HITACHI
Inspire the Next

Hitachi Application Server V10 Command Reference Guide (For UNIX® Systems)

Reference

3021-3-419-10(E)

Notices

■ Relevant program products

Applicable OSs: Red Hat Enterprise Linux 5 Advanced Platform (AMD/Intel 64), Red Hat Enterprise Linux 5 (AMD/Intel 64), Red Hat Enterprise Linux Server 6 (64-bit x86_64)

P-9W43-5KA2 Hitachi Application Server 10-10

Note that OSs other than those listed above in "Applicable OSs" might become usable. For details, see the *Release Notes*.

■ Export restrictions

If you export this product, please check all restrictions (for example, Japan's Foreign Exchange and Foreign Trade Law, and USA export control laws and regulations), and carry out all required procedures.

If you require more information or clarification, please contact your Hitachi sales representative.

■ Trademarks

Adobe and Flash Player are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

AMD, AMD Opteron, and combinations thereof, are trademarks of Advanced Micro Devices, Inc.

Apache is either a registered trademark or a trademark of the Apache Software Foundation in the United States and/or other countries.

CORBA is a registered trademark of Object Management Group, Inc. in the United States.

Firefox is a registered trademark of the Mozilla Foundation.

IBM, AIX , POWER and WebSphere are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide.

IIOP is a trademark of Object Management Group, Inc. in the United States.

Intel is a trademark of Intel Corporation in the U.S. and/or other countries.

Kerberos is a name of network authentication protocol created by Massachusetts Institute of Technology.

Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

Microsoft, Excel, Hyper-V, Internet Explorer, SQL Server, Visual C++, Windows, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Mozilla is a trademark of the Mozilla Foundation in the U.S and other countries.

OMG, CORBA, IIOP, UML, Unified Modeling Language, MDA and Model Driven Architecture are either registered trademarks or trademarks of Object Management Group, Inc. in the United States and/or other countries.

Oracle and Java are registered trademarks of Oracle and/or its affiliates.

RC4 is either a registered trademark or a trademark of EMC Corporation in the United States and/or other countries.

Red Hat is a trademark or a registered trademark of Red Hat Inc. in the United States and other countries.

RSA and BSAFE are either registered trademarks or trademarks of EMC Corporation in the United States and/or other countries.

SOAP is an XML-based protocol for sending messages and making remote procedure calls in a distributed environment.

UNIX is a registered trademark of The Open Group in the United States and other countries.

VMware, VMware vSphere ESX and VMware vSphere ESXi are registered trademarks or trademarks of VMware, Inc. in the United States and/or other jurisdictions.

W3C is a trademark (registered in numerous countries) of the World Wide Web Consortium.

Other product and company names mentioned in this document may be the trademarks of their respective owners. Throughout this document Hitachi has attempted to distinguish trademarks from descriptive terms by writing the name with the capitalization used by the manufacturer, or by writing the name with initial capital letters. Hitachi cannot attest to the accuracy of this information. Use of a trademark in this document should not be regarded as affecting the validity of the trademark.

This product includes RSA BSAFE® Cryptographic software of EMC Corporation.

This product includes software developed by Ben Laurie for use in the Apache-SSL HTTP server project.

Portions of this software were developed at the National Center for Supercomputing Applications (NCSA) at the University of Illinois at Urbana-Champaign.

This product includes software developed by the University of California, Berkeley and its contributors.

This software contains code derived from the RSA Data Security Inc. MD5 Message-Digest Algorithm, including various modifications by Spyglass Inc., Carnegie Mellon University, and Bell Communications Research, Inc (Bellcore).

Regular expression support is provided by the PCRE library package, which is open source software, written by Philip Hazel, and copyright by the University of Cambridge, England. The original software is available from ftp://ftp.csx.cam.ac.uk/pub/software/programming/pcre/

This product includes software developed by Ralf S. Engelschall <rse@engelschall.com> for use in the mod_ssl project (http://www.modssl.org/).

This product includes software developed by IAIK of Graz University of Technology.

This product includes software developed by Daisuke Okajima and Kohsuke Kawaguchi (http://relaxngcc.sf.net/).

This product includes software developed by Andy Clark.

Java is a registered trademark of Oracle and/or its affiliates.

HITACHI
Inspire the Next

Hitachi, Ltd.

■ Restrictions

Information in this document is subject to change without notice and does not represent a commitment on the part of Hitachi. The software described in this manual is furnished according to a license agreement with Hitachi. The license

agreement contains all of the terms and conditions governing your use of the software and documentation, including all warranty rights, limitations of liability, and disclaimers of warranty.

Material contained in this document may describe Hitachi products not available or features not available in your country.

No part of this material may be reproduced in any form or by any means without permission in writing from the publisher.

■ Issued

Mar. 2015: 3021-3-419-10(E)

■ Copyright

Preface

■ Purpose of this manual

This manual describes the commands used in operations and the configuration of the system infrastructure of Hitachi Application Server. By using this manual, users are able to configure and design in detail the system infrastructure of Hitachi Application Server.

■ Intended readers

The intended readers of this manual are as follows:

- For systems engineer

The prerequisite is as follows:

- Information related to the operations of Windows or UNIX.
- Information related to the Application Server development.
- Information related to the Java EE Standard specifications.
- Information related to the peripheral environment (such as database, network, and job management, and so on) used in the system development.

■ Related manuals

The following table shows the titles of related manuals, and related manuals used in this manual.

