module	76
listener-config	77
log-service	78
mail-resource	79
manager-properties	81
mdb-container	83
message-security-config	84
module-log-levels	85
module-monitoring-levels	87
monitoring-service	88
node-agent	88
node-agents	90
orb	90
persistence-manager-factory-resource	91
principal	92
profiler	92
property	93
provider-config	95
request-policy	96
request-processing	96
resource-adapter-config	97
resource-ref	98
resources	99
response-policy	100
security-map	101
security-service	101
server	103
server-ref	104
servers	105
session-config	105
session-manager	105
session-properties	106
ssl	107
ssl-client-config	108
store-properties	109
system-property	110
thread-pool	112
thread-pools	113
transaction-service	113

user-group	115
virtual-server	116
web-container	118
web-container-availability	119
web-module	122
Index	125

Preface

This *Administration Reference* provides information about the Sun Java™ System Application Server configuration file, `domain.xml`. This file contains most of the Sun Java System Application Server configuration.

Who Should Use This Book

This *Administration Reference* is intended for use by administrators and software developers who maintain and use Sun Java System servers and software. Application Server administrators and software developers should already understand the following technologies:

- Java technology
- The Java™ 2 Platform, Enterprise Edition (J2EE™ platform), version 1.4
- Extensible Markup Language (XML)

Before You Read This Book

Application Server is a component of Sun Java™ Enterprise System, a software infrastructure that supports enterprise applications distributed across a network or Internet environment. You should be familiar with the documentation provided with Sun Java Enterprise System, which can be accessed online at <http://docs.sun.com/app/docs/prod/entsys.05q1#hic>.

How This Book Is Organized

This book contains one chapter, “[The domain.xml File](#),” which describes in detail the structure and content of this file. The file’s elements are listed in alphabetical order.

Conventions Used in This Book

The tables in this section describe the conventions used in this book.

Typographic Conventions

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

Table 1 Typographic Conventions

Typeface	Meaning	Examples
AaBbCc123 (Monospace)	API and language elements, HTML tags, web site URLs, command names, file names, directory path names, onscreen computer output, sample code.	Edit your <code>.login</code> file. Use <code>ls -a</code> to list all files. <code>% You have mail.</code>
AaBbCc123 (Monospace bold)	What you type, when contrasted with onscreen computer output.	<code>% su</code> Password:
<i>AaBbCc123</i> (Italic)	Book titles, new terms, words to be emphasized. A placeholder in a command or path name to be replaced with a real name or value.	Read Chapter 6 in the <i>User’s Guide</i> . These are called <i>class</i> options. <i>Do not</i> save the file. The file is located in the <i>install-dir/bin</i> directory.

Symbols

The following table describes the symbol conventions used in this book.

Table 2 Symbol Conventions

Symbol	Description	Example	Meaning
[]	Contains optional 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.
-	Joins simultaneous multiple keystrokes.	Control-A	Press the Control key while you press the A key.
+	Joins consecutive multiple keystrokes.	Ctrl+A+N	Press the Control key, release it, and then press the subsequent keys.
>	Indicates menu item selection in a graphical user interface.	File > New > Templates	From the File menu, choose New. From the New submenu, choose Templates.

Default Paths and File Names

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

Table 3 Default Paths and File Names

Term	Description
<i>install_dir</i>	<p>By default, the Application Server installation directory is located here:</p> <ul style="list-style-type: none"> • Sun Java Enterprise System installations on the Solaris™ platform: <code>/opt/SUNWappserver/appserver</code> • Sun Java Enterprise System installations on the Linux platform: <code>/opt/sun/appserver/</code> • Other Solaris and Linux installations, non-root user: <code>user's home directory/SUNWappserver</code> • Other Solaris and Linux installations, root user: <code>/opt/SUNWappserver</code> • Windows, all installations: <code>SystemDrive:\Sun\AppServer</code>

Table 3 Default Paths and File Names

Term	Description
<i>domain_root_dir</i>	<p>By default, the directory containing all domains is located here:</p> <ul style="list-style-type: none"> • Sun Java Enterprise System installations on the Solaris platform: <i>/var/opt/SUNWappserver/domains/</i> • Sun Java Enterprise System installations on the Linux platform: <i>/var/opt/sun/appserver/domains/</i> • All other installations: <i>install_dir/domains/</i>
<i>domain_dir</i>	<p>By default, each domain directory is located here: <i>domain_root_dir/domain_dir</i></p> <p>In configuration files, you might see <i>domain_dir</i> represented as follows: <code>#{com.sun.aas.instanceRoot}</code></p>
<i>instance_dir</i>	<p>By default, each instance directory is located here: <i>domain_dir/instance_dir</i></p>

Shell Prompts

The following table describes the shell prompts used in this book.

Table 4 Shell Prompts

Shell	Prompt
C shell on UNIX or Linux	<i>machine-name%</i>
C shell superuser on UNIX or Linux	<i>machine-name#</i>
Bourne shell and Korn shell on UNIX or Linux	\$
Bourne shell and Korn shell superuser on UNIX or Linux	#
Windows command line	C:\

Related Documentation

The <http://docs.sun.com>SM web site enables you to access Sun technical documentation online. You can browse the archive or search for a specific book title or subject.

Books in This Documentation Set

The Sun Java System Application Server manuals are available as online files in Portable Document Format (PDF) and Hypertext Markup Language (HTML).

The following table summarizes the books included in the Application Server core documentation set.

Table 5 Books in This Documentation Set

Book Title	Description
<i>Release Notes</i>	Late-breaking information about the software and the documentation. Includes a comprehensive, table-based summary of the supported hardware, operating system, JDK, and JDBC/RDBMS.
<i>Quick Start Guide</i>	How to get started with the Sun Java System Application Server product.
<i>Installation Guide</i>	Installing the Sun Java System Application Server software and its components.
<i>Deployment Planning Guide</i>	Evaluating your system needs and enterprise to ensure that you deploy Sun Java System Application Server in a manner that best suits your site. General issues and concerns that you must be aware of when deploying an application server are also discussed.
<i>Developer's Guide</i>	Creating and implementing Java™ 2 Platform, Enterprise Edition (J2EE™ platform) applications intended to run on the Sun Java System Application Server that follow the open Java standards model for J2EE components and APIs. Includes general information about developer tools, security, assembly, deployment, debugging, and creating lifecycle modules.
<i>J2EE 1.4 Tutorial</i>	Using J2EE 1.4 platform technologies and APIs to develop J2EE applications and deploying the applications on the Sun Java System Application Server.
<i>Administration Guide</i>	Configuring, managing, and deploying the Sun Java System Application Server subsystems and components from the Administration Console.
<i>High Availability Administration Guide</i>	Post-installation configuration and administration instructions for the high-availability database.
<i>Administration Reference</i>	Editing the Sun Java System Application Server configuration file, <code>domain.xml</code> .
<i>Upgrade and Migration Guide</i>	Migrating your applications to the new Sun Java System Application Server programming model, specifically from Application Server 6.x and 7. This guide also describes differences between adjacent product releases and configuration options that can result in incompatibility with the product specifications.

Table 5 Books in This Documentation Set (*Continued*)

Book Title	Description
<i>Performance Tuning Guide</i>	Tuning the Sun Java System Application Server to improve performance.
<i>Troubleshooting Guide</i>	Solving Sun Java System Application Server problems.
<i>Error Message Reference</i>	Solving Sun Java System Application Server error messages.
<i>Reference Manual</i>	Utility commands available with the Sun Java System Application Server; written in manpage style. Includes the <code>asadmin</code> command line interface.

Other Server Documentation

For other server documentation, go to the following:

- Message Queue documentation
<http://docs.sun.com/db?p=prod/s1.s1msgqu>
- Directory Server documentation
http://docs.sun.com/coll/DirectoryServer_04q2
- Web Server documentation
http://docs.sun.com/coll/S1_websvr61_en

Accessing Sun Resources Online

For product downloads, professional services, patches and support, and additional developer information, go to the following:

- Download Center
<http://www.sun.com/software/download/>
- Professional Services
<http://www.sun.com/service/sunps/sunone/index.html>
- Sun Enterprise Services, Solaris Patches, and Support
<http://sunsolve.sun.com/>
- Developer Information
<http://developers.sun.com/prodtech/index.html>

Contacting Sun Technical Support

If you have technical questions about this product that are not answered in the product documentation, go to <http://www.sun.com/service/contacting>.

Related Third-Party Web Site References

For general information about DTD files and XML, see the XML specification at:

<http://www.w3.org/TR/REC-xml>

For a list of valid cache control response directives, see section 14.9 of the following document:

<http://www.ietf.org/rfc/rfc2616.txt>

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 document title and part number. The part number is a seven-digit or nine-digit number that can be found on the title page of the book or at the top of the document. For example, the title of this book is *Sun Java System Application Server 2005Q1 Administration Reference*, and the part number is 819-0221.

Sun Welcomes Your Comments

The domain.xml File

This chapter describes the `domain.xml` configuration file in these sections:

- [About the domain.xml File](#)
- [Alphabetical List of Elements](#)

About the domain.xml File

The `domain.xml` file contains most of the Sun Java™ System Application Server configuration. The encoding is UTF-8 to maintain compatibility with regular UNIX text editors. The `domain.xml` file is located in the domain configuration directory, which is typically `domain_dir/config`. This file is further described in the following sections:

- [The sun-domain_1_1.dtd File](#)
- [Default Values](#)
- [Variables](#)
- [Element Referencing](#)
- [Element Hierarchy](#)

The sun-domain_1_1.dtd File

The sun-domain_1_1.dtd file defines the structure of the domain.xml file, including the elements it can contain and the subelements and attributes these elements can have. The sun-domain_1_1.dtd file is located in the *install_dir/lib/dtds* directory.

NOTE Do not edit the sun-domain_1_1.dtd file; its contents change only with new versions of Sun Java System Application Server.

The sun-domain_1_1.dtd interface is unstable. An unstable interface might be experimental or transitional, and hence might change incompatibly, be removed, or be replaced by a more stable interface in the next release.

Elements or attributes that appear in the sun-domain_1_1.dtd file but are not described in this chapter are not implemented and should not be used.

For general information about DTD files and XML, see the XML specification at:

<http://www.w3.org/TR/REC-xml>

NOTE Subelements must be defined in the order in which they are listed under each **Subelements** heading in this chapter unless otherwise noted.

Default Values

In this manual, the term *default* is used in its broader sense, and not in the specific way it is used in the XML 1.0 standard. A default value is an initial value or the value used if no value is present in the XML file. A default value can be any of the following:

- A value supplied by the XML parser when no value is found in the domain.xml file. The relevant element or attribute is optional.
- A value supplied by the Sun Java System Application Server when no value is found in the domain.xml file and the XML parser doesn't provide a value. The relevant element or attribute is optional.
- An initial value supplied when the domain.xml file is created. The relevant element or attribute might or might not be optional.

Variables

Variables and variable references are needed for two reasons:

- Parts of the Application Server share much configuration information but differ in specific details. For example, server instances in a cluster typically share the same configuration except for their port numbers.
- Parts of the configuration come from the system environment but must still be captured in the configuration.

Variable references appear in the `domain.xml` file as strings that begin with the characters `${` and end with the character `}`. For example, the string `${com.sun.enterprise.myVar}` is a reference to the variable `com.sun.enterprise.myVar`.

Variables are defined both outside of and within `domain.xml`. Predefined variables that exist outside of `domain.xml` are defined as Java System Properties. Within `domain.xml`, a variable is defined using the `system-property` element or the `jvm-options` element.

The `system-property` element's `name` attribute is the name of a variable; its `value` attribute is the definition of the variable. For example, the following `system-property` element defines a `port-number` variable with the value `6500`:

```
<system-property name="port-number" value="6500" />
```

Multiple `system-property` subelements are permitted within `server`, `cluster`, `config`, and `domain` elements.

A variable defined in the `jvm-options` element is a Java System Property with the `-D` flag. For example, the following `jvm-options` element defines a `port-number` variable with the value `5500`:

```
<jvm-option>-Dport-number=5500</jvm-option>
```

Multiple definitions for the same variable are permitted. The Sun Java System Application Server determines the actual value of a variable by searching for its first definition in a strict hierarchy of the elements within `domain.xml`. The hierarchy is as follows:

```
server -> cluster -> config -> jvm-options -> domain -> System
```

Implicit in this hierarchy is the notion of reference and containment. A variable referenced in a `server` element is only looked up:

- In the `cluster` element referenced by that specific `server`
- In the `config` element that references that specific `server`
- In the `jvm-options` subelements of the `config` element referenced by that `server`

Element Referencing

One element *references* another when an attribute of the referencing element has the same value as an attribute of the referenced element. For example, the `application-ref` element references an application or module that is deployed to its parent `server` element. The `application-ref` element's `ref` attribute has the same value as the `name` attribute of a `lifecycle-module`, `j2ee-application`, `ejb-module`, `web-module`, `connector-module`, or `appclient-module` element.

The referencing `application-ref` element might look like this:

```
<application-ref ref="MyServlet"/>
```

The referenced `web-module` element might look like this:

```
<web-module name="MyServlet" location="myservletdir"/>
```

Element Hierarchy

The element hierarchy for the `domain.xml` file is as follows. To make the hierarchy more readable, elements having *property* as their last or only subelement are marked with a ^P, and the *property* subelements are not shown. Parent/child relationships between elements are shown, but not order or cardinality. For those details, see the element descriptions.

```
domain P
.  applications
.  .  lifecycle-module P
.  .  .  description
.  .  .  j2ee-application
.  .  .  .  description
.  .  .  web-module
.  .  .  .  description
.  .  .  ejb-module
.  .  .  .  description
.  .  .  connector-module
.  .  .  .  description
.  .  .  appclient-module
.  .  .  .  description
.  resources
.  .  custom-resource P
.  .  .  description
.  .  .  external-jndi-resource P
.  .  .  .  description
.  .  .  jdbc-resource P
.  .  .  .  description
```

```

. . mail-resource P
. . . description
. . persistence-manager-factory-resource P
. . . description
. . admin-object-resource P
. . . description
. . connector-resource P
. . . description
. . resource-adapter-config P
. . jdbc-connection-pool P
. . . description
. . connector-connection-pool P
. . . description
. . . security-map
. . . . principal
. . . . user-group
. . . . backend-principal
. configs
. . config P
. . . http-service P
. . . . access-log
. . . . request-processing
. . . . keep-alive
. . . . connection-pool
. . . . http-protocol
. . . . http-file-cache
. . . . http-listener P
. . . . . ssl
. . . . . virtual-server P
. . . . . http-access-log
. . . iiop-service
. . . . orb P
. . . . . ssl-client-config
. . . . . . ssl
. . . . . iiop-listener P
. . . . . . ssl
. . . . admin-service P
. . . . . jmx-connector P
. . . . . . ssl
. . . . . das-config P
. . . . connector-service
. . . . web-container P
. . . . . session-config
. . . . . . session-manager
. . . . . . . manager-properties P

```

- store-properties ^P
- session-properties ^P
- ejb-container ^P
- ejb-timer-service ^P
- mdb-container ^P
- jms-service ^P
- jms-host ^P
- log-service ^P
- module-log-levels ^P
- security-service ^P
- auth-realm ^P
- jacc-provider ^P
- audit-module ^P
- message-security-config
- provider-config ^P
- request-policy
- response-policy
- transaction-service ^P
- monitoring-service ^P
- module-monitoring-levels ^P
- java-config ^P
- profiler ^P
- jvm-options
- jvm-options
- availability-service ^P
- web-container-availability ^P
- ejb-container-availability ^P
- thread-pools
- thread-pool
- alert-service ^P
- alert-subscription
- listener-config ^P
- filter-config ^P
- system-property
- description
- . servers
- . . server ^P
- . . . application-ref
- . . . resource-ref
- . . . system-property
- description
- . clusters
- . . cluster ^P
- . . . server-ref
- health-checker

```

. . . resource-ref
. . . application-ref
. . . system-property
. . . . description
. node-agents
. . node-agent P
. . . jmx-connector P
. . . . ssl
. . . auth-realm P
. . . log-service P
. . . . module-log-levels P
. lb-configs
. . lb-config P
. . . cluster-ref
. . . . health-checker
. . . server-ref
. . . . health-checker
. system-property
. . description

```

Alphabetical List of Elements

A B C D E F H I J K L M N O P R S T U V W

A

access-log

Defines access log settings for each [http-access-log](#) subelement of each [virtual-server](#).

Superelements

[http-service](#)

Subelements

none

Attributes

The following table describes attributes for the `access-log` element.

Table 1-1 access-log Attributes

Attribute	Default	Description
format	%client.dns;%auth-user-name;%system.date;%request;%status;%response.length;	(optional) Specifies the format of the access log.
rotation-policy	time	(optional) Specifies the condition that triggers log rotation. The only legal value is <code>time</code> , which rotates log files at the <code>rotation-interval-in-minutes</code> interval.
rotation-interval-in-minutes	1440	(optional) Specifies the time interval between log rotations if <code>rotation-policy</code> is set to <code>time</code> .
rotation-suffix	%YYYY;%MM;%DD;-%hh;h%mm;m%ss;s	(optional) Specifies the format of the timestamp appended to the access log name when log rotation occurs.
rotation-enabled	true	(optional) If <code>true</code> , enables log rotation.

admin-object-resource

Defines an administered object for an inbound resource adapter.

Superelements

[resources](#)

Subelements

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

Table 1-2 admin-object-resource Subelements

Element	Required	Description
description	zero or one	Contains a text description of this element.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `admin-object-resource` element.

Table 1-3 admin-object-resource Attributes

Attribute	Default	Description
jndi-name	none	Specifies the JNDI name for the resource.
res-type	none	Specifies the fully qualified type of the resource.
res-adapter	none	Specifies the name of the inbound resource adapter, as specified in the name attribute of a <code>connector-module</code> element.
object-type	user	(optional) Defines the type of the resource. Allowed values are: <ul style="list-style-type: none"> system-all - A system resource for all server instances and the domain application server. system-admin - A system resource only for the the domain application server. system-instance - A system resource for all server instances only. user - A user resource.
enabled	true	(optional) Determines whether this resource is enabled at runtime.

Properties

Properties of the `admin-object-resource` element are the names of setter methods of the `adminobject-class` specified in the `adminobject` element of the `ra.xml` file. Some of the property names can be specified in the `adminobject` element itself. For example, in `jmsra`, the resource adapter used to communicate with the Sun Java system Message Queue software, `jmsra`, `Name` and `Description` are valid properties.

For a complete list of the available properties (called *administered object attributes* in Sun Java System Message Queue), see the *Sun Java System Message Queue Administration Guide*.

admin-service

Determines whether the server instance is a regular instance, a domain administration server, or a combination.

Superelements

[config](#)

Subelements

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

Table 1-4 `admin-service` Subelements

Element	Required	Description
jmx-connector	zero or more	Configures a JSR 160 compliant remote JMX connector.
das-config	zero or one	Defines a domain administration server configuration.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `admin-service` element.