Application Server related manuals

Abbreviation	Formal name	Reference number
User's Guide	*Hitachi Application Server V10 User's Guide* (For UNIX® Systems)	3021-3-415(E)
GUI Reference	*Hitachi Application Server V10 GUI Reference Guide* (For UNIX® Systems)	3021-3-417(E)
Command Reference	*Hitachi Application Server V10 Command Reference Guide* (For UNIX® Systems)	3021-3-419(E)
Definition Reference	*Hitachi Application Server V10 Definition Reference Guide* (For UNIX® Systems)	3021-3-421(E)
Messages	*Hitachi Application Server V10 Messages*	3021-3-422(E)
API Reference	*Hitachi Application Server V10 API Reference Guide*	3021-3-423(E)

■ Abbreviations for products and functions

This manual uses the following abbreviations for product names and function name:

Abbreviations	Product name and function name
Application Server	Hitachi Application Server

Abbreviations		Product name and function name
Application Server - Base		Hitachi Application Server - Base
Application Server - Optional License for Java		Hitachi Application Server - Optional License for Java
Application Server for Developers		Hitachi Application Server for Developers
APV		IBM Advanced POWER Virtualization
DAS		Domain Administration Server
domain administration server		
Developer's Kit for Java		Hitachi Developer's Kit for Java
Excel		Microsoft® Excel
Firefox		Firefox®
HiRDB	HiRDB Version 9	HiRDB Server Version 9
	HiRDB/Single Server	HiRDB/Single Server Version 9
HWS		Hitachi Web Server
Web Server		
Internet Explorer		Windows® Internet Explorer®
Java EE Server		Hitachi Java EE Server
JP1/AJS3		Job Management Partner 1 Version 10 Job Management Partner 1/Automatic Job Management System 3 - Agent
		Job Management Partner 1 Version 10 Job Management Partner 1/Automatic Job Management System 3 - Manager
		Job Management Partner 1 Version 10 Job Management Partner 1/Automatic Job Management System 3 - View
JP1/IM		Job Management Partner 1 Version 10 Job Management Partner 1/Integrated Management - Manager
		Job Management Partner 1 Version 10 Job Management Partner 1/Integrated Management - View
Microsoft IIS	Microsoft IIS 7.5	Microsoft® Internet Information Services 7.5
	Microsoft IIS 8.0	Microsoft® Internet Information Services 8.0
	Microsoft IIS 8.5	Microsoft® Internet Information Services 8.5
Microsoft Visual C++		Microsoft® Visual C++®
Oracle	Oracle 11g	Oracle Database 11g
		Oracle Database 11g R2
	Oracle 12c	Oracle Database 12c
performance tracer		Performance Tracer
SQL Server		Microsoft® SQL Server
UNIX	AIX	AIX V6.1
		AIX V7.1

Abbreviations			Product name and function name
	Linux	Linux (x86/AMD64 & Intel EM64T)	Red Hat Enterprise Linux® 5 Advanced Platform (AMD/Intel 64)
			Red Hat Enterprise Linux® 5 (AMD/Intel 64)
			Red Hat Enterprise Linux® Server 6 (64-bit x86_64)
Virtage			Hitachi Virtage
VMware ESX			VMware vSphere ESX
VMware vSphere ESXi			VMware vSphere ESXi
Windows	Windows Server 2008 R2		Microsoft® Windows Server® 2008 R2 Standard
			Microsoft® Windows Server® 2008 R2 Enterprise
			Microsoft® Windows Server® 2008 R2 Datacenter
	Windows Server 2012		Microsoft® Windows Server® 2012 Standard
			Microsoft® Windows Server® 2012 Datacenter
	Windows Server 2012 R2		Microsoft® Windows Server® 2012 R2 Standard
			Microsoft® Windows Server® 2012 R2 Datacenter
	Windows 7	Windows 7 x86	Microsoft® Windows® 7 Professional (32 bit)
			Microsoft® Windows® 7 Enterprise (32 bit)
			Microsoft® Windows® 7 Ultimate (32 bit)
		Windows 7 x64	Microsoft® Windows® 7 Professional (64 bit)
			Microsoft® Windows® 7 Enterprise (64 bit)
			Microsoft® Windows® 7 Ultimate (64 bit)
	Windows 8	Windows 8 x86	Windows® 8 Pro (32 bit)
			Windows® 8 Enterprise (32 bit)
		Windows 8 x64	Windows® 8 Pro (64 bit)
			Windows® 8 Enterprise (64 bit)
		Windows 8.1 x86	Windows® 8.1 Pro (32 bit)
			Windows® 8.1 Enterprise (32 bit)
		Windows 8.1 x64	Windows® 8.1 Pro (64 bit)
			Windows® 8.1 Enterprise (64 bit)
Windows Server Failover Cluster			Windows Server® Failover Cluster
Class-wise statistics			Hitachi Class-wise statistics

For each version of Linux, the following abbreviations might be used.

Abbreviation		OS
Red Hat Enterprise Linux 5	Red Hat Enterprise Linux 5 Advanced Platform (AMD/Intel 64)	Red Hat Enterprise Linux® 5 Advanced Platform (AMD/Intel 64)

Abbreviation		OS
	Red Hat Enterprise Linux 5 (AMD/Intel 64)	Red Hat Enterprise Linux® 5 (AMD/Intel 64)
Red Hat Enterprise Linux Server 6	Red Hat Enterprise Linux Server 6 (64-bit x86_64)	Red Hat Enterprise Linux® Server 6 (64-bit x86_64)

■ Conventions: Acronyms

This manual also uses the following acronyms:

Acronym	Full name or meaning
ACC	Application Client Cotainer
ACL	Access Control List
AES	Advanced Encryption Standard
API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
BLOB	Binary Large Object
CA	Certificate Authority
CDI	Contexts and Dependency Injection
CGI	Common Gateway Interface
CMP	Container-Managed Persistence
CMT	Container-Managed Transaction
CopyGC	Copy Garbage Collection
CORBA	Common Object Request Broker Architecture
	CORBA®
CPU	Central Processing Unit
CRL	Certificate Revocation List
CSR	Certificate Signing Request
CSV	Comma Separated Value
CVS	Concurrent Versions System
DBMS	Database Management System
DCOM	Distributed Component Object Model
DD	Deployment Descriptor
DDE	Dynamic Data Exchange
DER	Distinguished Encoding Rules
DES	Data Encryption Standard
DI	Dependency Injection

Acronym	Full name or meaning
DLL	Dynamic Link Library
DMZ	Demilitarized Zone
DN	Distinguished Name
DNS	Domain Name System
DoS	Denial of Service
DSO	Dynamic Shared Object
DTD	Document Type Definition
EAR	Enterprise Archive
ear	
EIS	Enterprise Information System
EJB	Enterprise JavaBeans™
EJB QL	EJB™ Query Language
EL	Expression Language
EUC	Extended UNIX Code
FQDN	Fully Qualified Domain Name
FullGC	Full Garbage Collection
G1GC	Garbage First Garbage Collection
GC	Garbage Collection
GMS	Group Management Service
GMT	Greenwich Mean Time
GUI	Graphical User Interface
HA	High Availability
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transfer Protocol
HTTPS	Hyper Text Transfer Protocol Security
I/O	Input/Output
ID	Identifier
IDE	Integrated Development Environment
IEC	International Electrotechnical Commission
IIOP	IIOP™
	Internet Inter-Orb Protocol
IIS	Internet Information Services
IMAP	Internet Message Access Protocol

Acronym	Full name or meaning
IP	Internet Protocol
IPv6	Internet Protocol Version 6
ISO	International Organization for Standardization
J2EE	J2EE™
	Java™ 2 Platform, Enterprise Edition
JAAS	Java™ Authentication and Authorization Service
JACC	Java™ Authorization Service Provider Contract for Containers
JAF	JavaBeans™ Activation Framework Specification
JAR	Java™ Archive
jar	
JASPIC	Java™ Authentication Service Provider Interface for Containers
Java	Java™
Java EE	Java™ Platform, Enterprise Edition
Java EE RI	Java EE Reference Implementation
Java HotSpot Client VM	Java HotSpot™ Client Virtual Machine
Java Platform Debugger Architecture	Java™ Platform Debugger Architecture
JPDA	
Java SE	Java™ Platform, Standard Edition
Java VM	Java™ Virtual Machine
JVM	
JavaMail	JavaMail™
JAX-RPC	Java™ API for XML-based RPC
JAX-RS	Java™ API for RESTful Web Services
JAX-WS	Java™ API for XML-based Web Services
JAXB	Java™ Architecture for XML Binding
JAXP	Java™ API for XML Processing
JAXR	Java™ API for XML Registries
JCA	J2EE™ Connector Architecture
JDBC	Java™ Database Connectivity
	JDBC™
JDK	Java™ Development Kit
	JDK™
JIS	Japanese Industrial Standards

Acronym	Full name or meaning
JMS	Java™ Message Service
JMX	Java™ Management Extensions
JNDI	Java Naming and Directory Interface™
JNI	Java™ Native Interface
JPA	Java™ Persistence API
JSF	JavaServer™ Faces
	JavaServer™ Faces Reference Implementation (RI) Version: 1.1_01 FCS
JSON-P	Java™ API for JSON Processing
JSP	JavaServer Pages™
	JSP™
JST	Japan Standard Time
JSTL	JavaServer Pages™ Standard Tag Library
JTA	Java™ Transaction API
JVMPI	Java™ Virtual Machine Profiler Interface
JVMTI	Java™ Virtual Machine Tool Interface
KVM	Kernel-based Virtual Machine
LAN	Local Area Network
LDAP	Lightweight Directory Access Protocol
MAC	Message Authentication Code
MIME	Multipurpose Internet Mail Extensions
OASIS	Organization for the Advancement of Structured Information Standards
OMG	Object Management Group
ORB	Object Request Broker
OS	Operating System
OTS	Object Transaction Service
QNAME	Qualified Name
REST	Representational State Transfer
RMI	Remote Method Invocation
RPC	Remote Procedure Call
RSA	Rivest, Shamir and Adleman
SAAJ	SOAP with Attachments API for Java™
SAX	Simple API for XML
SEI	Service Endpoint Interface

Acronym	Full name or meaning
Servlet	Java™ Servlet
SHA	Secure Hash Algorithm
SMAP	Source Map
SMTP	Simple Mail Transfer Protocol
SOAP	Simple Object Access Protocol
SSH	Secure Shell
ssh	
SSL	Secure Sockets Layer
StAX	Streaming API for XML
TCP	Transmission Control Protocol
TLD	Tag Library Descriptor
TLS	Transport Layer Security
UCS	Universal multi-octet coded Character Set
UDP	User Datagram Protocol
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
URN	Uniform Resource Name
UTC	Coordinated Universal Time
UTF	UCS Transformation Format
UTF-8	8-bit UCS Transformation Format
VM	Virtual Machine
VTL	Velocity Template Language
W3C	World Wide Web Consortium
WAR	Web Archive
war	
WBEM	Web-Based Enterprise Management
WSDL	Web Services Description Language
XML	Extensible Markup Language

■ Conventions: KB, MB, GB, TB, and PB

This manual uses the following conventions:

- 1 KB (kilobyte) is 1,024 bytes.
- 1 MB (megabyte) is $1{,}024^2$ bytes.