Table 1-5 `admin-service` Attributes

Attribute	Default	Description
<code>type</code>	<code>server</code>	Specifies whether the server instance is a regular instance (<code>server</code>), a domain administration server (<code>das</code>), or a combination (<code>das-and-server</code>).
<code>system-jmx-connector-name</code>	<code>none</code>	Specifies the name of the internal jmx-connector .

alert-service

Configures the alert service, which allows you to register for and receive system status alerts.

Superelements

[config](#)

Subelements

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

Table 1-6 `alert-service` Subelements

Element	Required	Description
alert-subscription	zero or more	Configures a subscription to system status alerts.
property	zero or more	Specifies a property or a variable.

alert-subscription

Configures a subscription to system status alerts.

Superelements

[alert-service](#)

Subelements

The following table describes subelements for the `alert-subscription` element.

Table 1-7 `alert-subscription` Subelements

Element	Required	Description
listener-config	only one	Configures the listener class that listens for alerts from notification emitters.
filter-config	zero or one	Configures the filter class that filters alerts from notification emitters.

Attributes

The following table describes attributes for the `alert-subscription` element.

Table 1-8 `alert-subscription` Attributes

Attribute	Default	Description
<code>name</code>	none	Specifies the name of this alert subscription.

appclient-module

Specifies a deployed application client container (ACC) module.

Superelements

[applications](#)

Subelements

The following table describes subelements for the `appclient-module` element.

Table 1-9 `appclient-module` Subelements

Element	Required	Description
description	zero or one	Contains a text description of this element.

Attributes

The following table describes attributes for the `appclient-module` element.

Table 1-10 `appclient-module` Attributes

Attribute	Default	Description
<code>name</code>	<code>none</code>	The name of the ACC module.
<code>location</code>	<code>none</code>	The location of the ACC module in the Sun Java System Application Server file system.
<code>directory-deployed</code>	<code>false</code>	(optional) Specifies whether the application has been deployed to a directory.

application-ref

References an application or module deployed to the server instance.

Superelements

[cluster](#), [server](#)

Subelements

`none`

Attributes

The following table describes attributes for the `application-ref` element.

Table 1-11 `application-ref` Attributes

Attribute	Default	Description
<code>enabled</code>	<code>true</code>	(optional) Determines whether the application or module is enabled.
<code>virtual-servers</code>	<code>all virtual servers</code>	(optional) In a comma-separated list, references <code>id</code> attributes of the virtual-server elements to which the web-module or the web modules within this j2ee-application are deployed.
<code>lb-enabled</code>	<code>false</code>	(optional) If <code>true</code> , all load-balancers consider this application available to them.
<code>disable-timeout-in-minutes</code>	<code>30</code>	(optional) Specifies the time it takes this application to reach a quiescent state after having been disabled.
<code>ref</code>	<code>none</code>	References the <code>name</code> attribute of a lifecycle-module , j2ee-application , ejb-module , web-module , connector-module , or appclient-module element.

applications

Contains deployed J2EE applications, J2EE modules, and Lifecycle modules.

Superelements

[domain](#)

Subelements

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

Table 1-12 `applications` Subelements

Element	Required	Description
lifecycle-module	zero or more	Specifies a deployed lifecycle module.
j2ee-application	zero or more	Specifies a deployed J2EE application.
ejb-module	zero or more	Specifies a deployed EJB module.
web-module	zero or more	Specifies a deployed web module.
connector-module	zero or more	Specifies a deployed connector module.
appclient-module	zero or more	Specifies a deployed application client container (ACC) module.

NOTE Subelements of an `applications` element can occur in any order.

audit-module

Specifies an optional plug-in module that implements audit capabilities.

Superelements

[security-service](#)

Subelements

The following table describes subelements for the `audit-module` element.

Table 1-13 `audit-module` Subelements

Element	Required	Description
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `audit-module` element.

Table 1-14 `audit-module` Attributes

Attribute	Default	Description
<code>name</code>	<code>none</code>	Specifies the name of this audit module.
<code>classname</code>	<code>none</code>	Specifies the Java class that implements this audit module.

auth-realm

Defines a realm for authentication.

Authentication realms require provider-specific properties, which vary depending on what a particular implementation needs.

For more information about how to define realms, see the *Sun Java System Application Server Developer's Guide*.

Here is an example of the default `file` realm:

```
<auth-realm name="file"
  classname="com.iplanet.ias.security.auth.realm.file.FileRealm">
  <property name="file" value="domain_dir/config/keyfile"/>
  <property name="jaas-context" value="fileRealm"/>
</auth-realm>
```

Which properties an `auth-realm` element uses depends on the value of the `auth-realm` element's `name` attribute. The `file` realm uses `file` and `jaas-context` properties. Other realms use different properties.

Superelements

[node-agent](#), [security-service](#)

Subelements

The following table describes subelements for the `auth-realm` element.

Table 1-15 `auth-realm` Subelements

Element	Required	Description
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `auth-realm` element.

Table 1-16 `auth-realm` Attributes

Attribute	Default	Description
<code>name</code>	<code>none</code>	Specifies the name of this realm.
<code>classname</code>	<code>none</code>	Specifies the Java class that implements this realm.

Properties

The standard realms provided with Sun Java System Application Server have required and optional properties. A custom realm might have different properties.

The following table describes properties for the `auth-realm` element.

Table 1-17 `auth-realm` Properties

Property	Realms	Description
<code>jaas-context</code>	<code>file</code> , <code>ldap</code> , <code>solaris</code>	Specifies the JAAS (Java Authentication and Authorization Service) context.
<code>file</code>	<code>file</code>	Specifies the file that stores user names. The default is <code>domain_dir/config/keyfile</code> .
<code>assign-groups</code>	<code>certificate</code>	(optional) If this property is set, its value is taken to be a comma-separated list of group names. All clients who present valid certificates are assigned membership to these groups for the purposes of authorization decisions in the web and EJB containers.
<code>directory</code>	<code>ldap</code>	Specifies the LDAP URL to your server.
<code>base-dn</code>	<code>ldap</code>	Specifies the LDAP base DN for the location of user data. This base DN can be at any level above the user data, since a tree scope search is performed. The smaller the search tree, the better the performance.
<code>search-filter</code>	<code>ldap</code>	(optional) Specifies the search filter to use to find the user. The default is <code>uid=%s</code> (<code>%s</code> expands to the subject name).
<code>group-base-dn</code>	<code>ldap</code>	(optional) Specifies the base DN for the location of groups data. By default, it is same as the <code>base-dn</code> , but it can be tuned, if necessary.
<code>group-search-filter</code>	<code>ldap</code>	(optional) Specifies the search filter to find group memberships for the user. The default is <code>uniquemember=%d</code> (<code>%d</code> expands to the user element DN).
<code>group-target</code>	<code>ldap</code>	(optional) Specifies the LDAP attribute name that contains group name entries. The default is <code>CN</code> .

Table 1-17 `auth-realm` Properties (Continued)

Property	Realms	Description
<code>search-bind-dn</code>	<code>ldap</code>	(optional) Specifies an optional DN used to authenticate to the directory for performing the <code>search-filter</code> lookup. Only required for directories that do not allow anonymous search.
<code>search-bind-password</code>	<code>ldap</code>	(optional) Specifies the LDAP password for the DN given in <code>search-bind-dn</code> .

availability-service

Configures the availability service. Enables high-availability features, such as HTTP session and stateful session bean state persistence to the Sun Java System high-availability database (HADB).

Availability can be enabled or disabled at the following levels:

1. The server instance (attribute of `availability-service`). Default is `true` (enabled).
2. The EJB or web container (attribute of `ejb-container-availability` or `web-container-availability`). Default is `true` (enabled).
3. The application (attribute of `j2ee-application`). Default is `false` (disabled).
4. The stand-alone EJB or web module (attribute of `ejb-module` or `web-module`). Default is `false` (disabled).
5. The stateful session bean. Default is `false` (disabled). See the *Sun Java System Application Server Developer's Guide*.

For availability to be enabled at a given level, it must be enabled at all higher levels, as well. For example, to enable availability at the application level, you must also enable it at the server instance and container levels.

Superelements

[config](#)

Subelements

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

Table 1-18 `availability-service` Subelements

Element	Required	Description
web-container-availability	only one	Enables availability in the web container.
ejb-container-availability	only one	Enables availability in the EJB container.

Table 1-18 availability-service Subelements (*Continued*)

Element	Required	Description
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `availability-service` element.

Table 1-19 availability-service Attributes

Attribute	Default	Description
<code>availability-enabled</code>	<code>true</code>	(optional) If set to <code>true</code> , high-availability features apply to all applications deployed to the server instance that do not have availability disabled. All instances in a cluster should have the same availability value to ensure consistent behavior.
<code>store-pool-name</code>	<code>jdbc/hastore</code>	(optional) Specifies the <code>jndi-name</code> of the <code>jdbc-resource</code> used for connections to the HADB for session persistence. For more information about setting up a connection pool and JDBC resource for the HADB, see the description of the <code>configure-ha-cluster</code> command in the <i>Sun Java System Application Server Reference Manual</i> .

B**backend-principal**

Specifies the user name and password required by the EIS.

Superelements

[security-map](#)

Subelements

none

Attributes

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

Table 1-20 backend-principal Attributes

Attribute	Default	Description
<code>user-name</code>	none	Specifies the user name required by the EIS.

Table 1-20 backend-principal Attributes (*Continued*)

Attribute	Default	Description
password	none	Specifies the password required by the EIS.

C

cluster

Defines a cluster.

Superelements

[clusters](#)

Subelements

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

Table 1-21 cluster Subelements

Element	Required	Description
server-ref	zero or more	References a server instance that belongs to the cluster.
resource-ref	zero or more	References a resource deployed to the cluster.
application-ref	zero or more	References an application or module deployed to the cluster.
system-property	zero or more	Specifies a system property.
property	zero or more	Specifies a property or a variable.

Attributes

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

Table 1-22 cluster Attributes

Attribute	Default	Description
name	none	Specifies the name of the cluster.
config-ref	default config element's name attribute value, server-config	References the configuration used by the cluster.

cluster-ref

References a cluster.

Superelements

[lb-config](#)

Subelements

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

Table 1-23 `cluster-ref` Subelements

Element	Required	Description
health-checker	zero or one	Defines a health checker for the referenced cluster.

Attributes

The following table describes attributes for the `cluster-ref` element.

Table 1-24 `cluster-ref` Attributes

Attribute	Default	Description
<code>ref</code>	none	References the name attribute of a cluster element.

clusters

Contains clusters.

Superelements

[domain](#)

Subelements

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

Table 1-25 `clusters` Subelements

Element	Required	Description
cluster	zero or more	Defines a cluster.

config

Defines a configuration, which is a collection of settings that controls how a server instance functions.

Superelements

[configs](#)

Subelements

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

Table 1-26 `config` Subelements

Element	Required	Description
http-service	only one	Configures the HTTP service.
iiop-service	only one	Configures the IIOP service.
admin-service	only one	Determines whether the server to which the configuration applies is an administration server.
connector-service	zero or one	Configures the connector service.
web-container	only one	Configures the web container.
ejb-container	only one	Configures the Enterprise JavaBeans™ (EJB™) container.
mdb-container	only one	Configures the message-driven bean (MDB) container.
jms-service	zero or one	Configures the Java™ Message Service (JMS) provider.
log-service	only one	Configures the system logging service.
security-service	only one	Configures the J2EE security service.
transaction-service	only one	Configures the transaction service.
monitoring-service	only one	Configures the monitoring service.
java-config	only one	Configures the Java™ Virtual Machine (JVM™).
availability-service	zero or one	Configures the availability service.
thread-pools	only one	Configures thread pools.
alert-service	zero or one	Configures the alert service.
system-property	zero or more	Specifies a system property.
property	zero or more	Specifies a property or a variable.

Attributes

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

Table 1-27 `config` Attributes

Attribute	Default	Description
<code>name</code>	<code>server-config</code> (for default instance)	Specifies the name of the configuration.
<code>dynamic-reconfiguration-enabled</code>	<code>true</code>	(optional) If <code>true</code> , any changes to the system (for example, applications deployed, resources created) are automatically applied to the affected servers without a restart being required. If <code>false</code> , such changes are only picked up by the affected servers when each server restarts.

configs

Contains configurations.

Superelements

[domain](#)

Subelements

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

Table 1-28 `configs` Subelements

Element	Required	Description
config	one or more	Defines a configuration.

connection-pool

Defines a pool of client HTTP connections.

Superelements

[http-service](#)

Subelements

`none`

Attributes

The following table describes attributes for the `connection-pool` element.

Table 1-29 `connection-pool` Attributes

Attribute	Default	Description
<code>queue-size-in-bytes</code>	4096	(optional) Specifies the number of outstanding connections an <code>http-listener</code> can have.
<code>max-pending-count</code>	4096	(optional) Specifies the maximum number of pending connections on an <code>http-listener</code> .
<code>receive-buffer-size-in-bytes</code>	4096	(optional) Specifies the size of the receive buffer for all <code>http-listener</code> elements.
<code>send-buffer-size-in-bytes</code>	8092	(optional) Specifies the size of the send buffer for all <code>http-listener</code> elements.

connector-connection-pool

Defines a connector connection pool.

Superelements

[resources](#)

Subelements

The following table describes subelements for the `connector-connection-pool` element.

Table 1-30 `connector-connection-pool` Subelements

Element	Required	Description
description	zero or one	Contains a text description of this element.
security-map	zero or more	Maps the principal received during servlet or EJB authentication to the credentials accepted by the EIS.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `connector-connection-pool` element.

Table 1-31 connector-connection-pool Attributes

Attribute	Default	Description
name	none	Specifies the name of the connection pool. A <code>connector-resource</code> element's <code>pool-name</code> attribute refers to this name.
resource-adapter-name	none	Specifies the name attribute of the deployed <code>connector-module</code> . If no name is specified during deployment, the name of the <code>.rar</code> file is used. If the resource adapter is embedded in an application, then it is <code>app_name#rar_name</code> .
connection-definition-name	none	Specifies a unique name, identifying a resource adapter's <code>connection-definition</code> element in the <code>ra.xml</code> file. This is usually the <code>connectionfactory-interface</code> of the <code>connection-definition</code> element.
steady-pool-size	8	(optional) Specifies the initial and minimum number of connections maintained in the pool.
max-pool-size	32	(optional) Specifies the maximum number of connections that can be created to satisfy client requests.
max-wait-time-in-millis	60000	(optional) Specifies the amount of time, in milliseconds, that the caller is willing to wait for a connection. If 0, the caller is blocked indefinitely until a resource is available or an error occurs.
pool-resize-quantity	2	(optional) Specifies the number of connections to be destroyed if the existing number of connections is above the <code>steady-pool-size</code> (subject to the <code>max-pool-size</code> limit). This is enforced periodically at the <code>idle-time-out-in-seconds</code> interval. An idle connection is one that has not been used for a period of <code>idle-time-out-in-seconds</code> .
idle-timeout-in-seconds	300	(optional) Specifies the maximum time that a connection can remain idle in the pool. After this amount of time, the pool can close this connection.
fail-all-connections	false	(optional) If <code>true</code> , closes all connections in the pool if a single validation check fails.

Table 1-31 connector-connection-pool Attributes (*Continued*)

Attribute	Default	Description
transaction-support	none	(optional) Specifies the transaction support for this connection pool. Overrides the transaction support defined in the resource adapter in a downward compatible way: supports a transaction level lower than or equal to the resource adapter's, but not higher. Allowed values in descending order are: <ul style="list-style-type: none"> • <code>XATransaction</code> - Supports distributed transactions. • <code>LocalTransaction</code> - Supports local transactions only. • <code>NoTransaction</code> - No transaction support.

Properties

Properties of the `connector-connection-pool` element are the names of setter methods of the `managedconnectionfactory-class` element in the `ra.xml` file. Properties of this element override the `ManagedConnectionFactory` JavaBean configuration settings.

The following table describes the `connector-connection-pool` properties of `jmsra`, the resource adapter used to communicate with the Sun Java System Message Queue software. For a complete list of the available properties (called *administered object attributes* in Sun Java System Message Queue), see the *Sun Java System Message Queue Administration Guide*.

Table 1-32 connector-connection-pool Properties

Property	Default	Description
AddressList	none	Specifies a list of host/port combinations of the Sun Java System Message Queue. For JMS resources of the Type <code>javax.jms.TopicConnectionFactory</code> or <code>javax.jms.QueueConnectionFactory</code> .
ClientId	none	Specifies the JMS Client Identifier to be associated with a <code>Connection</code> created using the <code>createTopicConnection</code> method of the <code>TopicConnectionFactory</code> class. For JMS resources of the Type <code>javax.jms.TopicConnectionFactory</code> . Durable subscription names are unique and only valid within the scope of a client identifier. To create or reactivate a durable subscriber, the connection must have a valid client identifier. The JMS specification ensures that client identifiers are unique and that a given client identifier is allowed to be used by only one active connection at a time.

Table 1-32 connector-connection-pool Properties (*Continued*)

Property	Default	Description
UserName	guest	Specifies the user name for connecting to the Sun Java System Message Queue. For JMS resources of the Type <code>javax.jms.TopicConnectionFactory</code> or <code>javax.jms.QueueConnectionFactory</code> .
Password	guest	Specifies the password for connecting to the Sun Java System Message Queue. For JMS resources of the Type <code>javax.jms.TopicConnectionFactory</code> or <code>javax.jms.QueueConnectionFactory</code> .
ReconnectAttempts	6	Specifies the number of attempts to connect (or reconnect) for each address in the <code>imqAddressList</code> before the client runtime moves on to try the next address in the list. A value of -1 indicates that the number of reconnect attempts is unlimited (the client runtime attempts to connect to the first address until it succeeds).
ReconnectInterval	30000	Specifies the interval between reconnect attempts in milliseconds. This applies to attempts on each address in the <code>imqAddressList</code> and on successive addresses in the list. If too short, this time interval does not give a broker time to recover. If too long, the reconnect might represent an unacceptable delay.
ReconnectEnabled	false	If true, specifies that the client runtime attempts to reconnect to a message server (or the list of addresses in <code>imqAddressList</code>) when a connection is lost.
AddressListBehavior	priority	Specifies whether connection attempts are in the order of addresses in the <code>imqAddressList</code> attribute (<code>priority</code>) or in a random order (<code>random</code>). If many clients are attempting a connection using the same connection factory, use a random order to prevent them from all being connected to the same address.
AddressListIterations	-1	Specifies the number of times the client runtime iterates through the <code>imqAddressList</code> in an effort to establish (or re-establish) a connection. A value of -1 indicates that the number of attempts is unlimited.

NOTE All JMS administered object resource properties that worked with version 7 of the Application Server are supported for backward compatibility.

connector-module

Specifies a deployed connector module.

Superelements

[applications](#)

Subelements

The following table describes subelements for the `connector-module` element.