- 1 GB (gigabyte) is $1{,}024^3$ bytes.
- 1 TB (terabyte) is $1{,}024^4$ bytes.
- 1 PB (petabyte) is $1{,}024^5$ bytes.

Contents

1 About command reference

Command reference information explains the syntax and functionality of commands used for configuring and operating Application Server system infrastructure.

1.1 How to input a command

This section explains how to input commands.

Synopsis

The following show the format for inputting a command:

```
command_name [option...]
```

Each item is explained below. Note that `$` indicates the command prompt, and `cmd` indicates a command name.

command_name

Specify the file name of the command to be run.

When you run a command that specifies a path that contains a blank space, enclose the entire path in double quotation marks (""). For example:

- Incorrect example: `$ /opt/program path/bin/cmd`
- Correct example: `$ "/opt/program path/bin/cmd"`

(Note that the paths in the examples are different from the paths used in the actual product.)

Input format of options

Options always start with a hyphen (-). Option input formats include a format in which no option argument is specified and a format in which one option argument is specified.

- The following format is used to specify no option argument:

  ```
  $ cmd -option_flag
  ```

- The following format is used to specify one option argument:

  ```
  $ cmd -option_flag space_or_tab option_argument
  ```

Legend:

option_flag: This is one half-width alphanumeric character. Characters are case sensitive (uppercase and lowercase characters are distinguished).

option_argument: This is an argument of the option flag.

Rules for specifying options

- You cannot specify all the option flags together with a single hyphen (-). For example:

  Incorrect example: `$ cmd -abc`

  Correct example: `$ cmd -a -b -c`

- Option arguments of an option flag that require arguments cannot be omitted.
- A blank or tab space must be inserted between an option flag and its argument. For example:

  Incorrect example: `$ cmd -afile`

  Correct example: `$ cmd -a file`

- The same option flag cannot be specified multiple times. For example:

  Incorrect example: `$ cmd -a 1 -a 2`

- If an option argument includes a blank space, all the option arguments must be enclosed in double quotation marks (""). For example:

  Incorrect example: `$ cmd -a file 1`

  Correct example: `$ cmd -a "file 1"`

1.2 How to read command reference information

This section provides important information required for reading the Command Reference Guide.

Description format

This section describes the format used to explain commands. The explanation of a command contains only required items from among the following items. Additionally, there are cases in which a command explanation contains information that is specific to the command and which is not described below.

Synopsis
: Indicates the syntax for inputting the command.

Storage location
: Indicates the path where the commands are stored.

Function
: Describes the functionality of the command.

Execution permission
: Indicates the execution permissions required to run the command.

Precondition
: Indicates the requirements to run the command.

Arguments
: Describes the command arguments and their options.

  Specification of a value might be required, depending on the option.

  For commands used in the Java EE server, "Default Value" indicates the value for when the entire option is omitted, and not for when only the option value is omitted.

Output format
: Indicates the format of output from the command.

Examples
: Provides an example of command input and output.

Exit Status
: Describes the return values of the command.

Notes
: Describes important points to note when executing the command.

Symbols used

The following table describes the various symbols used in command syntax explanations:

Symbols	Meaning
`|`	Indicates that items are separate, when multiple items are arranged side-by-side. This symbol means OR. Example: `A|B` This example indicates that you need to specify either A or B.
`{ }`	Indicates that you need to select one of the items from among the multiple items enclosed within these symbols. If the items are listed side-by-side and separated by the symbol `|`, you can select one of the listed items. Example: `{A|B|C}` This example indicates that you need to specify A or B or C.

Symbols	Meaning
[ ]	Indicates that the items enclosed in the symbols can be omitted. If multiple items enclosed within these symbols are written horizontally, you can ignore all the items or select one of them (the same as with the symbols { }). Example 1: [A] This example indicates that you do not need to specify anything, or you can specify A. Example 2: [B|C] This example indicates that you do not need to specify anything, or you can specify B or C.
...	Indicates that the item specified before "..." can be specified multiple times. Example: [*property*]... This example indicates that you can specify multiple properties repeatedly.
item	Items that are in italics indicate corresponding files or elements that are specified in the command or the corresponding elements that might be displayed. Example 1: *property* This example indicates that a property is to be written here, or a property is displayed here. Example 2: *file_name* This example indicates that a file name here is specified here.

Syntax elements used

The following table defines the syntax elements used in command explanations:

Syntax element	Definition
Alphabetical characters	A to Z a to z
Lowercase letters	a to z
Uppercase letters	A to Z
Numbers	0 to 9
Alphanumeric characters	A to Z a to z 0 to 9
Symbols	! " # $ % & ' ( ) + , _ . / : ; < = > @ [ ] ^ - { } ~ Tab space

Note:

In all the syntax elements, use half-width characters.

2 Commands used in the Java EE server

This section describes the syntax and functionality of the commands used in the Java EE server.