Table 1-33 `connector-module` Subelements

Element	Required	Description
description	zero or one	Contains a text description of this element.

Attributes

The following table describes attributes for the `connector-module` element.

Table 1-34 `connector-module` Attributes

Attribute	Default	Description
<code>name</code>	name of the .rar file	The name of the connector module.
<code>location</code>	none	The location of the connector module in the Sun Java System Application Server file system.
<code>object-type</code>	user	(optional) Defines the type of the resource. Allowed values are: <ul style="list-style-type: none"> <code>system-all</code> - A system resource for all server instances and the domain application server. <code>system-admin</code> - A system resource only for the the domain application server. <code>system-instance</code> - A system resource for all server instances only. <code>user</code> - A user resource.
<code>enabled</code>	true	(optional) Determines whether the connector module is enabled.
<code>directory-deployed</code>	false	(optional) Specifies whether the application has been deployed to a directory.

connector-resource

Defines the connection factory object of a specific connection definition in a connector (resource adapter).

Superelements

[resources](#)

Subelements

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

Table 1-35 `connector-resource` Subelements

Element	Required	Description
<code>description</code>	zero or one	Contains a text description of this element.
<code>property</code>	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `connector-resource` element.

Table 1-36 `connector-resource` Attributes

Attribute	Default	Description
<code>jndi-name</code>	none	Specifies the JNDI name for the resource.
<code>pool-name</code>	none	Specifies the name of the associated connector connection pool, defined in a <code>connector-connection-pool</code> element.
<code>object-type</code>	user	(optional) Defines the type of the resource. Allowed values are: <ul style="list-style-type: none"> <code>system-all</code> - A system resource for all server instances and the domain application server. <code>system-admin</code> - A system resource only for the the domain application server. <code>system-instance</code> - A system resource for all server instances only. <code>user</code> - A user resource.
<code>enabled</code>	true	(optional) Determines whether this resource is enabled at runtime.

connector-service

Configures the connector service.

Superelements

`config`

Subelements

none

Attributes

The following table describes attributes for the `connector-service` element.

Table 1-37 connector-service Attributes

Attribute	Default	Description
shutdown-timeout -in-seconds	30	(optional) Specifies the maximum time allowed during application server shutdown for the <code>ResourceAdapter.stop()</code> method of a connector module's instance to complete. Resource adapters that take longer to shut down are ignored, and application server shutdown continues.

custom-resource

Defines a custom resource, which specifies a custom server-wide resource object factory. Such object factories implement the `javax.naming.spi.ObjectFactory` interface.

Superelements

[resources](#)

Subelements

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

Table 1-38 custom-resource Subelements

Element	Required	Description
description	zero or one	Contains a text description of this element.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `custom-resource` element.

Table 1-39 custom-resource Attributes

Attribute	Default	Description
jndi-name	none	Specifies the JNDI name for the resource.
res-type	none	Specifies the fully qualified type of the resource.
factory-class	none	Specifies the fully qualified name of the user-written factory class, which implements <code>javax.naming.spi.ObjectFactory</code> .

Table 1-39 custom-resource Attributes (*Continued*)

Attribute	Default	Description
object-type	user	(optional) Defines the type of the resource. Allowed values are: <ul style="list-style-type: none"> system-all - A system resource for all server instances and the domain application server. system-admin - A system resource only for the the domain application server. system-instance - A system resource for all server instances only. user - A user resource.
enabled	true	(optional) Determines whether this resource is enabled at runtime.

D

das-config

Defines a domain administration server configuration. The domain administration server runs the Administration Console.

Superelements

[admin-service](#)

Subelements

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

Table 1-40 das-config Subelements

Element	Required	Description
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `das-config` element. For more information about deployment topics such as dynamic reloading and autodeployment, see the *Sun Java System Application Server Developer's Guide*.

Table 1-41 das-config Attributes

Attribute	Default	Description
dynamic-reload-enabled	false	(optional) If <code>true</code> , checks the timestamp on a <code>.reload</code> file at every module and application directory level, to trigger dynamic reloading.
dynamic-reload-poll-interval-in-seconds	2	(optional) Controls the polling frequency of dynamic reloading.
autodeploy-enabled	false	(optional) If <code>true</code> , enables autodeployment, which lets you quickly deploy applications and modules to a running application server without performing an explicit server instance restart or a separate deployment operation.
autodeploy-polling-interval-in-seconds	2	(optional) Controls the polling frequency of autodeployment.
autodeploy-dir	autodeploy	(optional) Specifies the source directory (absolute or relative to <code>domain_dir</code>) in which autodeployment looks for deployable components.
autodeploy-verifier-enabled	false	(optional) If <code>true</code> , the verifier is run before autodeployment. If verification fails, deployment is not performed.
autodeploy-jsp-precompilation-enabled	false	(optional) If <code>true</code> , JSP pages are precompiled during autodeployment.
deploy-xml-validation	full	(optional) Specifies the type of XML validation performed on standard and Sun Java System Application Server deployment descriptors: <ul style="list-style-type: none"> • <code>full</code> - If XML validation fails, deployment fails. • <code>parsing</code> - XML validation errors are reported but deployment occurs. • <code>none</code> - No XML validation is performed.
admin-session-timeout-in-minutes	sun-web.xml timeoutSeconds property value or web.xml session-timeout attribute value	(optional) Specifies the Administration Console timeout.

description

Contains a text description of the parent element.

Superelements

[admin-object-resource](#), [appclient-module](#), [connector-connection-pool](#), [connector-module](#), [connector-resource](#), [custom-resource](#), [ejb-module](#), [external-jndi-resource](#), [j2ee-application](#), [jdbc-connection-pool](#), [jdbc-resource](#), [lifecycle-module](#), [mail-resource](#), [persistence-manager-factory-resource](#), [property](#), [system-property](#), [web-module](#)

Subelements

none - contains data

domain

Defines a domain. This is the root element; there can only be one domain element in a domain.xml file.

Superelements

none

Subelements

The following table describes subelements for the domain element.

Table 1-42 domain Subelements

Element	Required	Description
applications	zero or one	Contains deployed J2EE applications, J2EE modules, and lifecycle modules.
resources	zero or one	Contains configured resources.
configs	only one	Contains configurations.
servers	only one	Contains server instances.
clusters	zero or one	Contains clusters.
node-agents	zero or one	Contains node agents.
lb-configs	zero or one	Contains load balancing configurations.
system-property	zero or more	Specifies a system property.
property	zero or more	Specifies a property or a variable.

Attributes

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

Table 1-43 `domain` Attributes

Attribute	Default	Description
<code>application-root</code>	<code>domain_dir/applications</code>	(optional) Specifies the absolute path where deployed applications reside for this domain.
<code>log-root</code>	<code>domain_dir/logs</code>	(optional) Specifies where the domain's log files are kept. The directory in which the log is kept must be writable by whatever user account the server runs as. See the log-service description for details about logs.
<code>locale</code>	operating system default	(optional) Specifies the domain's language.

E

`ejb-container`

Configures the EJB container. Stateless session beans are maintained in pools. Stateful session beans have session affinity and are cached. Entity beans associated with a database primary key are also cached. Entity beans not yet associated with a primary key are maintained in pools. Pooled entity beans are used to run `ejbCreate()` and finder methods.

Superelements

[config](#)

Subelements

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

Table 1-44 `ejb-container` Subelements

Element	Required	Description
ejb-timer-service	zero or one	Configures the EJB timer service.
property	zero or more	Specifies a property or a variable.

Attributes

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

Table 1-45 `ejb-container` Attributes

Attribute	Default	Description
<code>steady-pool-size</code>	32	<p>(optional) Specifies the initial and minimum number of beans maintained in the pool. Must be 0 or greater and less than <code>max-pool-size</code>.</p> <p>Bean instances are removed from the pool and returned after use. The pool is replenished or cleaned up periodically to maintain this size.</p> <p>Applies to stateless session beans and entity beans.</p>
<code>pool-resize-quantity</code>	16	<p>(optional) Specifies the number of beans to be removed when the <code>pool-idle-timeout-in-seconds</code> timer expires. A cleaner thread removes any unused instances.</p> <p>Must be 0 or greater and less than <code>max-pool-size</code>. The pool is not resized below the <code>steady-pool-size</code>.</p> <p>Applies to stateless session beans and entity beans.</p>
<code>max-pool-size</code>	64	<p>(optional) Specifies the maximum number of beans that can be created to satisfy client requests. A value of 0 indicates an unbounded pool.</p> <p>Applies to stateless session beans and entity beans.</p>
<code>cache-resize-quantity</code>	32	<p>(optional) Specifies the number of beans to be:</p> <ul style="list-style-type: none"> created if a request arrives when the pool has no available beans (subject to the <code>max-cache-size</code> limit) passivated when the <code>cache-idle-timeout-in-seconds</code> timer expires and a cleaner thread removes any unused instances, or when the cache size exceeds <code>max-cache-size</code>. <p>Must be greater than 1 and less than <code>max-cache-size</code>.</p> <p>Applies to stateful session beans and entity beans.</p>
<code>max-cache-size</code>	512	<p>(optional) Specifies the maximum number of beans in the cache. A value of 0 indicates an unbounded cache.</p> <p>Applies to stateful session beans and entity beans.</p>
<code>pool-idle-timeout-in-seconds</code>	600	<p>(optional) Specifies the maximum time that a bean can remain idle in the pool. After this amount of time, the pool can remove this bean. A value of 0 specifies that idle beans can remain in the pool indefinitely.</p> <p>Applies to stateless session beans and entity beans.</p>

Table 1-45 `ejb-container` Attributes (*Continued*)

Attribute	Default	Description
<code>cache-idle-timeout-in-seconds</code>	600	(optional) 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. Applies to stateful session beans and entity beans.
<code>removal-timeout-in-seconds</code>	5400	(optional) Specifies the amount of time that a bean can remain passivated before it is removed from the session store. A value of 0 specifies that the container does not remove inactive beans automatically. If <code>removal-timeout-in-seconds</code> is less than or equal to <code>cache-idle-timeout-in-seconds</code> , beans are removed immediately without being passivated. The <code>session-store</code> attribute of the <code>server</code> element determines the location of the session store. Applies to stateful session beans.
<code>victim-selection-policy</code>	<code>nru</code>	(optional) Specifies how stateful session beans are selected for passivation. Allowed values are <code>fifo</code> , <code>lru</code> , and <code>nru</code> : <ul style="list-style-type: none"> <code>fifo</code> - Selects the oldest instance. <code>lru</code> - Selects the least recently accessed instance. <code>nru</code> - Selects a not recently used instance.
<code>commit-option</code>	<code>B</code>	(optional) Determines which commit option is used for entity beans. Legal values are <code>B</code> or <code>C</code> .
<code>session-store</code>	<code>domain_dir</code> <code>/session-store</code>	(optional) Specifies the directory where passivated stateful session beans and persisted HTTP sessions are stored in the file system.

ejb-container-availability

Enables availability in the EJB container, including stateful session bean (SFSB) state persistence to the high-availability database (HADB).

Superelements

[availability-service](#)

Subelements

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

Table 1-46 `ejb-container-availability` Subelements

Element	Required	Description
<code>property</code>	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `ejb-container-availability` element.

Table 1-47 `ejb-container-availability` Attributes

Attribute	Default	Description
<code>availability-enabled</code>	<code>true</code>	(optional) If set to <code>true</code> , and if availability is enabled for the server instance (see <code>availability-service</code>), high-availability features apply to all SFSBs deployed to the server instance that do not have availability disabled. All instances in a cluster should have the same availability value to ensure consistent behavior.
<code>sfsb-ha-persistence-type</code>	<code>ha</code>	(optional) Specifies the session persistence and passivation mechanism for SFSBs that have availability enabled. Allowed values are <code>file</code> (the file system) and <code>ha</code> (the HADB). For production environments that require session persistence, use <code>ha</code> . If set to <code>file</code> , the <code>ejb-container</code> element's <code>session-store</code> attribute specifies the file system location where the passivated session bean state is stored. Checkpointing to the file system is useful for internal testing but is not supported for production environments.
<code>sfsb-persistence-type</code>	<code>file</code>	(optional) Specifies the passivation mechanism for SFSBs that <i>do not</i> have availability enabled. Allowed values are <code>file</code> and <code>ha</code> .
<code>sfsb-store-pool-name</code>	<code>availability-service</code> element's <code>store-pool-name</code> attribute value	(optional) Specifies the <code>jndi-name</code> of the <code>jdbc-resource</code> used for connections to the HADB for session persistence. For more information about setting up a connection pool and JDBC resource for the HADB, see the description of the <code>configure-ha-cluster</code> command in the <i>Sun Java System Application Server Reference Manual</i> .

ejb-module

Specifies a deployed EJB module.

Superelements

[applications](#)

Subelements

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

Table 1-48 `ejb-module` Subelements

Element	Required	Description
description	zero or one	Contains a text description of this element.

Attributes

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

Table 1-49 `ejb-module` Attributes

Attribute	Default	Description
<code>name</code>	<code>none</code>	The name of the EJB module.
<code>location</code>	<code>none</code>	The location of the EJB module in the Sun Java System Application Server file system.
<code>object-type</code>	<code>user</code>	(optional) Defines the type of the resource. Allowed values are: <ul style="list-style-type: none"> <code>system-all</code> - A system resource for all server instances and the domain application server. <code>system-admin</code> - A system resource only for the the domain application server. <code>system-instance</code> - A system resource for all server instances only. <code>user</code> - A user resource.
<code>enabled</code>	<code>true</code>	(optional) Determines whether the EJB module is enabled.
<code>availability-enabled</code>	<code>false</code>	(optional) Specifies whether availability is enabled in this EJB module for SFSB checkpointing (and potentially passivation). Availability must also be enabled for the application or stand-alone EJB module during deployment. For more information about availability, see availability-service .
<code>directory-deployed</code>	<code>false</code>	(optional) Specifies whether the application has been deployed to a directory.

ejb-timer-service

Configures the EJB timer service.

Superelements

[ejb-container](#)

Subelements

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

Table 1-50 `ejb-timer-service` Subelements

Element	Required	Description
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `ejb-timer-service` element.

Table 1-51 `ejb-timer-service` Attributes

Attribute	Default	Description
<code>minimum-delivery-interval-in-millis</code>	7000	(optional) Specifies the minimum time before an expiration for a particular timer can occur. This guards against extremely small timer increments that can overload the server.
<code>max-redeliveries</code>	1	(optional) Specifies the maximum number of times the EJB timer service attempts to redeliver a timer expiration due for exception or rollback.
<code>timer-datasource</code>	<code>jdbc/</code> <code>__TimerPool</code>	(optional) Overrides, for the cluster or server instance, the <code>cmp-resource</code> value specified in <code>sun-ejb-jar.xml</code> for the timer service system application (<code>__ejb_container_timer_app</code>).
<code>redelivery-interval-interval-in-millis</code>	5000	(optional) Specifies how long the EJB timer service waits after a failed <code>ejbTimeout</code> delivery before attempting a redelivery.

external-jndi-resource

Defines a resource that resides in an external JNDI repository. For example, a generic Java object could be stored in an LDAP server. An external JNDI factory must implement the `javax.naming.spi.InitialContextFactory` interface.

Superelements

[resources](#)

Subelements

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

Table 1-52 `external-jndi-resource` Subelements

Element	Required	Description
description	zero or one	Contains a text description of this element.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `external-jndi-resource` element.

Table 1-53 `external-jndi-resource` Attributes

Attribute	Default	Description
<code>jndi-name</code>	none	Specifies the JNDI name for the resource.
<code>jndi-lookup-name</code>	none	Specifies the JNDI lookup name for the resource.
<code>res-type</code>	none	Specifies the fully qualified type of the resource.
<code>factory-class</code>	none	Specifies the fully qualified name of the factory class, which implements <code>javax.naming.spi.InitialContextFactory</code> . For more information about JNDI, see the <i>Sun Java System Application Server Developer's Guide</i> .

Table 1-53 external-jndi-resource Attributes (*Continued*)

Attribute	Default	Description
object-type	user	(optional) Defines the type of the resource. Allowed values are: <ul style="list-style-type: none"> system-all - A system resource for all server instances and the domain application server. system-admin - A system resource only for the the domain application server. system-instance - A system resource for all server instances only. user - A user resource.
enabled	true	(optional) Determines whether this resource is enabled at runtime.

F

filter-config

Configures the filter class that filters alerts from notification emitters. See also [listener-config](#).

Superelements

[alert-subscription](#)

Subelements

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

Table 1-54 filter-config Subelements

Element	Required	Description
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `filter-config` element.

Table 1-55 filter-config Attributes

Attribute	Default	Description
filter-class-name	none	Specifies the class name of the filter.

H

health-checker

Defines a health checker for the parent [server-ref](#) or [cluster-ref](#) element.

Superelements

[cluster-ref](#), [server-ref](#)

Subelements

none

Attributes

The following table describes attributes for the health-checker element.

Table 1-56 health-checker Attributes

Attribute	Default	Description
url	/	Specifies the URL to ping to determine the health state of a listener. This must be a relative URL.
interval-in-seconds	30	Specifies the interval between health checks. A value of zero means that health checking is disabled.
timeout-in-seconds	10	Specifies the maximum time in which a server must respond to a health check request to be considered healthy. If <code>interval-in-seconds</code> is greater than zero, <code>timeout-in-seconds</code> must be less than or equal to <code>interval-in-seconds</code> .

http-access-log

Defines an access log file for a [virtual-server](#). The [access-log](#) subelement of the virtual server's parent [http-service](#) element determines the access log file's format and rotation settings.

Superelements[virtual-server](#)**Subelements**

none

Attributes

The following table describes attributes for the `http-access-log` element.

Table 1-57 `http-access-log` Attributes

Attribute	Default	Description
<code>log-directory</code>	<code>\${com.sun.aas.instanceRoot}/logs/access</code>	(optional) Specifies the location of the access log file.
<code>iponly</code>	<code>true</code>	(optional) If <code>true</code> , specifies that only the IP address of the user agent is listed. If <code>false</code> , performs a DNL lookup.

http-file-cache

Configures the HTTP file cache.

Superelements[http-service](#)**Subelements**

none

Attributes

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

Table 1-58 `http-file-cache` Attributes

Attribute	Default	Description
<code>globally-enabled</code>	<code>true</code>	(optional) If <code>true</code> , enables the file cache.
<code>file-caching-enabled</code>	<code>on</code>	(optional) If <code>on</code> , enables caching of the file content if the file size exceeds the <code>medium-file-size-limit-in-bytes</code> .
<code>max-age-in-seconds</code>	<code>30</code>	(optional) Specifies the maximum age of a file cache entry.
<code>medium-file-size-limit-in-bytes</code>	<code>537600</code>	(optional) Specifies the maximum size of a file that can be cached as a memory mapped file.