2.1 List of commands to be used in the Java EE server

The following table shows a list of commands to be used in the Java EE server.

Commands used in the Java EE server

Command Name	Classification	Summary
appclient	Launches the Application Client Container and invokes the client application	Launches the Application Client Container and invokes the client application typically packaged in the application JAR file.
asadmin	Performs administrative tasks	Performs administrative tasks for Java EE Server.
imqcmd list dst	Lists physical and temporary destinations	Lists all physical and temporary destinations of the specified type.
jspc	Allows you to precompile JSP files	Allows you to precompile JSP files at the command line.
schemagen	Contains the schema generator	Performs the schema generator process for either the Java source files or class files.
wscompile	Generates stubs, ties, serializers, and WSDL files used in JAX-RPC clients and services	Generates stubs, ties, serializers, and WSDL files used in JAX-RPC clients and services.
wsdeploy	Generates a deployable WAR file for a service	Generates a deployable WAR file for a service.
wsgen	Generates the JAX-WS portable artifacts used in the JAX-WS web services	Generates the JAX-WS portable artifacts used in the JAX-WS Web services.
wsimport	Generates JAX-WS portable artifacts	Generates JAX-WS portable artifacts.
xjc	Xml to java compiler	Produces a set of packages containing Java source files and also `jaxb.properties` files, depending on the binding options you used for compilation.

Important note

- Specify the value which can be specified by each command as the argument of a command. Operation is not guaranteed when other values are specified.
- Even if a command is executed normally, errors may have occurred. Check the message log and stack trace log.

2.2 Commands used in the JavaEE server

This section describes the syntax and functionality of the commands used in the JavaEE server.

2.2.1 appclient

Launches the Application Client Container and invokes the client application typically packaged in the application JAR file.

Synopsis

```
appclient
    [client_application_classfile | -client client_application_jar]
    [-mainclass main_class_name | -name display_name]
    [-xml sun-acc.xml file] [-textauth]
    [-targetserver host[:port][,host[:port]...]] [-user username]
    [-passwordfile password_file] [application-options]

appclient [jvm-options]
    [-mainclass main_class_name | -name display_name]
    [-xml client_config_xml_file] [-textauth]
    [-targetserver host[:port][,host[:port]...]] [-user username]
    [-passwordfile password_file]
    class-selector [application-options]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `appclient` launches the Application Client Container and invokes the client application that is typically packaged in an application JAR file. You can retrieve the client JAR file using the `asadmin get-client-stubs` command. The Application Client Container is a set of Java classes, libraries, and other files that are required to execute a first-tier application client program on Java VM. The Application Client Container communicates with the server using RMI-IIOP. The client JAR file that is retrieved after deploying an application should be passed with the `-client` or `-jar` option when running the appclient utility. The client JAR file name is of the form *app-name*Client.jar. For multiple application clients in an EAR file, you must use the `-mainclass` or `-name` option to specify which client must be invoked. If the application client is a stand-alone module or the only client in an EAR file, the Application Client Container can find the client without using the `-mainclass` or `-name` option. If you provide a `-mainclass` or `-name` value that does not match the value in the client, the Application Client Container still launches the client but issues a warning that the selection did not match the value in the client. The warning also displays the actual main class and name values of the client.

You must give the absolute path locations. Else, the relative paths are used. The relative path is relative to the directory where the command is being executed.

Execution permission

Superuser

Environment variable

- To use appclient to connect to a server in which mutual authentication is enabled, specify the following values in the `VMARGS` environment variable:

  `-Djavax.net.ssl.keyStore=<keystore-file-path>`

  `-Djavax.net.ssl.trustStore=<truststore-file-path>`

- If you want to store the PRF trace, you need to set the following environment variables:

  `PRFSPOOL`

  Specify the environment variable `PRFSPOOL` set by the domain administration server.

- The environment variable `PRFSPOOL` is set in the following format:

  `Java EE Server-log-output-destination-directory-for-the-node-on-which-Java EE Server-is-built``/nodes/`*`node-name/PRF-name`*

- For `client_application_classfile` option, CLASSPATH is required if the class is dependent on other user classes and if the classpath JVM option is not used.
- For `-xml` option, client configuration XML file (`sun-acc.xml`) is required.

Files

- *app-name*Client.jar is used to invoke the client application.
- `sun-acc.xml` is required if `-xml` is used.

Arguments

`client_application_classfile`

Indicates the file system pathname of the client application .class file. Providing this value is optional. To specify a relative path, describe a path that is relative to the current directory. This class file must contain the `main()` method to be invoked by the Application Client Container.

If you specify the *client_application_classfile* option and the specified class is dependent on other user classes, you must also set the classpath.

Type: String

The following values can be specified:

- *Path of application .class file*

Default value: N/A

`-client` *`client_application_jar`*

Specifies the name and location for the client JAR file. Providing this value is optional.

Type: String

The following values can be specified:

- *Name of the client jar*

Default value: N/A

`-mainclass` *`main_class_name`*

Indicates the full class name of the main client application, as specified in the `Main-Class` entry of the `MANIFEST.MF` file. Providing this value is optional. Used for a multiple client applications.

Type: String

The following values can be specified:

- *Class name*

Default value:

Class specified in the client jar when multiple client applications are present.

`-name` *`display_name`*

Indicates the display name for the client application. Providing this value is optional. Used for multiple client applications.

Type: String

The following values can be specified:

- *Display name*

Default value:

Value set to display-name attribute of application-client.xml file in client jar.

`-xml` *`sun-acc.xml file`*

Indicates the name and location of the client configuration XML file. Providing this value is optional, if you are using the default domain, instance, and name (`sun-acc.xml`). Else, it is required.

Type: String

The following values can be specified:

- *Name of the configuration file*

Default value:

`sun-acc.xml` file in the *`domain-dir`*`/config` directory.

`-textauth`

Specifies the text format authentication when authentication is needed. Providing this value is optional.

`-targetserver` *`host`*`:`*`port`*

Indicates a comma-separated list of one or more server specifications for ORB endpoints. Providing this value is optional. Each server specification must include at least the host. Optionally, a server specification can include the port as well.

If the port is omitted from a server specification, the default value, 3700, is used for that host.

Type: String

The following values can be specified:

- *One or more comma separated server specifications*

Default value: N/A

`-user` *`username`*

Indicates the application user who is authorized to have access to particular guarded components in the application, such as EJB components. Providing this value is optional.

Type: String

The following values can be specified:

- *User name*

Default value: N/A

`-passwordfile` *`password_file`*

Specifies the name, including the full path, of a file that contains the password for application clients in the format, PASSWORD=*appclient-password*.

Providing this value is optional.

For security reasons, a password that is specified as an environment variable is not read by the `appclient` utility.

Type: String

The following values can be specified:

- *Path of the password file*

Default value: N/A

jvm-options

Allows you to set JVM options for the client application. Providing this value is optional. These can be any valid java command options except `-client` or `-jar`. JVM options can be intermixed with other `appclient` command options as long as both types of options appear before *class-selector*.

class-selector

Indicates the client application class to be specified using one of the following class selectors. Providing this value is mandatory.

Type: String

The following values can be specified:

- `-jar` *jar-file*

  the name and location of the client JAR file. The application client JAR file is specified and created during deployment by the asadmin `deploy` command. If specified, the `-classpath` setting is ignored in deference to the `Class-Path` setting in the client JAR file's manifest.

- *class-name*

  the fully qualified name of the application client's main class. The Application Client Container invokes the main method of this class to start the client.

Default value: N/A

application-options

Enables you to set client application arguments. Providing this value is optional.

Type: String

The following values can be specified:

- *Client application arguments*

Default value: N/A

Examples

Where: `sun-acc.xml` is the name of the client configuration XML file, `myclientapp.jar` is the client application .jar file, and `scott` and `sample` are arguments to pass to the application. If `sun-acc.xml` and `myclientapp.jar` are not in the current directory, you must give the absolute path locations; else the relative paths are used. The relative path is relative to the directory where the command is being executed.