Table 1-58 http-file-cache Attributes (Continued)

Attribute	Default	Description
medium-file-space-in-bytes	10485760	(optional) Specifies the total size of all files that are cached as memory mapped files.
small-file-size-limit-in-bytes	2048	(optional) Specifies the maximum size of a file that can be read into memory.
small-file-space-in-bytes	1048576	(optional) Specifies the total size of all files that are read into memory.
file-transmission-enabled	false	(optional) If true, enables the use of TransmitFileSystem calls. Meaningful only for Windows.
max-files-count	1024	(optional) Specifies the maximum number of files in the file cache.
hash-init-size	0	(optional) Specifies the initial number of hash buckets.

http-listener

Defines an HTTP listen socket. The [connection-pool](#) subelement of the parent [http-service](#) element also configures some listen socket settings.

Superelements

[http-service](#)

Subelements

The following table describes subelements for the `http-listener` element.

Table 1-59 http-listener Subelements

Element	Required	Description
ssl	zero or one	Defines SSL parameters.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `http-listener` element.

Table 1-60 http-listener Attributes

Attribute	Default	Description
id	none	The unique listener name. An <code>http-listener</code> name cannot begin with a number.

Table 1-60 http-listener Attributes (*Continued*)

Attribute	Default	Description
address	none	IP address of the listener. Can be in dotted-pair or IPv6 notation. Can be any (for <code>INADDR_ANY</code>) to listen on all IP addresses. Can be a hostname.
port	none	Port number on which the listener listens. Legal values are 1 - 65535. On UNIX, creating sockets that listen on ports 1 - 1024 requires superuser privileges. Configuring an SSL listener to listen on port 443 is standard.
external-port	none	(optional) Specifies the external port on which the connection is made.
family	inet	(optional) Specifies whether the IP address is an <code>inet</code> or <code>nca</code> address.
blocking-enabled	false	(optional) If <code>true</code> , enables blocking on both the external and listener ports.
acceptor-threads	1	(optional) Number of acceptor threads for the listener, typically the number of processors in the machine. Legal values are 1 - 1024.
security-enabled	false	(optional) Determines whether the listener runs SSL. To turn SSL2 or SSL3 on or off and set ciphers, use an <code>ssl</code> subelement.
default-virtual-server	none	References the <code>id</code> attribute of the default <code>virtual-server</code> for this particular listener.
server-name	none	Tells the server what to put in the host name section of any URLs it sends to the client. This affects URLs the server automatically generates; it doesn't affect the URLs for directories and files stored in the server. If your server uses an alias, the <code>server-name</code> should be the alias name. If a colon and port number are appended, that port is used in URLs the server sends to the client.
redirect-port	none	(optional) If the listener is supporting non-SSL requests and a request is received for which a matching <code><security-constraint></code> requires SSL transport, the request is automatically redirected to the port number specified here.
xpowered-by	true	(optional) If <code>true</code> , X-Powered-By headers are used according to the Servlet 2.4 and JSP 2.0 specifications.
enabled	true	(optional) Determines whether the listener is active.

http-protocol

Configures HTTP protocol settings.

Superelements

[http-service](#)

Subelements

none

Attributes

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

Table 1-61 `protocol` Attributes

Attribute	Default	Description
<code>version</code>	HTTP/1.1	(optional) Specifies the version of the HTTP protocol used.
<code>dns-lookup-enabled</code>	true	(optional) If true, looks up the DNS entry for the client.
<code>forced-response-type</code>	ISO-8859-1;en;ISO-8859-1	(optional) Specifies the response type used if no MIME mapping is available that matches the file extension. The format is a semicolon-delimited string consisting of the content-type, encoding, language, and charset.
<code>default-response-type</code>	text/html;ISO-8859-1;en;ISO-8859-1	(optional) Specifies the default response type. The format is a semicolon-delimited string consisting of the content-type, encoding, language, and charset.
<code>ssl-enabled</code>	true	(optional) If true, globally enables SSL for all <code>http-listener</code> subelements of the parent <code>http-service</code> element.

http-service

Defines the HTTP service.

Superelements

[config](#)

Subelements

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

Table 1-62 `http-service` Subelements

Element	Required	Description
access-log	zero or one	Defines access log settings for each http-access-log subelement of each virtual-server .
http-listener	one or more	Defines an HTTP listen socket.
virtual-server	one or more	Defines a virtual server.
request-processing	zero or one	Configures request processing threads.
keep-alive	zero or one	Configures keep-alive threads.
connection-pool	zero or one	Defines a pool of client HTTP connections.

Table 1-62 http-service Subelements (*Continued*)

Element	Required	Description
http-protocol	zero or one	Configures HTTP protocol settings.
http-file-cache	zero or one	Configures the HTTP file cache.
property	zero or more	Specifies a property or a variable.

Properties

The following table describes properties for the `http-service` element, which configure SSL for all [http-listener](#) subelements.

Table 1-63 http-service Properties

Property	Default	Description
<code>monitoring-cache-enabled</code>	<code>true</code>	If <code>true</code> , enables the monitoring cache.
<code>monitoring-cache-refresh-in-millis</code>	<code>5000</code>	Specifies the interval between refreshes of the monitoring cache.
<code>ssl-cache-entries</code>	<code>10000</code>	Specifies the number of SSL sessions to be cached.
<code>ssl3-session-timeout</code>	<code>86400</code>	Specifies the interval at which SSL3 sessions are cached.
<code>ssl-client-auth-data-limit</code>	<code>1048576</code>	Specifies the maximum amount of data cached during the handshake phase.
<code>ssl-client-auth-timeout</code>	<code>60</code>	Specifies the timeout for the client certificate phase.
<code>ssl-session-timeout</code>	<code>100</code>	Specifies the interval at which SSL2 sessions are cached.
<code>keep-alive-query-mean-time</code>	<code>100</code>	Specifies the keep-alive latency.
<code>keep-alive-query-max-sleep-time</code>	<code>100</code>	Specifies the upper limit to the time slept after polling keep-alive connections for further requests.
<code>stack-size</code>	depends on operating system	Specifies the maximum stack size of the native thread.

iiop-listener

Defines an IIOP listen socket.

To enable SSL for this listener, include an `ssl` subelement.

Superelements

[iiop-service](#)

Subelements

The following table describes subelements for the `iiop-listener` element.

Table 1-64 `iiop-listener` Subelements

Element	Required	Description
ssl	zero or one	Defines SSL parameters.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `iiop-listener` element.

Table 1-65 `iiop-listener` Attributes

Attribute	Default	Description
<code>id</code>	<code>none</code>	The listener name. An <code>iiop-listener</code> name cannot begin with a number.
<code>address</code>	<code>none</code>	IP address of the listener. Can be in dotted-pair or IPv6 notation, or just a name.
<code>port</code>	<code>1072</code>	(optional) Port number for the listener. Legal values are 1 - 65535. On UNIX, creating sockets that listen on ports 1 - 1024 requires superuser privileges.
<code>security-enabled</code>	<code>false</code>	(optional) Determines whether the listener runs SSL. To turn SSL2 or SSL3 on or off and set ciphers, use an <code>ssl</code> element.
<code>enabled</code>	<code>true</code>	(optional) Determines whether the listener is active.

iiop-service

Defines the IIOP service.

Superelements

[config](#)

Subelements

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

Table 1-66 `iiop-service` Subelements

Element	Required	Description
orb	only one	Configures the ORB.
ssl-client-config	zero or one	Defines SSL parameters for the ORB.
iiop-listener	zero or more	Defines an IIOP listen socket.

Attributes

The following table describes attributes for the `iiop-service` element.

Table 1-67 `iiop-service` Attributes

Attribute	Default	Description
<code>client-authentication-required</code>	<code>false</code>	(optional) If <code>true</code> , the server rejects unauthenticated requests and inserts an authentication-required bit in IORs sent to clients.

J

j2ee-application

Specifies a deployed J2EE application.

Superelements

[applications](#)

Subelements

The following table describes subelements for the `j2ee-application` element.

Table 1-68 j2ee-application Subelements

Element	Required	Description
description	zero or one	Contains a text description of this element.

Attributes

The following table describes attributes for the `j2ee-application` element.

Table 1-69 j2ee-application Attributes

Attribute	Default	Description
<code>name</code>	none	The name of the application.
<code>location</code>	none	The location of the application in the Sun Java System Application Server file system.
<code>object-type</code>	user	(optional) Defines the type of the resource. Allowed values are: <ul style="list-style-type: none"> <code>system-all</code> - A system resource for all server instances and the domain application server. <code>system-admin</code> - A system resource only for the the domain application server. <code>system-instance</code> - A system resource for all server instances only. <code>user</code> - A user resource.
<code>enabled</code>	true	(optional) Determines whether the application is enabled.
<code>availability-enabled</code>	false	(optional) Specifies whether availability is enabled in this J2EE application for HTTP session persistence and SFSB checkpointing (and potentially passivation). Availability must also be enabled for the application during deployment. For more information about availability, see availability-service .
<code>directory-deployed</code>	false	(optional) Specifies whether the application has been deployed to a directory.

jacc-provider

Specifies a Java Authorization Contract for Containers (JACC) provider for pluggable authorization.

Superelements

[security-service](#)

Subelements

The following table describes subelements for the `jacc-provider` element.

Table 1-70 `jacc-provider` Subelements

Element	Required	Description
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `jacc-provider` element.

Table 1-71 `jacc-provider` Attributes

Attribute	Default	Description
<code>name</code>	<code>default</code>	Specifies the name of the JACC provider.
<code>policy-provider</code>	<code>none</code>	Corresponds to and can be overridden by the system property <code>javax.security.jacc.policy.provider</code> .
<code>policy-configuration-factory-provider</code>	<code>none</code>	Corresponds to and can be overridden by the system property <code>javax.security.jacc.PolicyConfigurationFactory.provider</code> .

java-config

Specifies Java Virtual Machine (JVM) configuration parameters.

Superelements

[config](#)

Subelements

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

Table 1-72 `java-config` Subelements

Element	Required	Description
profiler	zero or one	Configures a profiler for use with Sun Java System Application Server.
jvm-options	zero or more	Contains JVM command line options.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `java-config` element.

Table 1-73 java-config Attributes

Attribute	Default	Description
java-home	none	The path to the directory where the JDK is installed.
debug-enabled	false	(optional) If <code>true</code> , the server starts up in debug mode ready for attachment with a JPDA-based debugger.
debug-options	-Xdebug -Xrunjdpw:transport =dt_socket,server=y ,suspend=n	(optional) Specifies JPDA (Java Platform Debugger Architecture) options. A list of debugging options is available here: http://java.sun.com/products/jpda/doc/conninv.html#Invocation For more information about debugging, see the <i>Sun Java System Application Server Developer's Guide</i> .
rmic-options	-iiop -poa -alwaysgenerate -keepgenerated -g	(optional) Specifies options passed to the RMI compiler at application deployment time. The <code>-keepgenerated</code> option saves generated source for stubs and ties. For details about the <code>rmic</code> command, see: http://java.sun.com/j2se/1.4.2/docs/tooldocs/solaris/rmic.html
javac-options	-g	(optional) Specifies options passed to the Java compiler at application deployment time.
classpath-prefix	none	(optional) Specifies a prefix for the system classpath. Only prefix the system classpath to override system classes, such as the XML parser classes. Use this attribute with caution.
classpath-suffix	none	(optional) Specifies a suffix for the system classpath.
server-classpath	none	(optional) Specifies the classpath for the environment from which the server was started. This classpath can be accessed using <code>System.getProperty("java.class.path")</code> .
native-library-path-prefix	none	(optional) Specifies a prefix for the native library path. The native library path is the automatically constructed concatenation of the Application Server installation relative path for its native shared libraries, the standard JRE native library path, the shell environment setting (<code>LD_LIBRARY_PATH</code> on UNIX), and any path specified in the <code>profiler</code> element. Since this is synthesized, it does not appear explicitly in the server configuration.
native-library-path-suffix	none	(optional) Specifies a suffix for the native library path.

Table 1-73 java-config Attributes (*Continued*)

Attribute	Default	Description
bytecode-preprocessors	none	(optional) A comma separated list of class names, each of which must implement the <code>com.sun.appserv.BytecodePreprocessor</code> interface. Each of the specified preprocessor classes is called in the order specified.
env-classpath-ignored	true	(optional) If <code>false</code> , the <code>CLASSPATH</code> environment variable is read and appended to the Sun Java System Application Server classpath. The <code>CLASSPATH</code> environment variable is added after the <code>classpath-suffix</code> , at the very end. For a development environment, this value should be set to <code>false</code> . To prevent environment variable side effects in a production environment, set this value to <code>true</code> .

jdbc-connection-pool

Defines the properties that are required for creating a JDBC connection pool.

Superelements

[resources](#)

Subelements

The following table describes subelements for the `jdbc-connection-pool` element.

Table 1-74 jdbc-connection-pool Subelements

Element	Required	Description
description	zero or one	Contains a text description of this element.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `jdbc-connection-pool` element.

Table 1-75 jdbc-connection-pool Attributes

Attribute	Default	Description
name	none	Specifies the name of the connection pool. A <code>jdbc-resource</code> element's <code>pool-name</code> attribute refers to this name.
datasource-classname	none	Specifies the class name of the associated vendor-supplied data source. This class must implement <code>java.sql.DataSource</code> , <code>java.sql.XADataSource</code> , <code>javax.sql.ConnectionPoolDataSource</code> , or a combination.
res-type	<code>javax.sql.DataSource</code>	(optional) Specifies the interface the data source class implements. The value of this attribute can be <code>javax.sql.DataSource</code> , <code>javax.sql.XADataSource</code> , or <code>javax.sql.ConnectionPoolDataSource</code> . If the value is not one of these interfaces, the default is used. An error occurs if this attribute has a legal value and the indicated interface is not implemented by the data source class.
steady-pool-size	8	(optional) Specifies the initial and minimum number of connections maintained in the pool.
max-pool-size	32	(optional) Specifies the maximum number of connections that can be created to satisfy client requests.
max-wait-time-in-millis	60000	(optional) Specifies the amount of time, in milliseconds, that the caller is willing to wait for a connection. If 0, the caller is blocked indefinitely until a resource is available or an error occurs.
pool-resize-quantity	2	(optional) Specifies the number of connections to be destroyed if the existing number of connections is above the <code>steady-pool-size</code> (subject to the <code>max-pool-size</code> limit). This is enforced periodically at the <code>idle-time-out-in-seconds</code> interval. An idle connection is one that has not been used for a period of <code>idle-time-out-in-seconds</code> .
idle-timeout-in-seconds	300	(optional) Specifies the maximum time that a connection can remain idle in the pool. After this amount of time, the pool can close this connection.

Table 1-75 jdbc-connection-pool Attributes (*Continued*)

Attribute	Default	Description
transaction-isolation-level	default JDBC driver isolation level	(optional) Specifies the transaction isolation level on the pooled database connections. Allowed values are <code>read-uncommitted</code> , <code>read-committed</code> , <code>repeatable-read</code> , or <code>serializable</code> . Applications that change the isolation level on a pooled connection programmatically risk polluting the pool, which can lead to errors. See <code>is-isolation-level-guaranteed</code> for more details.
is-isolation-level-guaranteed	true	(optional) Applicable only when <code>transaction-isolation-level</code> is explicitly set. If true, every connection obtained from the pool is guaranteed to have the desired isolation level. This might impact performance on some JDBC drivers. Only set this attribute to false if you are certain that the hosted applications do not return connections with altered isolation levels.
is-connection-validation-required	false	(optional) Specifies whether connections have to be validated before being given to the application. If a resource's validation fails, it is destroyed, and a new resource is created and returned.
connection-validation-method	auto-commit	(optional) Legal values are as follows: <ul style="list-style-type: none"> • <code>auto-commit</code> (default), which uses <code>Connection.setAutoCommit(Connection.getAutoCommit())</code> • <code>meta-data</code>, which uses <code>Connection.getMetaData()</code> • <code>table</code>, which performs a query on a table specified in the <code>validation-table-name</code> attribute
validation-table-name	none	(optional) Specifies the table name to be used to perform a query to validate a connection. This parameter is mandatory if and only if <code>connection-validation-type</code> is set to <code>table</code> .
fail-all-connections	false	(optional) If true, closes all connections in the pool if a single validation check fails. This parameter is mandatory if and only if <code>is-connection-validation-required</code> is set to true.

Properties

Most JDBC 3.0 drivers allow use of standard property lists to specify the user, password, and other resource configuration information. Although properties are optional with respect to Sun Java System Application Server, some properties might be necessary for most databases. For details, see the JDBC 3.0 Standard Extension API.

When properties are specified, they are passed to the vendor's data source class (specified by the `datasource-classname` attribute) as is using `setName(value)` methods.

The `user` and `password` properties are used as the default principal if container managed authentication is specified and a `default-resource-principal` is not found in the application deployment descriptors.

The following table describes some common properties for the `jdbc-connection-pool` element.

Table 1-76 `jdbc-connection-pool` Properties

Property	Description
<code>user</code>	Specifies the user name for this connection pool.
<code>password</code>	Specifies the password for this connection pool.
<code>databaseName</code>	Specifies the database for this connection pool.
<code>serverName</code>	Specifies the database server for this connection pool.
<code>port</code>	Specifies the port on which the database server listens for requests.
<code>networkProtocol</code>	Specifies the communication protocol.
<code>roleName</code>	Specifies the initial SQL role name.
<code>datasourceName</code>	Specifies an underlying <code>XADataSource</code> , or a <code>ConnectionPoolDataSource</code> if connection pooling is done.
<code>description</code>	Specifies a text description.
<code>url</code>	Specifies the URL for this connection pool. Although this is not a standard property, it is commonly used.

jdbc-resource

Defines a JDBC (`javax.sql.DataSource`) resource.

Superelements

[resources](#)

Subelements

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

Table 1-77 `jdbc-resource` Subelements

Element	Required	Description
<code>description</code>	zero or one	Contains a text description of this element.
<code>property</code>	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `jdbc-resource` element.

Table 1-78 `jdbc-resource` Attributes

Attribute	Default	Description
<code>jndi-name</code>	none	Specifies the JNDI name for the resource.
<code>pool-name</code>	none	Specifies the name of the associated <code>jdbc-connection-pool</code> .
<code>object-type</code>	user	(optional) Defines the type of the resource. Allowed values are: <ul style="list-style-type: none"> <code>system-all</code> - A system resource for all server instances and the domain application server. <code>system-admin</code> - A system resource only for the the domain application server. <code>system-instance</code> - A system resource for all server instances only. <code>user</code> - A user resource.
<code>enabled</code>	true	(optional) Determines whether this resource is enabled at runtime.

jms-host

Configures the host of the built-in Java Message Service (JMS) that is managed by the Sun Java System Application Server.

Superelements

`jms-service`

Subelements

The following table describes subelements for the `jms-host` element.

Table 1-79 `.jms-host` Subelements

Element	Required	Description
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `.jms-host` element.

Table 1-80 `.jms-host` Attributes

Attribute	Default	Description
<code>name</code>	<code>none</code>	Specifies the name of the JMS host.
<code>host</code>	<code>machine_name</code>	(optional) Specifies the host name of the JMS host.
<code>port</code>	<code>7676</code>	(optional) Specifies the port number used by the JMS provider.
<code>admin-user-name</code>	<code>admin</code>	(optional) Specifies the administrator user name for the JMS provider.
<code>admin-password</code>	<code>admin</code>	(optional) Specifies the administrator password for the JMS provider.

`.jms-service`

Configures the built-in Java Message Service (JMS) that is managed by the Sun Java System Application Server.

Superelements

[config](#)

Subelements

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

Table 1-81 `.jms-service` Subelements

Element	Required	Description
.jms-host	zero or more	Specifies a host.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `.jms-service` element.

Table 1-82 `jms-service` Attributes

Attribute	Default	Description
<code>init-timeout-in-seconds</code>	60	(optional) Specifies the amount of time the server instance waits at startup for its configured default JMS host to respond. If there is no response, startup is aborted. If set to 0, the server instance waits indefinitely.
<code>type</code>	LOCAL	(optional) Specifies the type of JMS service: <ul style="list-style-type: none"> • LOCAL means the JMS provider is started along with the application server • REMOTE means the JMS provider is remote and is not started by the application server
<code>start-args</code>	none	(optional) Specifies the string of arguments supplied for startup of the corresponding JMS instance.
<code>default-jms-host</code>	none	Specifies the name of the default <code>jms-host</code> . If <code>type</code> is set to LOCAL, this <code>jms-host</code> is automatically started at application server start-up.
<code>reconnect-interval-in-seconds</code>	60	(optional) Specifies the interval between reconnect attempts.
<code>reconnect-attempts</code>	3	(optional) Specifies the number of reconnect attempts.
<code>reconnect-enabled</code>	true	(optional) If true, reconnection is enabled. The JMS service automatically tries to reconnect to the JMS provider when the connection is broken. When the connection is broken, depending on the message processing stage, the <code>onMessage()</code> method might not be able to complete successfully or the transaction might be rolled back due to a JMS exception. When the JMS service reestablishes the connection, JMS message redelivery semantics apply.
<code>addresslist-behavior</code>	random	(optional) Specifies whether the reconnection logic selects the broker from the <code>imqAddressList</code> in a random or sequential (priority) fashion.
<code>addresslist-iterations</code>	3	(optional) Specifies the number of times the reconnection logic iterates over the <code>imqAddressList</code> if <code>addresslist-behavior</code> is set to PRIORITY.
<code>mq-scheme</code>	mq	(optional) Specifies the scheme for establishing connection with the broker. For example, specify <code>http</code> for connecting to the broker over HTTP.
<code>mq-service</code>	jms	(optional) Specifies the type of broker service. If a broker supports SSL, the type of service can be <code>ssljms</code> .

Properties

The following table describes properties for the `jms-service` element.

Table 1-83 `jms-service` Properties

Property	Default	Description
<code>instance-name</code>	<code>imqbroker</code>	Specifies the full Sun Java System Message Queue broker instance name.
<code>instance-name-suffix</code>	<code>none</code>	Specifies a suffix to add to the full Sun Java System Message Queue broker instance name. The suffix is separated from the instance name by an underscore character (<code>_</code>). For example, if the instance name is <code>imqbroker</code> , appending the suffix <code>xyz</code> changes the instance name to <code>imqbroker_xyz</code> .
<code>append-version</code>	<code>false</code>	If <code>true</code> , appends the major and minor version numbers, preceded by underscore characters (<code>_</code>), to the full Sun Java System Message Queue broker instance name. For example, if the instance name is <code>imqbroker</code> , appending the version numbers changes the instance name to <code>imqbroker_8_0</code> .

jmx-connector

Configures a JSR 160 compliant remote JMX connector, which handles communication between the domain administration server and the node agents for remote server instances. See also [admin-service](#) and [node-agent](#).

Superelements

[admin-service](#), [node-agent](#)

Subelements

The following table describes subelements for the `jmx-connector` element.

Table 1-84 `jmx-connector` Subelements

Element	Required	Description
<code>ssl</code>	zero or one	Defines SSL parameters.
<code>property</code>	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `jmx-connector` element.

Table 1-85 jmx-connector Attributes

Attribute	Default	Description
name	none	Specifies the JMX connector name.
protocol	rmi_jrmp	(optional) Specifies the protocol that this JMX connector supports.
address	none	Specifies the IP address of the JMX connector. Can be in dotted-pair or IPv6 notation. Can be any (for INADDR_ANY) to listen on all IP addresses. Can be a hostname.
port	none	Specifies the port number on which the JMX connector listens. Legal values are 1 - 65535. On UNIX, creating sockets that listen on ports 1 - 1024 requires superuser privileges. Configuring an SSL-enabled JMX connector to listen on port 443 is standard.
auth-realm-name	none	Specifies the name of the <code>auth-realm</code> subelement of the parent <code>node-agent</code> element that represents the special administrative realm. This realm handles all authentication for the Administration Console and the <code>asadmin</code> command.
security-enabled	true	(optional) Determines whether the JMX connector runs SSL. To turn SSL2 or SSL3 on or off and set ciphers, use an <code>ssl</code> subelement.

jvm-options

Contains JVM command line options, for example:

```
<jvm-options>-Xdebug -Xmx128m</jvm-options>
```

For information about JVM options, see:

<http://java.sun.com/docs/hotspot/VMOptions.html>

Superelements

[java-config](#), [profiler](#)

Subelements

none - contains data

K

keep-alive

Configures keep-alive threads.

Superelements

[http-service](#)

Subelements

none

Attributes

The following table describes attributes for the `keep-alive` element.

Table 1-86 keep-alive Attributes

Attribute	Default	Description
<code>thread-count</code>	1	(optional) Specifies the number of keep-alive threads.
<code>max-connections</code>	256	(optional) Specifies the maximum number of keep-alive connections.
<code>timeout-in-seconds</code>	30	(optional) Specifies the maximum time for which a keep alive connection is kept open.

L

lb-config

Defines a load balancer configuration.

Superelements

[lb-configs](#)

Subelements

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

Table 1-87 lb-config Subelements

Element	Required	Description
cluster-ref	zero or more; zero if a <code>server-ref</code> is defined	References a cluster.

Table 1-87 lb-config Subelements (*Continued*)

Element	Required	Description
server-ref	zero or more; zero if a cluster-ref is defined	References a server instance that does not belong to a cluster.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the lb-config element.

Table 1-88 lb-config Attributes

Attribute	Default	Description
name	none	Specifies the name of the load balancer configuration.
response-timeout-in-seconds	60	(optional) Specifies the time within which a server must return a response or it is considered unhealthy.
https-routing	false	(optional) If true, HTTPS requests to the load balancer result in HTTPS requests to the server. If false, HTTPS requests to the load balancer result in HTTP requests to the server.
reload-poll-interval-in-seconds	60	(optional) Specifies the interval between checks for changes to the load balancer configuration file (loadbalancer.xml). When changes are detected, the file is reloaded. A value of zero disables reloading.
monitoring-enabled	false	(optional) If true, enables monitoring of load balancing.

lb-configs

Contains load balancer configurations.

Superelements

[domain](#)

Subelements

The following table describes subelements for the lb-configs element.

Table 1-89 lb-configs Subelements

Element	Required	Description
lb-config	zero or more	Defines a load balancer configuration.

lifecycle-module

Specifies a deployed lifecycle module. For more information about lifecycle modules, see the *Sun Java System Application Server Developer's Guide*.

Superelements

[applications](#)

Subelements

The following table describes subelements for the `lifecycle-module` element.

Table 1-90 `lifecycle-module` Subelements

Element	Required	Description
description	zero or one	Contains a text description of this element.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `lifecycle-module` element.

Table 1-91 `lifecycle-module` Attributes

Attribute	Default	Description
<code>name</code>	none	The name of the lifecycle module.
<code>class-name</code>	none	The fully qualified name of the lifecycle module's class file, which must implement the <code>com.sun.appserv.server.LifecycleListener</code> interface.
<code>classpath</code>	value of <code>application-root</code> attribute of server element	(optional) The classpath for the lifecycle module. Specifies where the module is located.
<code>load-order</code>	none	(optional) Determines the order in which lifecycle modules are loaded at startup. Modules with smaller integer values are loaded sooner. Values can range from 101 to the operating system's <code>MAXINT</code> . Values from 1 to 100 are reserved.
<code>is-failure-fatal</code>	false	(optional) Determines whether the server is shut down if the lifecycle module fails.
<code>enabled</code>	true	(optional) Determines whether the lifecycle module is enabled.

listener-config

Configures the listener class that listens for alerts from notification emitters. For example:

```
<listener-config
  listener-class-name="com.sun.enterprise.admin.notification.MailAlert"
  subscribe-listener-with="LogMBean,ServerStatusMonitor" >
  <property name="recipients" value="Huey@sun.com,Dewey@sun.com,Louie@sun.com" />
  <property name="fromAddress" value="DonaldDuck@sun.com" />
  <property name="subject" value="Help!" />
  <property name="includeDiagnostics" value="false" />
  <property name="mailSMTPHost" value="ducks.sun.com" />
</listener-config>
```

Superelements

[alert-subscription](#)

Subelements

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

Table 1-92 `listener-config` Subelements

Element	Required	Description
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `listener-config` element.

Table 1-93 `listener-config` Attributes

Attribute	Default	Description
<code>listener-class-name</code>	none	Specifies the class name of the listener. The <code>com.sun.appserv.admin.notification.MailAlert</code> class is provided with Sun Java System Application Server, but a custom listener can be used.
<code>subscribe-listener-with</code>	none	Specifies a comma-separated list of notification emitters to which the listener listens. The <code>LogMBean</code> and <code>ServerStatusMonitor</code> notification emitters are provided with Sun Java System Application Server, but custom emitters can be used.

log-service

Configures the **server log** file, which stores messages from the default virtual server. Messages from other configured virtual servers also go here, unless the `log-file` attribute is explicitly specified in the `virtual-server` element. The default name is `server.log`.

Other log files are configured by other elements:

- A **virtual server log** file stores messages from a `virtual-server` element that has an explicitly specified `log-file` attribute. See [virtual-server](#).
- The **access log** file stores HTTP access messages from the default virtual server. The default name is `access.log`. See [access-log](#) and [http-access-log](#).
- The **transaction log** files store transaction messages from the default virtual server. The default name of the directory for these files is `tx`. See [transaction-service](#).

Superelements

[config](#), [node-agent](#)

Subelements

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

Table 1-94 `log-service` Subelements

Element	Required	Description
module-log-levels	zero or one	Specifies log levels.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `log-service` element.

Table 1-95 `log-service` Attributes

Attribute	Default	Description
<code>file</code>	<code>server.log</code> in the directory specified by the <code>log-root</code> attribute of the <code>domain</code> element	(optional) Overrides the name or location of the server log. The file and directory in which the server log is kept must be writable by the user account under which the server runs. An absolute path overrides the <code>log-root</code> attribute of the <code>domain</code> element. A relative path is relative to the <code>log-root</code> attribute of the <code>domain</code> element. If no <code>log-root</code> value is specified, it is relative to <code>domain_dir/config</code> .

Table 1-95 log-service Attributes (*Continued*)

Attribute	Default	Description
use-system-logging	false	(optional) If true, uses the UNIX syslog service to produce and manage logs.
log-handler	none	(optional) Specifies a custom log handler to be added to end of the chain of system handlers to log to a different destination.
log-filter	none	(optional) Specifies a log filter to do custom filtering of log records.
log-to-console	false	(optional) Deprecated and ignored.
log-rotation-limit-in-bytes	2000000	(optional) Log files are rotated when the file size reaches the specified limit.
log-rotation-timelimit-in-minutes	0	(optional) Enables time-based log rotation. The valid range is 60 minutes (1 hour) to 14400 minutes (10*24*60 minutes or 10 days). If the value is zero, the files are rotated based on the size specified in log-rotation-limit-in-bytes. If the value is greater than zero, log-rotation-timelimit-in-minutes takes precedence over log-rotation-limit-in-bytes.
alarms	false	(optional) Deprecated and ignored.

M

mail-resource

Defines a JavaMail (`javax.mail.Session`) resource.

Superelements

[resources](#)

Subelements

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

Table 1-96 mail-resource Subelements

Element	Required	Description
description	zero or one	Contains a text description of this element.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the mail-resource element.

Table 1-97 mail-resource Attributes

Attribute	Default	Description
jndi-name	none	Specifies the JNDI name for the resource.
store-protocol	imap	(optional) Specifies the storage protocol service, which connects to a mail server, retrieves messages, and saves messages in folder(s). Allowed values are <code>imap</code> , <code>pop3</code> , <code>imaps</code> , and <code>pop3s</code> .
store-protocol-class	<code>com.sun.mail.imap.IMAPStore</code>	(optional) Specifies the service provider implementation class for storage. Allowed values are: <code>com.sun.mail.imap.IMAPStore</code> <code>com.sun.mail.pop3.POP3Store</code> <code>com.sun.mail.imap.IMAPSSLStore</code> <code>com.sun.mail.pop3.POP3SSLStore</code>
transport-protocol	smtp	(optional) Specifies the transport protocol service, which sends messages. Allowed values are <code>smtp</code> and <code>smtps</code> .
transport-protocol-class	<code>com.sun.mail.smtp.SMTPTransport</code>	(optional) Specifies the service provider implementation class for transport. Allowed values are: <code>com.sun.mail.smtp.SMTPTransport</code> <code>com.sun.mail.smtp.SMTPSSLTransport</code>
host	none	The mail server host name.
user	none	The mail server user name.
from	none	The e-mail address the mail server uses to indicate the message sender.
debug	false	(optional) Determines whether debugging for this resource is enabled.

Table 1-97 mail-resource Attributes (*Continued*)

Attribute	Default	Description
object-type	user	(optional) Defines the type of the resource. Allowed values are: <ul style="list-style-type: none"> • system-all - A system resource for all server instances and the domain application server. • system-admin - A system resource only for the the domain application server. • system-instance - A system resource for all server instances only. • user - A user resource.
enabled	true	(optional) Determines whether this resource is enabled at runtime.

Properties

You can set properties for the mail-resource element and then get these properties in a JavaMail Session object later. Every property name must start with a mail- prefix. Sun Java System Application Server changes the dash (-) character to a period (.) in the name of the property, then saves the property to the MailConfiguration and JavaMail Session objects. If the name of the property doesn't start with mail-, the property is ignored.

For example, to define the property mail.password in a JavaMail Session object, first edit domain.xml as follows:

```
...
<mail-resource jndi-name="mail/Session" ...>
  <property name="mail-password" value="adminadmin"/>
</mail-resource>
...
```

After getting the JavaMail Session object, get the mail.password property to retrieve the value adminadmin, as follows:

```
String password = session.getProperty("mail.password");
```

manager-properties

Specifies session manager properties.

Superelements

[session-manager](#)

Subelements

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

Table 1-98 `manager-properties` Subelements

Element	Required	Description
property	zero or more	Specifies a property, which has a name and a value.

Attributes

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

Table 1-99 `manager-properties` Attributes

Attribute	Default	Description
<code>session-file-name</code>	none; state is not preserved across restarts	(optional) 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. Applicable only if the <code>persistence-type</code> attribute of the web-container-availability element is <code>memory</code> .
<code>reap-interval-in-seconds</code>	60	(optional) Specifies the time between checks for expired sessions. If the <code>persistence-type</code> attribute of the web-container-availability element is <code>file</code> or <code>ha</code> , sessions are passivated if <code>max-sessions</code> has been exceeded. If the <code>persistence-frequency</code> attribute of the web-container-availability element is set to <code>time-based</code> , active sessions are stored at this interval. 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 website, or you could lose the last few hits each time you restart the server.
<code>max-sessions</code>	-1	(optional) Specifies the maximum number of sessions that can be in cache, or -1 for no limit. After this, an attempt to create a new session causes an <code>IllegalStateException</code> to be thrown. If the <code>persistence-type</code> attribute of the web-container-availability element is <code>file</code> or <code>ha</code> , the session manager passivates sessions to the persistent store when this maximum is reached.
<code>session-id-generator-classname</code>	internal class generator	(optional) Specifies the name of the class that generates session IDs.

mdb-container

Configures the message-driven bean (MDB) container.

Superelements

[config](#)

Subelements

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

Table 1-100 `mdb-container` Subelements

Element	Required	Description
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `mdb-container` element.

Table 1-101 `mdb-container` Attributes

Attribute	Default	Description
<code>steady-pool-size</code>	10	(optional) Specifies the initial and minimum number of beans maintained in the pool.
<code>pool-resize-quantity</code>	2	(optional) Specifies the number of beans to be created if a request arrives when the pool is empty (subject to the <code>max-pool-size</code> limit), or the number of beans to remove if idle for more than <code>idle-timeout-in-seconds</code> .
<code>max-pool-size</code>	60	(optional) Specifies the maximum number of beans that can be created to satisfy client requests.
<code>idle-timeout-in-seconds</code>	600	(optional) Specifies the maximum time that a bean can remain idle in the pool. After this amount of time, the bean is destroyed. A value of 0 means a bean can remain idle indefinitely.

Properties

The following table describes properties for the `mdb-container` element.

Table 1-102 `mdb-container` Properties

Property	Default	Description
<code>cmt-max-runtime-exceptions</code>	1	Specifies the maximum number of <code>RuntimeException</code> occurrences allowed from a message-driven bean's <code>onMessage()</code> method when container-managed transactions are used. Deprecated.

message-security-config

Specifies configurations for message security providers.

Superelements

[security-service](#)

Subelements

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

Table 1-103 `message-security-config` Subelements

Element	Required	Description
provider-config	one or more	Specifies a configuration for one message security provider.

Attributes

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

Table 1-104 `message-security-config` Attributes

Attribute	Default	Description
<code>auth-layer</code>	none	Specifies the message layer at which authentication is performed. The value must be <code>SOAP</code> .
<code>default-provider</code>	none	(optional) Specifies the server provider that is invoked for any application not bound to a specific server provider.
<code>default-client-provider</code>	none	(optional) Specifies the client provider that is invoked for any application not bound to a specific client provider.

module-log-levels

Controls the level of messages logged by server subsystems to the server log. Allowed values of each subsystem attribute are, from highest to lowest: `FINEST`, `FINER`, `FINE`, `CONFIG`, `INFO`, `WARNING`, `SEVERE`, and `OFF`. Each value logs all messages for all lower values. The default value is `INFO`, which logs all `INFO`, `SEVERE`, and `WARNING` messages.

Superelements

[log-service](#)

Subelements

The following table describes subelements for the `module-log-levels` element.

Table 1-105 `module-log-levels` Subelements

Element	Required	Description
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `module-log-levels` element.