```
appclient -xml sun-acc.xml -jar myclientapp.jar scott sample
```

2.2.2 asadmin

Performs administrative tasks.

Synopsis

```
asadmin [--host host] [--port port] [--user admin-user]
    [--passwordfile filename] [--terse={true|false}]
    [--secure={false|true}] [--echo={true|false}]
```

```
    [--interactive={true|false}] [--detach={true|false}] [--help]
    [subcommand [options] [operands]]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The asadmin utility command performs administrative tasks for Java EE Server.

The asadmin utility has the following options:

- asadmin utility options: These options control the behavior of the asadmin utility, not the subcommand. The asadmin utility options may precede or follow the subcommand, but asadmin utility options after the subcommands are deprecated. All asadmin utility options must either precede or follow the subcommand. If asadmin utility options are specified both before and after the subcommand, an error occurs.
- Subcommand options: These options control the behavior of the subcommand, not the asadmin utility. Subcommand options must follow the subcommand.

The asadmin utility options and some subcommand options have a long form and a short form.

- The long form of an option has two dashes (--) followed by an option word.
- The short form of an option has a single dash (-) followed by a single character.

Most options require argument values, except Boolean options, which toggle to enable or disable a feature.

Operands specify the items on which the subcommand is to act. Operands must follow the argument values of subcommand options and are set off by a space, tab, or double dashes (--). The asadmin utility treats anything that follows the subcommand options and their values as an operand.

asadmin subcommands have timeouts which target communication between servers or the command processing time. As common timeouts for all the subcommands, there are read timeout and request timeout. For more details, refer to "AS_ADMIN_READTIMEOUT" in the asenv file, "configs.config.*name*.network-config.protocols.protocol.admin-listener.http.request-timeout-seconds" and "configs.config.*name*.network-config.protocols.protocol.sec-admin-listener.http.request-timeout-seconds" in "List of parameters used with the set and get subcommands".

The backslash (\) escape character is also required before a single quotes (') character or a double quotes (") character in an option that uses the colon as a delimiter.

The asadmin utility uses the colon character (:) as a delimiter for some options. The backslash (\) escape character is required if the colon is part of a property or an option of Java VM. Instead of using the backslash (\) escape character, you can use the double quotes (") character or single quotes (') character. The effects of the different types of quotes characters on the backslash (\) character are as follows:

- Between double quotes (") characters, the backslash (\) character is a special character.
- Between single quotes (') characters, the backslash (\) character is not a special character.

Execution permission

- To run the subcommand and have access to the installation directory and domain directory, the user must be logged in to the machine that hosts the domain.
- Superuser

Environment variable

The listed environment variables can be used to modify the default values of `asadmin` utility options.