Table 1-106 `module-log-levels` Attributes

Attribute	Default	Description
<code>root</code>	<code>INFO</code>	(optional) Specifies the default level of messages logged by the entire Sun Java System Application Server installation.
<code>server</code>	<code>INFO</code>	(optional) Specifies the default level of messages logged by the server instance.
<code>ejb-container</code>	<code>INFO</code>	(optional) Specifies the level of messages logged by the EJB container.
<code>cmp-container</code>	<code>INFO</code>	(optional) Specifies the level of messages logged by the CMP subsystem of the EJB container.
<code>mdb-container</code>	<code>INFO</code>	(optional) Specifies the level of messages logged by the MDB container.
<code>web-container</code>	<code>INFO</code>	(optional) Specifies the level of messages logged by the web container.
<code>classloader</code>	<code>INFO</code>	(optional) Specifies the level of messages logged by the classloader hierarchy.

Table 1-106 module-log-levels Attributes (*Continued*)

Attribute	Default	Description
configuration	INFO	(optional) Specifies the level of messages logged by the configuration subsystem.
naming	INFO	(optional) Specifies the level of messages logged by the naming subsystem.
security	INFO	(optional) Specifies the level of messages logged by the security subsystem.
jts	INFO	(optional) Specifies the level of messages logged by the Java Transaction Service.
jta	INFO	(optional) Specifies the level of messages logged by the Java Transaction API.
admin	INFO	(optional) Specifies the level of messages logged by the Administration Console subsystem.
deployment	INFO	(optional) Specifies the level of messages logged by the deployment subsystem.
verifier	INFO	(optional) Specifies the level of messages logged by the deployment descriptor verifier.
jaxr	INFO	(optional) Specifies the level of messages logged by the XML registry.
jaxrpc	INFO	(optional) Specifies the level of messages logged by the XML RPC module.
saaj	INFO	(optional) Specifies the level of messages logged by the SOAP with Attachments API for Java module.
corba	INFO	(optional) Specifies the level of messages logged by the ORB.
javamail	INFO	(optional) Specifies the level of messages logged by the JavaMail subsystem.
jms	INFO	(optional) Specifies the level of messages logged by the Java Message Service.
connector	INFO	(optional) Specifies the level of messages logged by the connector subsystem.
jdo	INFO	(optional) Specifies the level of messages logged by the Java Data Objects module.
cmp	INFO	(optional) Specifies the level of messages logged by the CMP subsystem.

Table 1-106 module-log-levels Attributes (*Continued*)

Attribute	Default	Description
util	INFO	(optional) Specifies the level of messages logged by the utility subsystem.
resource-adapter	INFO	(optional) Specifies the level of messages logged by the resource adapter subsystem.
synchronization	INFO	(optional) Specifies the level of messages logged by the synchronization subsystem.
node-agent	INFO	(optional) Specifies the level of messages logged by the node agent subsystem.

module-monitoring-levels

Controls the level of monitoring of server subsystems. Allowed values of each subsystem attribute are LOW, HIGH, and OFF.

Superelements

[monitoring-service](#)

Subelements

The following table describes subelements for the module-monitoring-levels element.

Table 1-107 module-monitoring-levels Subelements

Element	Required	Description
property	zero or more	Specifies a property or a variable.

Attributes

Table 1-108 module-monitoring-levels Attributes

Attribute	Default	Description
thread-pool	OFF	(optional) Specifies the level of monitoring of the thread pool subsystem.
orb	OFF	(optional) Specifies the level of monitoring of the ORB.
ejb-container	OFF	(optional) Specifies the level of monitoring of the EJB container.
web-container	OFF	(optional) Specifies the level of monitoring of the web container.
transaction-service	OFF	(optional) Specifies the level of monitoring of the transaction service.

Table 1-108 module-monitoring-levels Attributes (*Continued*)

Attribute	Default	Description
http-service	OFF	(optional) Specifies the level of monitoring of the HTTP service.
jdbc-connection-pool	OFF	(optional) Specifies the level of monitoring of the JDBC connection pool subsystem.
connector-connection-pool	OFF	(optional) Specifies the level of monitoring of the connector connection pool subsystem.
connector-service	OFF	(optional) Specifies the level of monitoring of the connector service.
jms-service	OFF	(optional) Specifies the level of monitoring of the JMS service.
jvm	OFF	(optional) Specifies the level of monitoring of the JVM.

monitoring-service

Configures the monitoring service.

Superelements

[config](#)

Subelements

The following table describes subelements for the monitoring-service element.

Table 1-109 monitoring-service Subelements

Element	Required	Description
module-monitoring-levels	zero or one	Controls the level of monitoring of server subsystems.
property	zero or more	Specifies a property or a variable.

node-agent

Defines a node agent, which manages server instances on a host machine.

Superelements

[node-agents](#)

Subelements

The following table describes subelements for the `node-agent` element.

Table 1-110 `node-agent` Subelements

Element	Required	Description
jmx-connector	zero or one	Configures a JSR 160 compliant remote JMX connector.
auth-realm	zero or one	Defines a realm for authentication.
log-service	only one	Configures the system logging service.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `node-agent` element.

Table 1-111 `node-agent` Attributes

Attribute	Default	Description
<code>name</code>	<code>none</code>	Specifies the node agent name.
<code>system-jmx-connector-name</code>	<code>none</code>	Specifies the name of the internal jmx-connector .
<code>start-servers-in-startup</code>	<code>true</code>	(optional) If <code>true</code> , starts all managed server instances when the node agent is started.

Properties

The following table describes properties for the `node-agent` element.

Table 1-112 `node-agent` Properties

Property	Default	Description
<code>INSTANCE-SYNC-JVM-OPTIONS</code>	<code>default</code> Application Server JVM options	Sets options for the server instance synchronization JVM. Setting options that limit memory usage helps prevent <code>OutOfMemory</code> errors when large applications are synchronized or when memory is constrained. For more information, see the <i>Sun Java System Application Server Administration Guide</i> .

node-agents

Contains node agents.

Superelements

[domain](#)

Subelements

The following table describes subelements for the `node-agents` element.

Table 1-113 `node-agents` Subelements

Element	Required	Description
node-agent	zero or more	Defines a node agent, which manages server instances on a host machine.

orb

Configures the ORB.

To enable SSL for outbound connections, include an `ssl-client-config` subelement in the parent `iiop-service` element.

Superelements

[iiop-service](#)

Subelements

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

Table 1-114 `orb` Subelements

Element	Required	Description
property	zero or more	Specifies a property or a variable.

Attributes

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

Table 1-115 orb Attributes

Attribute	Default	Description
use-thread-pool-ids	none	Specifies a comma-separated list of <code>thread-pool-id</code> values defined in thread-pool elements used by the ORB.
message-fragment-size	1024	(optional) GIOPv1.2 messages larger than this number of bytes are fragmented.
max-connections	1024	(optional) The maximum number of incoming connections on all IIOp listeners. Legal values are integers.

P

persistence-manager-factory-resource

Defines a persistence manager factory resource for container-managed persistence (CMP).

Superelements

[resources](#)

Subelements

The following table describes subelements for the `persistence-manager-factory-resource` element.

Table 1-116 persistence-manager-factory-resource Subelements

Element	Required	Description
description	zero or one	Contains a text description of this element.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `persistence-manager-factory-resource` element.

Table 1-117 persistence-manager-factory-resource Attributes

Attribute	Default	Description
jndi-name	none	Specifies the JNDI name for the resource.

Table 1-117 persistence-manager-factory-resource Attributes (*Continued*)

Attribute	Default	Description
factory-class	com.sun.jdo.spi.persistence.support.sqlstore.impl.PersistenceManagerFactoryImpl	(optional) Deprecated. Do not specify this attribute for the built-in CMP implementation.
jdbc-resource-jndi-name	none	Specifies the <code>jdbc-resource</code> from which database connections are obtained. Must be the <code>jndi-name</code> of an existing <code>jdbc-resource</code> .
object-type	user	(optional) Defines the type of the resource. Allowed values are: <ul style="list-style-type: none"> <code>system-all</code> - A system resource for all server instances and the domain application server. <code>system-admin</code> - A system resource only for the the domain application server. <code>system-instance</code> - A system resource for all server instances only. <code>user</code> - A user resource.
enabled	true	(optional) Determines whether this resource is enabled at runtime.

principal

Contains the principal of the servlet or EJB client.

Superelements

[security-map](#)

Subelements

none - contains data

profiler

Configures a profiler for use with Sun Java System Application Server. For more information about profilers, see the *Sun Java System Application Server Developer's Guide*.

Superelements

[java-config](#)

Subelements

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

Table 1-118 profiler Subelements

Element	Required	Description
<code>jvm-options</code>	zero or more	Contains profiler-specific JVM command line options.
<code>property</code>	zero or more	Specifies a property or a variable.

NOTE Subelements of a `profiler` element can occur in any order.

Attributes

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

Table 1-119 profiler Attributes

Attribute	Default	Description
<code>name</code>	none	Specifies the name of the profiler.
<code>classpath</code>	none	(optional) Specifies the classpath for the profiler.
<code>native-library-path</code>	none	(optional) Specifies the native library path for the profiler.
<code>enabled</code>	true	(optional) Determines whether the profiler is enabled.

property

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

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

For example, an `auth-realm` element can include `property` subelements:

```
<auth-realm name="file"
  classname="com.sun.enterprise.security.auth.realm.file.FileRealm">
  <property name="file" value="domain_dir/config/keyfile"/>
  <property name="jaas-context" value="fileRealm"/>
</auth-realm>
```

Which properties an `auth-realm` element uses depends on the value of the `auth-realm` element's `name` attribute. The `file` realm uses `file` and `jaas-context` properties. Other realms use different properties.

Superelements

`admin-object-resource`, `admin-service`, `alert-service`, `audit-module`, `auth-realm`, `availability-service`, `cluster`, `config`, `connector-connection-pool`, `connector-resource`, `custom-resource`, `das-config`, `domain`, `ejb-container`, `ejb-container-availability`, `ejb-timer-service`, `external-jndi-resource`, `filter-config`, `http-listener`, `http-service`, `iiop-listener`, `jacc-provider`, `java-config`, `jdbc-connection-pool`, `jdbc-resource`, `jms-host`, `jms-service`, `jmx-connector`, `lb-config`, `lifecycle-module`, `listener-config`, `log-service`, `mail-resource`, `manager-properties`, `mdb-container`, `module-log-levels`, `module-monitoring-levels`, `monitoring-service`, `node-agent`, `orb`, `persistence-manager-factory-resource`, `profiler`, `provider-config`, `resource-adapter-config`, `security-service`, `server`, `session-properties`, `store-properties`, `transaction-service`, `virtual-server`, `web-container`, `web-container-availability`

Subelements

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

Table 1-120 `property` Subelements

Element	Required	Description
<code>description</code>	zero or one	Contains a text description of this element.

Attributes

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

Table 1-121 `property` Attributes

Attribute	Default	Description
<code>name</code>	<code>none</code>	Specifies the name of the property or variable.
<code>value</code>	<code>none</code>	Specifies the value of the property or variable.

provider-config

Specifies a configuration for one message security provider.

Although the `request-policy` and `response-policy` subelements are optional, the `provider-config` element does nothing if they are not specified.

Use property subelements to configure provider-specific properties. Property values are passed to the provider when its `initialize` method is called.

Superelements

[message-security-config](#)

Subelements

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

Table 1-122 `provider-config` Subelements

Element	Required	Description
request-policy	zero or one	Defines the authentication policy requirements of the authentication provider's request processing.
response-policy	zero or one	Defines the authentication policy requirements of the authentication provider's response processing.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `provider-config` element.

Table 1-123 `provider-config` Attributes

Attribute	Default	Description
<code>provider-id</code>	none	Specifies the provider ID.
<code>provider-type</code>	none	Specifies whether the provider is a client, server, or client-server authentication provider.
<code>class-name</code>	none	Specifies the Java implementation class of the provider. Client authentication providers must implement the <code>com.sun.enterprise.security.jauth.ClientAuthModule</code> interface. Server authentication providers must implement the <code>com.sun.enterprise.security.jauth.ServerAuthModule</code> interface. Client-server providers must implement both interfaces.

R

request-policy

Defines the authentication policy requirements of the authentication provider's request processing.

Superelements

[provider-config](#)

Subelements

none

Attributes

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

Table 1-124 `request-policy` Attributes

Attribute	Default	Description
<code>auth-source</code>	none	Specifies the type of required authentication, either <code>sender</code> (user name and password) or <code>content</code> (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> .

request-processing

Configures request processing threads.

Superelements

[http-service](#)

Subelements

none

Attributes

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

Table 1-125 request-processing Attributes

Attribute	Default	Description
thread-count	128	(optional) Specifies the maximum number of request processing threads.
initial-thread-count	48	(optional) Specifies the number of request processing threads that are available when the server starts up.
thread-increment	10	(optional) Specifies the number of request processing threads added when the number of requests exceeds the <code>initial-thread-count</code> .
request-timeout-in-seconds	30	(optional) Specifies the time at which the request times out.
header-buffer-in-bytes	4096	(optional) Specifies the size of the buffer used by the request processing threads to read the request data.

resource-adapter-config

Defines a connector (resource adapter) configuration. Stores configuration information for the resource adapter JavaBean in property subelements.

Superelements

[resources](#)

Subelements

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

Table 1-126 resource-adapter-config Subelements

Element	Required	Description
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `resource-adapter-config` element.

Table 1-127 resource-adapter-config Attributes

Attribute	Default	Description
name	none	(optional) Not used. See <code>resource-adapter-name</code> .

Table 1-127 resource-adapter-config Attributes (*Continued*)

Attribute	Default	Description
thread-pool-ids	none	(optional) Specifies the id of a thread-pool element.
object-type	user	(optional) Defines the type of the resource. Allowed values are: <ul style="list-style-type: none"> • <code>system-all</code> - A system resource for all server instances and the domain application server. • <code>system-admin</code> - A system resource only for the the domain application server. • <code>system-instance</code> - A system resource for all server instances only. • <code>user</code> - A user resource.
resource-adapter-name	none	Specifies the name attribute of a deployed connector-module . If the resource adapter is embedded in an application, then it is <code>app_name#rar_name</code> .

Properties

Properties of the `resource-adapter-config` element are the names of setter methods of the `resourceadapter-class` element in the `ra.xml` file, which defines the class name of the resource adapter JavaBean. Any properties defined here override the default values present in `ra.xml`.

resource-ref

References a resource deployed to the server instance.

Superelements

[cluster](#), [server](#)

Subelements

none

Attributes

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

Table 1-128 resource-ref Attributes

Attribute	Default	Description
enabled	true	(optional) Determines whether the resource is enabled.
ref	none	References the name attribute of a custom-resource , external-jndi-resource , jdbc-resource , mail-resource , persistence-manager-factory-resource , admin-object-resource , connector-resource , resource-adapter-config , jdbc-connection-pool , or connector-connection-pool element.

resources

Contains configured resources, such as database connections, JavaMail™ sessions, and so on.

NOTE You must specify a Java Naming and Directory Interface™ (JNDI) name for each resource. To avoid collisions with names of other enterprise resources in JNDI, and to avoid portability problems, all names in a Sun Java System Application Server application should begin with the string `java:comp/env`.

Superelements

[domain](#)

Subelements

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

Table 1-129 resources Subelements

Element	Required	Description
custom-resource	zero or more	Defines a custom resource.
external-jndi-resource	zero or more	Defines a resource that resides in an external JNDI repository.
jdbc-resource	zero or more	Defines a JDBC (Java Database Connectivity) resource.
mail-resource	zero or more	Defines a JavaMail resource.
persistence-manager-factory-resource	zero or more	Defines a persistence manager factory resource for CMP.

Table 1-129 resources Subelements (*Continued*)

Element	Required	Description
admin-object-resource	zero or more	Defines an administered object for an inbound resource adapter.
connector-resource	zero or more	Defines a connector (resource adapter) resource.
resource-adapter-config	zero or more	Defines a resource adapter configuration.
jdbc-connection-pool	zero or more	Defines the properties that are required for creating a JDBC connection pool.
connector-connection-pool	zero or more	Defines the properties that are required for creating a connector connection pool.

NOTE Subelements of a `resources` element can occur in any order.

response-policy

Defines the authentication policy requirements of the authentication provider's response processing.

Superelements

[provider-config](#)

Subelements

none

Attributes

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

Table 1-130 response-policy Attributes

Attribute	Default	Description
<code>auth-source</code>	none	Specifies the type of required authentication, either <code>sender</code> (user name and password) or <code>content</code> (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> .

S

security-map

Maps the principal received during servlet or EJB authentication to the credentials accepted by the EIS.

Superelements

[connector-connection-pool](#)

Subelements

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

Table 1-131 `security-map` Subelements

Element	Required	Description
principal	one or more	Contains the principal of the servlet or EJB client.
user-group	one or more	Contains the group to which the principal belongs.
backend-principal	only one	Specifies the user name and password required by the EIS.

Attributes

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

Table 1-132 `security-map` Attributes

Attribute	Default	Description
<code>name</code>	<code>none</code>	Specifies a name for the security mapping.

security-service

Defines parameters and configuration information needed by the J2EE security service.

Superelements

[config](#)

Subelements

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

Table 1-133 security-service Subelements

Element	Required	Description
auth-realm	one or more	Defines a realm for authentication.
jacc-provider	one or more	Specifies a Java Authorization Contract for Containers (JACC) provider for pluggable authorization.
audit-module	zero or more	Specifies an optional plug-in module that implements audit capabilities.
message-security-config	zero or more	Specifies configurations for message security providers.
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `security-service` element.

Table 1-134 security-service Attributes

Attribute	Default	Description
<code>default-realm</code>	<code>file</code>	(optional) Specifies the active authentication realm (an <code>auth-realm</code> name attribute) for this server instance.
<code>default-principal</code>	<code>none</code>	(optional) Used as the identity of the default security context when necessary and when no principal is provided. This attribute need not be set for normal server operation.
<code>default-principal-password</code>	<code>none</code>	(optional) The password of the default principal. This attribute need not be set for normal server operation.
<code>anonymous-role</code>	<code>ANYONE</code>	(optional) Used as the name for default, or anonymous, role. The anonymous role is always assigned to all principals. This role value can be used in J2EE deployment descriptors to grant access to anyone.
<code>audit-enabled</code>	<code>false</code>	(optional) If <code>true</code> , additional access logging is performed to provide audit information. Audit information consists of: <ul style="list-style-type: none"> • Authentication success and failure events • Servlet and EJB access grants and denials
<code>jacc</code>	<code>default</code>	(optional) Specifies the name of the jacc-provider element to use for setting up the JACC infrastructure. Do not change the default value unless you are adding a custom JACC provider.

Table 1-134 security-service Attributes (*Continued*)

Attribute	Default	Description
audit-modules	default	(optional) Specifies a comma-separated list of audit provider modules used by the audit subsystem. The default value refers to the internal log-based audit module.

server

Defines a server instance.

NOTE Server instances are not the same thing as virtual servers. Each server instance is a completely separate server that contains one or more virtual servers.

Superelements

[servers](#)

Subelements

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

Table 1-135 server Subelements

Element	Required	Description
application-ref	zero or more	References an application or module deployed to the server instance.
resource-ref	zero or more	References a resource deployed to the server instance.
system-property	zero or more	Specifies a system property.
property	zero or more	Specifies a property or a variable.

Attributes

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

Table 1-136 server Attributes

Attribute	Default	Description
name	none	Specifies the name of the server instance.

Table 1-136 *server* Attributes (Continued)

Attribute	Default	Description
<code>config-ref</code>	default <code>config</code> element's name, <code>server-config</code>	(optional) References the name of the <code>config</code> used by the server instance.
<code>node-agent-ref</code>	node agent created when the server instance was created	(optional) References the name of the <code>node-agent</code> used by the server instance.

server-ref

References a server instance.

Superelements

`cluster`, `lb-config`

Subelements

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

Table 1-137 *server-ref* Subelements

Element	Required	Description
<code>health-checker</code>	zero or one	Defines a health checker for the referenced server instance.

Attributes

The following table describes attributes for the `server-ref` element.

Table 1-138 *server-ref* Attributes

Attribute	Default	Description
<code>ref</code>	none	References the name attribute of a <code>server</code> element.
<code>disable-timeout-in-minutes</code>	30	(optional) Specifies the time it takes this server instance to reach a quiescent state after having been disabled.
<code>lb-enabled</code>	false	(optional) If <code>true</code> , all load-balancers consider this server instance available to them.
<code>enabled</code>	true	(optional) Determines whether the server instance is enabled.

servers

Contains server instances.

Superelements

[domain](#)

Subelements

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

Table 1-139 `servers` Subelements

Element	Required	Description
server	zero or more	Defines a server instance.

session-config

Specifies session configuration information for the entire web container. Individual web applications can override these settings using the corresponding elements in their `sun-web.xml` files.

Superelements

[web-container](#)

Subelements

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

Table 1-140 `session-config` Subelements

Element	Required	Description
session-manager	zero or one	Specifies session manager configuration information.
session-properties	zero or one	Specifies session properties.

session-manager

Specifies session manager information.

NOTE The session manager interface is unstable. An unstable interface might be experimental or transitional, and hence might change incompatibly, be removed, or be replaced by a more stable interface in the next release.

Superelements[session-config](#)**Subelements**

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

Table 1-141 `session-manager` Subelements

Element	Required	Description
manager-properties	zero or one	Specifies session manager properties.
store-properties	zero or one	Specifies session persistence (storage) properties.

session-properties

Specifies session properties.

Superelements[session-config](#)**Subelements**

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

Table 1-142 `session-properties` Subelements

Element	Required	Description
property	zero or more	Specifies a property, which has a name and a value.

Attributes**Table 1-143** `session-properties` Attributes

Attribute	Default	Description
<code>timeout-in-seconds</code>	600	<p>(optional) 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.</p> <p>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>timeout-in-seconds</code> value. If neither <code>session-timeout</code> nor <code>timeout-in-seconds</code> is specified, the <code>timeout-in-seconds</code> default is used.</p> <p>Note that the <code>session-timeout</code> element in <code>web.xml</code> is specified in minutes, not seconds.</p>

Properties

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

Table 1-144 `session-properties` Properties

Property	Default	Description
<code>enableCookies</code>	<code>true</code>	Uses cookies for session tracking if set to <code>true</code> .
<code>enableURLRewriting</code>	<code>true</code>	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 JavaServer Pages™ (JSP™) page.
<code>idLengthBytes</code>	128	Specifies the number of bytes in this web module's session ID.

ssl

Defines SSL (Secure Socket Layer) parameters.

An `ssl` element is required inside an `http-listener` or `iiop-listener` element that has its `security-enabled` attribute set to `on`.

The grandparent `http-service` element has properties that configure global SSL settings, and the `http-protocol` subelement of the grandparent `http-service` element has the `ssl-enabled` attribute, which globally enables SSL.

Superelements

[http-listener](#), [iiop-listener](#), [jmx-connector](#), [ssl-client-config](#)

Subelements

none

Attributes

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

Table 1-145 `ssl` Attributes

Attribute	Default	Description
<code>cert-nickname</code>	<code>none</code>	The nickname of the server certificate in the certificate database or the PKCS#11 token. In the certificate, the name format is <code>tokenname:nickname</code> . Including the <code>tokenname:</code> part of the name in this attribute is optional.

Table 1-145 ssl Attributes (*Continued*)

Attribute	Default	Description
ssl2-enabled	false	(optional) Determines whether SSL2 is enabled. 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.
ssl2-ciphers	none	(optional) A comma-separated list of the SSL2 ciphers used, with the prefix + to enable or - to disable, for example +rc4. Allowed values are rc4, rc4export, rc2, rc2export, idea, des, desede3.
ssl3-enabled	true	(optional) Determines whether SSL3 is enabled. The default is true. 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.
ssl3-tls-ciphers	none	(optional) A comma-separated list of the SSL3 ciphers used, with the prefix + to enable or - to disable, for example +rsa_des_sha. Allowed SSL3 values are rsa_rc4_128_md5, rsa_3des_sha, rsa_des_sha, rsa_rc4_40_md5, rsa_rc2_40_md5, rsa_null_md5. Allowed TLS values are rsa_des_56_sha, rsa_rc4_56_sha.
tls-enabled	true	(optional) Determines whether TLS is enabled.
tls-rollback-enabled	true	(optional) Determines whether TLS rollback is enabled. TLS rollback should be enabled for Microsoft Internet Explorer 5.0 and 5.5. For more information, see the <i>Sun Java System Application Server Administrator's Guide</i> .
client-auth-enabled	false	(optional) Determines whether SSL3 client authentication is performed on every request, independent of ACL-based access control.

ssl-client-config

Defines SSL parameters for the ORB when it makes outbound SSL connections and behaves as a client.

Superelements

[iiop-service](#)

Subelements

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

Table 1-146 `ssl-client-config` Subelements

Element	Required	Description
<code>ssl</code>	only one	Defines SSL parameters.

store-properties

Specifies session persistence (storage) properties.

Superelements

`session-manager`

Subelements

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

Table 1-147 `store-properties` Subelements

Element	Required	Description
<code>property</code>	zero or more	Specifies a property, which has a name and a value.

Attributes

Table 1-148 `store-properties` Attributes

Attribute	Default	Description
<code>directory</code>	<code>domain_dir/generated</code> <code>/jsp/j2ee-apps/</code> <code>appname/appname_war</code>	(optional) 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 <code>web-container-availability</code> element is <code>file</code> .

Table 1-148 store-properties Attributes (*Continued*)

Attribute	Default	Description
reap-interval-in-seconds	60	<p>(optional) Specifies the time between checks for expired sessions.</p> <p>If the <code>persistence-type</code> attribute of the <code>web-container-availability</code> element is <code>file</code> or <code>ha</code>, sessions are passivated if <code>maxSessions</code> has been exceeded.</p> <p>If the <code>persistence-frequency</code> attribute of the <code>web-container-availability</code> element is set to <code>time-based</code>, active sessions are stored at this interval.</p> <p>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 website, or you could lose the last few hits each time you restart the server.</p>

system-property

Specifies a system property. A system property defines a common value for a setting at one of these levels, from highest to lowest: [domain](#), [cluster](#), [server](#), or [config](#). A value set at a higher level can be overridden at a lower level. Some system properties are predefined; see [Table 1-151](#). You can also create system properties using this element.

The following example shows the use of a predefined system property:

```
<log-service file="{com.sun.aas.instanceRoot}/logs/server.log">
  <module-log-levels admin=INFO .../>
</log-service>
```

The following example shows the creation and use of a system property:

```
<config name="config1">
  ...
  <http-service>
    ...
    <http-listener id="ls1" host="0.0.0.0" port="{ls1-port}"/>
    ...
  </http-service>
  ...
  <system-property name="ls1-port" value="8080"/>
</config>
```

Superelements

[cluster](#), [config](#), [domain](#), [server](#)

Subelements

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

Table 1-149 `system-property` Subelements

Element	Required	Description
<code>description</code>	zero or one	Contains a text description of this element.

Attributes

The following table describes attributes for the `system-property` element.

Table 1-150 `system-property` Attributes

Attribute	Default	Description
<code>name</code>	none	Specifies the name of the system property.
<code>value</code>	none	Specifies the value of the system property.

Properties

The following table lists predefined system properties.

Table 1-151 Predefined System Properties

Property	Default	Description
<code>com.sun.aas.installRoot</code>	depends on operating system	Specifies the directory where the Sun Java System Application Server is installed.
<code>com.sun.aas.instanceRoot</code>	depends on operating system	Specifies the top level directory for a server instance.
<code>com.sun.aas.hostName</code>	none	Specifies the name of the host (machine).
<code>com.sun.aas.javaRoot</code>	depends on operating system	Specifies the J2SE installation directory.
<code>com.sun.aas.imqLib</code>	depends on operating system	Specifies the library directory for Sun Java System Message Queue.
<code>com.sun.aas.configName</code>	<code>server-config</code>	Specifies the name of the <code>config</code> used by a server instance.
<code>com.sun.aas.instanceName</code>	<code>server1</code>	Specifies the name of the server instance. This property is not used in the default configuration, but can be used to customize configuration.

Table 1-151 Predefined System Properties (*Continued*)

Property	Default	Description
com.sun.aas.clusterName	cluster1	Specifies the name of the cluster. This property is only set on clustered server instances. This property is not used in the default configuration, but can be used to customize configuration.
com.sun.aas.domainName	domain1	Specifies the name of the domain. This property is not used in the default configuration, but can be used to customize configuration.

T

thread-pool

Defines a thread pool.

Superelements

[thread-pools](#)

Subelements

none

Attributes

Table 1-152 thread-pool Attributes

Attribute	Default	Description
thread-pool-id	none	Specifies the thread pool ID.
min-thread-pool-size	0	(optional) Specifies the minimum number of threads in the pool. These are created when the thread pool is instantiated.
max-thread-pool-size	200	(optional) Specifies the maximum number of threads the pool can contain.
idle-thread-timeout-in-seconds	120	(optional) Specifies the amount of time after which idle threads are removed from the pool.
num-work-queues	1	(optional) Specifies the total number of work queues serviced by this thread pool.

thread-pools

Contains thread pools.

Superelements

[config](#)

Subelements

The following table describes subelements for the `thread-pools` element.

Table 1-153 `thread-pools` Subelements

Element	Required	Description
thread-pool	one or more	Defines a thread pool.

transaction-service

Configures the Java Transaction Service (JTS).

Superelements

[config](#)

Subelements

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

Table 1-154 `transaction-service` Subelements

Element	Required	Description
property	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `transaction-service` element.

Table 1-155 `transaction-service` Attributes

Attribute	Default	Description
<code>automatic-recovery</code>	<code>true</code>	(optional) If <code>true</code> , the server instance attempts transaction recovery during startup.
<code>timeout-in-seconds</code>	<code>0</code>	(optional) Specifies the amount of time after which the transaction is aborted. If set to <code>0</code> , the transaction never times out.

Table 1-155 transaction-service Attributes (*Continued*)

Attribute	Default	Description
tx-log-dir	directory specified by the log-root attribute of the domain element	(optional) Specifies the parent directory of the transaction log directory tx. The directory in which the transaction logs are kept must be writable by the user account under which the server runs. A relative path is relative to the log-root attribute of the domain element.
heuristic-decision	rollback	(optional) If the outcome of a distributed transaction cannot be determined because other participants are unreachable, this property determines the outcome. Allowed values are rollback and commit.
retry-timeout-in-seconds	600	(optional) Determines the retry time in the following scenarios: <ul style="list-style-type: none"> At the transaction recovery time, if resources are unreachable. If there are any transient exceptions in the second phase of a two phase commit protocol. A negative value specifies infinite retries. A value of 0 (zero) specifies no retries. A positive value indicates the time after which a retry is attempted.
keypoint-interval	2048	(optional) Specifies the number of transactions between keypoint operations in the log. Keypoint operations reduce the size of the transaction log file by compressing it. A larger value for this attribute (for example, 4096) results in a larger transaction log file, but fewer keypoint operations and potentially better performance. A smaller value (for example, 100) results in smaller log files, but slightly reduced performance due to the greater frequency of keypoint operations.

Properties

The following table describes properties for the transaction-service element.

Table 1-156 transaction-service Properties

Property	Default	Description
oracle-xa-recovery-workaround	true	If true, the Oracle XA Resource workaround is used in transaction recovery.
disable-distributed-transaction-logging	false	If true, disables transaction logging, which might improve performance. If the automatic-recovery attribute is set to true, this property is ignored.

Table 1-156 transaction-service Properties (*Continued*)

Property	Default	Description
xaresource-txn-timeout	specific to the XAResource used	Changes the XAResource timeout. In some cases, the XAResource default timeout can cause transactions to be aborted, so it is desirable to change it.
pending-txn-cleanup-interval	none if this property is absent, 60 if this property is present but has no value	Specifies the interval, in seconds, at which an asynchronous thread checks for pending transactions and completes them.
use-last-agent-optimization	true	If true, enables last agent optimization, which improves the throughput of transactions. If one non-XA resource is used with XA resources in the same transaction, the non XA resource is the last agent.

U

user-group

Contains the group to which the principal belongs.

Superelements

[security-map](#)

Subelements

none - contains data

virtual-server

Defines a virtual server. A virtual server, also called a virtual host, is a virtual web server that serves content targeted for a specific URL. Multiple virtual servers can serve content using the same or different host names, port numbers, or IP addresses. The HTTP service can direct incoming web requests to different virtual servers based on the URL.

When the Sun Java System Application Server is first installed, a default virtual server is created. (You can also assign a default virtual server to each new `http-listener` you create.)

NOTE Virtual servers are not the same thing as server instances. Each server instance is a completely separate server that contains one or more virtual servers.

Before the Sun Java System Application Server can process a request, it must accept the request via a listener, then direct the request to the correct virtual server. The virtual server is determined as follows:

- If the listener is configured to only a default virtual server, that virtual server is selected.
- If the listener has more than one virtual server configured to it, the request `Host` header is matched to the `hosts` attribute of a virtual server. If no `Host` header is present or no `hosts` attribute matches, the default virtual server for the listener is selected.

If a virtual server is configured to an SSL listener, its `hosts` attribute is checked against the subject pattern of the certificate at server startup, and a warning is generated and written to the server log if they don't match.

Superelements

`http-service`

Subelements

The following table describes subelements for the `virtual-server` element.

Table 1-157 `virtual-server` Subelements

Element	Required	Description
<code>http-access-log</code>	zero or one	Defines an access log file.

Table 1-157 `virtual-server` Subelements (*Continued*)

Element	Required	Description
<code>property</code>	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `virtual-server` element.

Table 1-158 `virtual-server` Attributes

Attribute	Default	Description
<code>id</code>	none	Virtual server ID. This is a unique ID that allows lookup of a specific virtual server. A virtual server ID cannot begin with a number.
<code>http-listeners</code>	none	(optional) In a comma-separated list, references <code>id</code> attributes of <code>http-listener</code> elements that specify the connection(s) the virtual server uses. Required only for a <code>virtual-server</code> that is not referenced by the <code>default-virtual-server</code> attribute of an <code>http-listener</code> .
<code>default-web-module</code>	system default web module	(optional) References the name attribute of the default <code>web-module</code> for this virtual server, which responds to requests that cannot be resolved to other web modules deployed to this virtual server (see the <code>application-ref</code> element).
<code>hosts</code>	none	A comma-separated list of values, each of which selects the current virtual server when included in the <code>Host</code> request header. Two or more <code>virtual-server</code> elements that reference or are referenced by the same <code>http-listener</code> cannot have any <code>hosts</code> values in common.
<code>state</code>	on	(optional) Determines whether a <code>virtual-server</code> is active (<code>on</code>) or inactive (<code>off</code> , <code>disabled</code>). The default is <code>on</code> (active). When inactive, a <code>virtual-server</code> does not service requests. If a <code>virtual-server</code> is disabled, only the global server administrator can turn it <code>on</code> .
<code>docroot</code>	none	(optional) Specifies the document root for this virtual server.
<code>log-file</code>	<code>server.log</code> in the directory specified by the <code>log-root</code> attribute of the <code>domain</code> element	(optional) Writes this virtual server's log messages to a log file separate from the server log. The file and directory in which the virtual server log is kept must be writable by the user account under which the server runs. See the <code>log-service</code> description for details about logs.

Properties

The following table describes properties for the `virtual-server` element.

Table 1-159 virtual-server Properties

Property	Default	Description
sso-enabled	true	If <code>true</code> , single sign-on is enabled for web applications on this virtual server that are configured for the same realm. If <code>false</code> , single sign-on is disabled for this virtual server, and users must authenticate separately to every application on the virtual server.
sso-max-inactive-seconds	300	Specifies the time after which a user's single sign-on record becomes eligible for purging if no client activity is received. Since single sign-on applies across several applications on the same virtual server, access to any of the applications keeps the single sign-on record active. Higher values provide longer single sign-on persistence for the users at the expense of more memory use on the server.
sso-reap-interval-seconds	60	Specifies the interval between purges of expired single sign-on records.
setCacheControl	none	Specifies a comma-separated list of <code>Cache-Control</code> response directives. For a list of valid directives, see section 14.9 of the following document: http://www.ietf.org/rfc/rfc2616.txt

W

web-container

Configures the web container.

Superelements

[config](#)

Subelements

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

Table 1-160 web-container Subelements

Element	Required	Description
session-config	zero or one	Specifies session configuration information for the web container.
property	zero or more	Specifies a property or a variable.

web-container-availability

Enables availability in the web container, including HTTP session persistence to the high-availability database (HADB).

If availability is disabled, there is no high availability for HTTP session persistence. In other words, `persistence-type=memory`.

If availability is enabled but no other `web-container-availability` attributes are specified, the default session persistence configuration is as follows:

`persistence-type=ha`

`persistence-frequency=time-based`

`persistence-scope=session`

The default configuration for all applications can be changed by setting the various `web-container-availability` attributes and properties.

You can override the various `web-container-availability` attributes and properties for a specific application in `sun-web.xml`. For details, see the *Sun Java System Application Server Developer's Guide*.

Superelements

`availability-service`

Subelements

The following table describes subelements for the `web-container-availability` element.

Table 1-161 `web-container-availability` Subelements

Element	Required	Description
<code>property</code>	zero or more	Specifies a property or a variable.

Attributes

The following table describes attributes for the `web-container-availability` element.