- `AS_ADMIN_TERSE`
- `AS_ADMIN_ECHO`
- `AS_ADMIN_INTERACTIVE`
- `AS_ADMIN_HOST`
- `AS_ADMIN_PORT`
- `AS_ADMIN_SECURE`
- `AS_ADMIN_USER`
- `AS_ADMIN_PASSWORDFILE`
- `AS_ADMIN_HELP`
- `AS_ADMIN_READTIMEOUT`

Files

`--passwordfile` option requires a file containing password entries.

Arguments

`--help | -?`

Displays the help text for the `asadmin` utility.

`--host` *host* | `-H` *host*

Indicates the machine name where the DAS is running.

Type: String

The following values can be specified:

- *Machine name*

Default value: `localhost`

`--port` *port* | `-p` *port*

Indicates the HTTP port or HTTPS port for administration.

Type: String

The following values can be specified:

- *Machine name*

Default value: `4848`

`--user` *admin-user* | `-u` *admin-user*

Indicates the user name of the authorized administrative user of the DAS. If you have authenticated to a domain using the `login` subcommand, you need not specify the `--user` option for subsequent operations on the domain.

Type: String

The following values can be specified:

- *Name of the user*

Default value: N/A

`--passwordfile` *`filename`* `|` `-W` *`filename`*

Specifies the name, including the full path of a file that contains password entries in a specific format. These password entries are stored in clear text in the password file. To provide additional security, the `create-password-alias` subcommand can be used to create aliases for passwords that are used by remote subcommands. The password for which the alias is created is stored in an encrypted form.

If an alias exists for a password, the alias is specified in the entry for the password as follows:

`AS_ADMIN_password-name=${ALIAS=password-alias-name}`

Type: String

The following values can be specified:

- *Path and name of the password file*

Default value: N/A

`--terse={true|false}` | `-t={true|false}`

Displays the output data in a concise format, optimized for use in scripts instead of, for reading by humans, if set to `true`. Typically, descriptive text and detailed status messages are also omitted from the output data.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--secure={true|false}` | `-s={true|false}`

Indicates whether to use SSL/TLS to communicate with the DAS, if set to `true`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--echo={true|false}` | `-e={true|false}`

Echoes the command-line statement on the standard output, if set to `true`. The default value is `false`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--interactive={true|false}` | `-I={true|false}`

Specifies whether only the required options are prompted.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value:

- `true`: If the `asadmin` utility is run from a console window.
- `false`: If the `asadmin` utility is run from within a script.

`--detach={true|false}`

Detaches and executes the specified `asadmin` subcommand in the background in detach mode, if set to `true`. The `--detach` option is useful for long-running subcommands and enables you to execute several independent subcommands from one console or script. The `--detach` option is specified before the subcommand.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

Examples

The following example shows all the applications deployed on the local host.

```
asadmin list-applications
```

Related topics

- 2.11.11 List of parameters used with the set and get subcommands

2.2.3 imqcmd list dst

Lists the physical and temporary destinations.

Synopsis

```
imqcmd list dst -b hostName:portNumber
    [-t destType][-tmp]
```

Storage location

Application Server installation directory/javaee/mq/bin

Function

The `imqcmd list dst` subcommand lists all the physical and temporary destinations of the specified type.

Arguments

`-b` *hostName*:*portNumber*

Specifies the host name and the port number of the broker.

You can specify a literal IPv4 or IPv6 address as the host name. If you use a literal IPv6 address, its format must conform to the RFC2732 standards.

Type: String

The following values can be specified:

- *The host name separated by port number of the broker.*

Default value: `localhost:7676`

`-t` *`destType`*

Specifies the type of the physical destination.

Type: String

The following values can be specified:

- `t`
  Specifies the physical destination of the topic.
- `q`
  Specifies the physical destination of the queue.

Default value: N/A

`-tmp`

Lists the temporary destinations.

Examples

The following command lists all the physical destinations available on the broker, running on the host `myHost`, at the port number `4545`.

```
imqcmd list dst -b myHost:4545
```

Exit Status

Exit Status	Explanation
`0`	command executed successfully.
`1`	error in executing the command.

2.2.4 jspc

Allows you to precompile JSP files at the command line.