Table 1-162 web-container-availability Attributes

Attribute	Default	Description
availability-enabled	true	(optional) If set to <code>true</code> , and if availability is enabled for the server instance (see availability-service), high-availability features apply to all web applications deployed to the server instance that do not have availability disabled. All instances in a cluster should have the same availability value to ensure consistent behavior.
persistence-type	memory	<p>(optional) Specifies the session persistence mechanism for web applications that have availability enabled. Allowed values are <code>memory</code> (no persistence) <code>file</code> (the file system) and <code>ha</code> (the HADB). For production environments that require session persistence, use <code>ha</code>.</p> <p>If set to <code>memory</code>, the manager-properties element's <code>sessionFilename</code> property specifies the file system location where the HTTP session state is stored if the server instance is gracefully shut down. This is useful for internal testing but is not supported for production environments.</p> <p>If set to <code>file</code>, the store-properties element's <code>directory</code> property specifies the file system location where the HTTP session state is stored. Persisting to the file system is useful for internal testing but is not supported for production environments.</p>
persistence-frequency	web-method	<p>(optional) Specifies how often the session state is stored. Applicable only if the <code>persistence-type</code> is <code>ha</code>. Allowed values are as follows:</p> <ul style="list-style-type: none"> • <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. • <code>time-based</code> - The session state is stored in the background at the frequency set by the <code>reapIntervalSeconds</code> property. 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. •

Table 1-162 web-container-availability Attributes (*Continued*)

Attribute	Default	Description
<code>persistence-scope</code>	<code>session</code>	<p>(optional) Specifies how much of the session state is stored. Applicable only if the <code>persistence-type</code> is <code>ha</code>. Allowed values are as follows:</p> <ul style="list-style-type: none"> <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. <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 J2EE specification requirement, but it is required for this mode to work properly. <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.
<code>sso-failover-enabled</code>	<code>false</code>	<p>(optional) If <code>true</code>, the single sign-on state is highly available. To enable single sign-on, use the <code>sso-enabled</code> property of the <code>virtual-server</code> element.</p>
<code>http-session-store-pool-name</code>	<p>availability-service element's <code>store-pool-name</code> attribute value</p>	<p>(optional) Specifies the <code>jndi-name</code> of the <code>jdbc-resource</code> used for connections to the HADB for session persistence. For more information about setting up a connection pool and JDBC resource for the HADB, see the description of the <code>configure-ha-cluster</code> command in the <i>Sun Java System Application Server Reference Manual</i>.</p>

If the `persistence-scope` attribute is set to `modified-attribute`, your 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.

web-module

Specifies a deployed web module.

Superelements

[applications](#)

Subelements

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

Table 1-163 `web-module` Subelements

Element	Required	Description
description	zero or one	Contains a text description of this element.

Attributes

The following table describes attributes for the `web-module` element.

Table 1-164 `web-module` Attributes

Attribute	Default	Description
<code>name</code>	<code>none</code>	The name of the web module.
<code>context-root</code>	<code>none</code>	The context root at which the web module is deployed. The context root can be the empty string or just <code>/</code> . The context root can start with the <code>/</code> character, but doesn't have to. For load balancing to work, web module context roots must be unique within a cluster. See the <i>Sun Java System Application Server Administrator's Guide</i> for more information about load balancing.
<code>location</code>	<code>none</code>	A fully qualified or relative path to the directory to which the contents of the <code>.war</code> file have been extracted. If relative, it is relative to the following directory: <code>domain_dir/applications/j2ee-modules/</code>
<code>object-type</code>	<code>user</code>	(optional) Defines the type of the resource. Allowed values are: <ul style="list-style-type: none"> <code>system-all</code> - A system resource for all server instances and the domain application server. <code>system-admin</code> - A system resource only for the the domain application server. <code>system-instance</code> - A system resource for all server instances only. <code>user</code> - A user resource.
<code>enabled</code>	<code>true</code>	(optional) Determines whether the web module is enabled.

Table 1-164 web-module Attributes (*Continued*)

Attribute	Default	Description
availability-enabled	false	(optional) Specifies whether availability is enabled in this web application for HTTP session persistence (and potentially passivation). Availability must also be enabled for the application or stand-alone web module during deployment. For more information about availability, see availability-service .
directory-deployed	false	(optional) Specifies whether the application has been deployed to a directory.

Index

A

- acceptor-threads attribute 57
- access log file 78
- access-log element 21
- address attribute 57, 60, 73
- AddressList property 38
- addresslist-behavior attribute 71
- addresslist-iterations attribute 71
- admin-object-resource element 22
- admin-password attribute 70
- admin-service element 23
- admin-session-timeout-in-minutes attribute 44
- admin-user-name attribute 70
- alarms attribute 79
- alert-service element 24
- alert-subscription element 25
- anonymous-role attribute 102
- applicant-module element 25
- append-version property 72
- application-ref element 26
- application-root attribute 46
- applications element 27
- applications, location 46
- assign-groups property 29
- audit-enabled attribute 102
- audit-module element 27
- audit-modules attribute 103
- auth-layer attribute 84
- auth-realm element 28

- auth-realm-name attribute 73
- auth-recipient attribute 96, 100
- auth-source attribute 96, 100
- autodeploy-dir attribute 44
- autodeploy-enabled attribute 44
- autodeploy-jsp-precompilation-enabled attribute 44
- autodeploy-polling-interval-in-seconds attribute 44
- autodeploy-verifier-enabled attribute 44
- automatic-recovery attribute 113
- availability-enabled attribute 31, 49, 50, 62, 120, 123
- availability-service element 30

B

- backend-principal element 31
- base-dn property 29
- blocking-enabled attribute 57
- bytecode-preprocessors attribute 65

C

- cache-idle-timeout-in-seconds attribute 48
- cache-resize-quantity attribute 47
- certificates 29
- cert-nickname attribute 107
- class-name attribute 76, 95
- classname attribute 28, 29

Section D

- classpath attribute 76, 93
- classpath-prefix attribute 64
- classpath-suffix attribute 64
- client-auth-enabled attribute 108
- client-authentication-required attribute 61
- ClientId property 38
- cluster element 32
- cluster-ref element 33
- clusters element 33
- CMP 91
- cmt-max-runtime-exceptions property 84
- commit-option attribute 48
- config element 34
- config-ref attribute 32, 104
- configs element 35
- connection-definition-name attribute 37
- connection-pool element 35
- connection-validation-method attribute 67
- connector-connection-pool element 36
- connector-module element 39
- connector-resource element 40
- connector-service element 41
- container-managed persistence 91
- context-root attribute 122
- custom-resource element 42

D

- das-config element 43
- databaseName property 68
- datasource-classname attribute 66
- datasourceName property 68
- debug-enabled attribute 64
- debug-options attribute 64
- default virtual server
 - for an http-listener element 57
 - for the entire server 116
- default-client-provider attribute 84
- default-jms-host attribute 71
- default-principal attribute 102

- default-principal-password attribute 102
- default-provider attribute 84
- default-realm attribute 102
- default-response-type attribute 58
- default-virtual-server attribute 57
- default-web-module attribute 117
- deploy-xml-validation attribute 44
- description element 45
- description property 68
- directory attribute 109
- directory property 29
- directory-deployed attribute 26, 40, 50, 62, 123
- disable-distributed-transaction-logging property 114
- disable-timeout-in-minutes attribute 26, 104
- dns-lookup-enabled attribute 58
- docroot attribute 117
- documentation
 - overview 11
- domain element 45
- domain.xml file
 - DTD file for 16
 - element hierarchy 18
 - elements in 21
 - location 15
- dynamic-reconfiguration-enabled attribute 35
- dynamic-reload-enabled attribute 44
- dynamic-reload-poll-interval-in-seconds attribute 44

E

- ejb-container element 46
- ejb-container-availability element 48
- ejb-module element 50
- ejb-timer-service element 51
- elements
 - descriptions of 21
 - hierarchy 18
 - referencing 18
- enableCookies property 107
- enableURLRewriting property 107
- env-classpath-ignored attribute 65

external-jndi-resource element [52](#)
 external-port attribute [57](#)

F

factory-class attribute [42](#), [52](#), [92](#)
 fail-all-connections attribute [37](#), [67](#)
 family attribute [57](#)
 file attribute [78](#)
 file property [29](#)
 file-caching-enabled attribute [55](#)
 file-transmission-enabled attribute [56](#)
 filter-class-name attribute [54](#)
 filter-config element [53](#)
 forced-response-type attribute [58](#)
 format attribute [22](#)

G

globally-enabled attribute [55](#)
 group-base-dn property [29](#)
 group-search-filter property [29](#)
 group-target property [29](#)

H

hash-init-size attribute [56](#)
 header-buffer-in-bytes attribute [97](#)
 health-checker element [54](#)
 heuristic-decision attribute [114](#)
 host attribute [70](#)
 hosts attribute [117](#)
 checking against subject pattern [116](#)
 HTTP listen socket [56](#)
 http-access-log element [54](#)
 http-file-cache element [55](#)
 http-listener element [56](#)

http-listeners attribute [117](#)
 http-protocol element [57](#)
 http-service element [58](#)
 http-session-store-pool-name attribute [121](#)
 https-routing attribute [75](#)

I

id attribute
 for http-listener element [56](#), [60](#)
 for virtual-server element [117](#)
 idLengthBytes property [107](#)
 idle-thread-timeout-in-seconds attribute [112](#)
 idle-timeout-in-seconds attribute [37](#), [66](#), [83](#)
 IIOP listen socket [60](#)
 iiop-listener element [60](#)
 iiop-service element [61](#)
 initial-thread-count attribute [97](#)
 init-timeout-in-seconds attribute [71](#)
 instance-name property [72](#)
 instance-name-suffix property [72](#)
 INSTANCE-SYNC-JVM-OPTIONS property [89](#)
 interval-in-seconds attribute [54](#)
 iponly attribute [55](#)
 is-connection-validation-required attribute [67](#)
 is-failure-fatal attribute [76](#)
 is-isolation-level-guaranteed attribute [67](#)

J

j2ee-application element [61](#)
 jaas-context property [29](#)
 jacc attribute [102](#)
 jacc-provider element [62](#)
 Java Database Connectivity *see* JDBC
 Java Naming and Directory Interface *see* JNDI
 Java Platform Debugger Architecture *see* JPDA
 Java Transaction Service *see* JTS
 Java Virtual Machine *see* JVM

[java-config element](#) 63
[javac-options attribute](#) 64
[java-home attribute](#) 64
[JavaMail](#) 79
[JDBC connection pool](#) 65
[JDBC resource](#) 68
[jdbc-connection-pool element](#) 65
[jdbc-resource element](#) 68
[jdbc-resource-jndi-name attribute](#) 92
[JMS](#) 69, 70
[jms-host element](#) 69
[jms-service element](#) 70
[jmx-connector element](#) 72
[JNDI](#) 52
[jndi-lookup-name attribute](#) 52
[jndi-name attribute](#)
 for custom-resource element 23, 42
 for external-jndi-resource element 52
 for jdbc-resource element 41, 69
 for mail-resource element 80
 for persistence-manager-factory-resource element 91
[JPDA debugging options](#) 64
[JTS](#) 113
[JVM](#) 63
 adding options to the server 73
[jvm-options element](#) 73

K

[keep-alive element](#) 74
[keep-alive-query-max-sleep-time property](#) 59
[keep-alive-query-mean-time property](#) 59
[keypoint-interval attribute](#) 114

L

[lb-config element](#) 74
[lb-configs element](#) 75
[lb-enabled attribute](#) 26, 104

[lifecycle-module element](#) 76
[listen socket](#) 56, 60
[listener-class-name attribute](#) 77
[listener-config element](#) 77
[load balancing](#) 122
[load-order attribute](#) 76
[locale attribute](#) 46
[location attribute](#) 26, 40, 50, 62, 122
[log-directory attribute](#) 55
[log-file attribute](#) 117
[log-filter attribute](#) 79
[logging settings in domain.xml](#) 78
[log-handler attribute](#) 79
[log-root attribute](#) 46
[log-rotation-limit-in-bytes attribute](#) 79
[log-rotation-timelimit-in-minutes attribute](#) 79
[log-service element](#) 78
[log-to-console attribute](#) 79

M

[mail-resource element](#) 79
[manager-properties element](#) 81
[max-age-in-seconds attribute](#) 55
[max-cache-size attribute](#) 47
[max-connections attribute](#) 74, 91
[max-files-count attribute](#) 56
[max-pending-count attribute](#) 36
[max-pool-size attribute](#) 37, 47, 66, 83
[max-redeliveries attribute](#) 51
[max-sessions attribute](#) 82
[max-thread-pool-size attribute](#) 112
[max-wait-time-in-millis attribute](#) 37, 66
[mdb-container element](#) 83
[medium-file-size-limit-in-bytes attribute](#) 55
[medium-file-space-in-bytes attribute](#) 56
[message-driven beans](#) 83
[message-fragment-size attribute](#) 91
[message-security-config element](#) 84
[minimum-delivery-interval-in-millis attribute](#) 51

- min-thread-pool-size attribute 112
- module-log-levels element 85
- module-monitoring-levels element 87
- monitoring-cache-enabled property 59
- monitoring-cache-refresh-in-millis property 59
- monitoring-enabled attribute 75
- monitoring-service element 88
- mq-scheme attribute 71
- mq-service attribute 71

N

- name attribute 25, 70, 73, 89
 - for auth-realm element 28, 29, 63
 - for connector-module element 26, 40
 - for ejb-module element 50
 - for j2ee-application element 62
 - for jdbc-connection-pool element 37, 66
 - for lifecycle-module element 76
 - for profiler element 93
 - for property element 94, 101, 111
 - for resource-adapter-config element 97
 - for server element 32, 35, 75, 103
 - for web-module element 122
- native library path, configuring 64
- native-library-path attribute 93
- native-library-path-prefix attribute 64
- native-library-path-suffix attribute 64
- networkProtocol property 68
- node-agent element 88
- node-agent-ref attribute 104
- node-agents element 90
- num-work-queues attribute 112
- for j2ee-application element 62
- for jdbc-resource element 69
- for web-module element 122
- oracle-xa-recovery-workaround property 114
- orb element 90

P

- password attribute 32
- Password property 39
- password property 68
- pending-txn-cleanup-interval property 115
- persistence-frequency attribute 120
- persistence-manager-factory-resource element 91
- persistence-scope attribute 121
- persistence-type attribute 120
- policy-configuration-factory-provider attribute 63
- policy-provider attribute 63
- pool-idle-timeout-in-seconds attribute 47
- pool-name attribute 41, 69
- pool-resize-quantity attribute 37, 47, 66, 83
- port attribute 57, 60, 70, 73
- port property 68
- principal element 92
- profiler element 92
- properties, about 93
- properties, system 110, 111
- property element 93
- protocol attribute 73
- provider-config element 95
- provider-id attribute 95
- provider-type attribute 95

O

- object-type attribute
 - for connector-module element 40
 - for connector-resource element 41
 - for ejb-module element 23, 43, 50, 53, 81, 92, 98

Q

- queue-size-in-bytes attribute 36

R

reap-interval-in-seconds attribute 82, 110
 receive-buffer-size-in-bytes attribute 36
 reconnect-attempts attribute 71
 reconnect-enabled attribute 71
 reconnect-interval-in-seconds attribute 71
 redelivery-interval-internal-in-millis attribute 51
 redirect-port attribute 57
 ref attribute 26, 33, 99, 104
 referencing elements 18
 reload-poll-interval-in-seconds attribute 75
 removal-timeout-in-seconds attribute 48
 request-policy element 96
 request-processing element 96
 request-timeout-in-seconds attribute 97
 res-adapter attribute 23
 resource adapters 39
 resource-adapter-config element 97
 resource-adapter-name attribute 37, 98
 resource-ref element 98
 resources element 99
 response-policy element 100
 response-timeout-in-seconds attribute 75
 res-type attribute 23, 42, 52, 66
 retry-timeout-in-seconds attribute 114
 rmic-options attribute 64
 roleName property 68
 rotation-enabled attribute 22
 rotation-interval-in-minutes attribute 22
 rotation-policy attribute 22
 rotation-suffix attribute 22

S

search-bind-dn property 30
 search-bind-password property 30
 search-filter property 29
 Secure Socket Layer *see* SSL
 security 101

security-enabled attribute 57, 60, 73
 security-map element 101
 security-service element 101
 send-buffer-size-in-bytes attribute 36
 server element 103
 server log file 46, 78
 server-classpath attribute 64
 server-name attribute 57
 serverName property 68
 server-ref element 104
 servers element 105
 session-config element 105
 session-file-name attribute 82
 session-id-generator-classname attribute 82
 session-manager element 105
 session-properties element 106
 session-store attribute 48
 session-timeout element 106
 setCacheControl property 118
 sfsb-ha-persistence-type attribute 49
 sfsb-persistence-type attribute 49
 sfsb-store-pool-name attribute 49
 shutdown-timeout-in-seconds attribute 42
 single sign-on 118
 small-file-size-limit-in-bytes attribute 56
 small-file-space-in-bytes attribute 56
 Solaris

- patches 12
- support 12

 ssl element 107
 SSL, configuring 107
 ssl2-ciphers attribute 108
 ssl2-enabled attribute 108
 SSL3 client authentication 108
 ssl3-enabled attribute 108
 ssl3-session-timeout property 59
 ssl3-tls-ciphers attribute 108
 ssl-cache-entries property 59
 ssl-client-auth-data-limit property 59
 ssl-client-auth-timeout property 59
 ssl-client-config element 108
 ssl-enabled attribute 58

- ssl-session-timeout property 59
- sso-enabled property 118
- sso-failover-enabled attribute 121
- sso-max-inactive-seconds property 118
- sso-reap-interval-seconds property 118
- stack-size property 59
- start-args attribute 71
- start-servers-in-startup attribute 89
- state attribute 117
- steady-pool-size attribute 37, 47, 66, 83
- store-pool-name attribute 31
- store-properties element 109
- store-protocol attribute 80
- store-protocol-class attribute 80
- subscribe-listener-with attribute 77
- Sun Java System Message Queue 72
- sun-domain_1_1.dtd file 16
- system-jmx-connector-name attribute 24, 89
- system-property element 110

T

- thread-count attribute 74, 97
- thread-increment attribute 97
- thread-pool element 112
- thread-pool-id attribute 112
- thread-pool-ids attribute 98
- thread-pools element 113
- timeout-in-seconds attribute 54, 74, 106, 113
- timer-datasource attribute 51
- tls-enabled attribute 108
- tls-rollback-enabled attribute 108
- transaction log file 78, 114
- transaction-isolation-level attribute 67
- transaction-service element 113
- transaction-support attribute 38
- transport-protocol attribute 80
- transport-protocol-class attribute 80
- tx-log-dir attribute 114
- type attribute 24, 71

U

- url attribute 54
- url property 68
- use-last-agent-optimization property 115
- user property 68
- user-group element 115
- user-name attribute 31
- UserName property 39
- use-system-logging attribute 79
- use-thread-pool-ids attribute 91

V

- validation-table-name attribute 67
- value attribute 94, 111
- version attribute 58
- victim-selection-policy attribute 48
- virtual servers, log file for 78
- virtual-server element 116
- virtual-servers attribute 26

W

- web module, default 117
- web.xml file session-timeout element 106
- web-container element 118
- web-container-availability element 119
- web-module element 122

X

- xaresource-txn-timeout property 115
- XML specification 13, 16
- xpowered-by attribute 57