Synopsis

```
jspc OPTION [--] jsp_files

    OPTION:
    [-webapp dir]
    [-help]
    [-v={false|true}]
    [-d dir]
    [-l={false|true}]
    [-s={false|true}]
    [-p name]
```

```
    [-c name]
    [-mapped={false|true}]
    [-die[#]]
    [-uribase dir]
    [-uriroot dir]
    [-compile={false|true}]
    [-genclass={false|true}]
    [-webinc file]
    [-webxml file]
    [-classpath path]
    [-xpoweredBy={false|true}]
    [-trimSpaces={false|true}]
    [-smap={false|true}]
    [-dumpsmap={false|true}]
    [-validate={false|true}]
    [-compilerSourceVM jdk_release]
    [-compilerTargetVM jdk_release]
    [-ignoreJspFragmentErrors={false|true}]
    [-disablePooling={false|true}]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The jspc command line tool allows you to precompile JSP files at the command line.

JSP files are automatically compiled at runtime.

Execution permission

Superuser

Files

This tool takes .jsp files as input and generates servlet files.

Arguments

jsp files

Specifies a file to be parsed as a JSP page.

Type: String

The following values can be specified:

- *Name of the file*

Default value: N/A

-webapp *dir*

Specifies a directory containing a web-app, whose JSP pages will be processed recursively.

Type: String

The following values can be specified:

- *Directory path containing web-app*

Default value: N/A

`-help`

Prints this help message.

`-v={false|true}`

Displays detailed information.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-d` *`dir`*

Specifies the output Directory (default `-Djava.io.tmpdir`).

Type: String

The following values can be specified:

- *Directory to place the output files*

Default value: `-Djava.io.tmpdir`

`-l={false|true}`

Outputs the name of the JSP page upon failure.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-s={false|true}`

Outputs the name of the JSP page upon success.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-p` *`name`*

Specifies the name of the target package (The default value is `org.apache.jsp`).

Type: String

The following values can be specified:

- *Name of the package*

Default value: `org.apache.jsp`

`-c` *`name`*

Specifies the name of the target class name (only applies to first JSP page).

Type: String

The following values can be specified:

- *Name of the target class name*

Default value: N/A

`-mapped={false|true}`

Generates separate `write()` calls for each HTML line in the JSP.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-die[#]`

Generates an error return code (`#`) on fatal errors (default 1).

Default value: `1`

`-uribase` *dir*

Indicates that the uri directory compilations should be relative to (default `/`).

Type: String

The following values can be specified:

- *Directory name*

Default value: `/`

`-uriroot` *dir*

Specifies the directory containing a web-app, whose JSP pages will be processed recursively.

Type: String

The following values can be specified:

- *Directory name*

Default value: N/A

`-compile={false|true}`

Compiles generated servlets.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-genclass={false|true}`

Compiles generated servlets.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-webinc` *file*

Creates partial servlet mappings in the file.

Type: String

The following values can be specified:

- *Filename with path*

Default value: N/A

`-webxml` *`file`*

Creates a complete `web.xml` in the file.

Type: String

The following values can be specified:

- *Web-app name*

Default value: N/A

`-classpath` *`path`*

Appends the path to `java.class.path` system property.

Type: String

The following values can be specified:

- *Classpath*

Default value: N/A

`-xpoweredBy={false|true}`

Adds an `X-Powered-By` response header.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-trimSpaces={false|true}`

Trims the spaces in the template text between actions and directives.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-smap={false|true}`

Generates SMAP information for JSR45 debugging.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`-dumpsmap={false|true}`

Dumps the SMAP information for JSR45 debugging into a file.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-validate={false|true}`

Validates the `.tld` and `web.xml` files against their schemas and dtds.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-compilerSourceVM` *`jdk_release`*

Provides source compatibility with the specified JDK release.

Type: float

The following values can be specified:

- *JDK releases*

Default value: N/A

`-compilerTargetVM` *`jdk_release`*

Generates the class files for the specified VM version.

Type: float

The following values can be specified:

- *JDK releases*

Default value: N/A

`-ignoreJspFragmentErrors={false|true}`

Ignores the compilation errors of JSP fragments.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-disablePooling={false|true}`

Disables the custom tag pooling.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.2.5 schemagen

Contains the schema generator that can process either Java source files or class files.

Synopsis

```
schemagen [OPTION]... <java files>

    OPTION:
    [-d path]
    [-cp path]
    [-classpath path]
    [-episode file]
    [-version]
    [-help]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The schemagen command performs the schema generator process for either the Java source files or class files.

The current schema generator simply creates a schema file for each namespace referenced in your Java classes.

Execution permission

Superuser

Environment variable

If Java sources/classes reference other classes, they must be accessible on system CLASSPATH environment variable, or they need to be input to the tool by using the -classpath/ -cp options.

Else, errors will occur when generating schema.

Files

Java source files or class files will be input for this command.

Schema file will be output of this command.

Arguments

`-d path`

Specifies where to place the processor and `javac` generated class files.

Type: String

The following values can be specified:

- *Path to place generated files*

Default value: N/A

`-cp path | -classpath path`

Specifies where to find the user specified files.

Type: String

The following values can be specified:

- *Path where user specified files present*

Default value: N/A

`-episode file`

Generates the episode file, which is just a JAXB customization file.

Type: String

The following values can be specified:

- *Name of the episode file to be created*

Default value: N/A

`-version`

Displays the version information.

`-help`

Displays the usage message.

Examples

The following example performs the schema generator process for `Foo.java` and `Bar.java`.

```
schemagen Foo.java Bar.java
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.2.6 wscompile

Generates stubs, ties, serializers, and WSDL files used in JAX-RPC clients and services.

Synopsis

```
wscompile [OPTION] configuration_file

    OPTION:
    [-classpath path]
    [-cp path]
    [-d directory]
    [-define servicename]
    [-f:features]
    [-features:features]
    [-g]
    [-gen]
    [-gen:client]
    [-gen:server]
    [-help]
    [-httpproxy:host:port]
    [-import]
    [-keep]
    [-model file]
    [-nd directory]
    [-O]
    [-s directory]
    [-verbose={false|true}]
    [-version]
    [-mapping file]
    [-security file]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The wscompile command generates stubs, ties, serializers, and WSDL files used in JAX-RPC clients and services. The tool reads a configuration file, which specifies either a WSDL file, a model file, or a compiled service endpoint interface.

Execution permission

Superuser

Files

The tool reads a configuration file, which specifies a WSDL file, a model file, or a compiled service endpoint interface.

Arguments

-classpath *path* | -cp *path*

Specifies where to find input class files.

Type: String

The following values can be specified:

- *Path of the input class files*

Default value: N/A

`-d directory`

Specifies where to place generated output files.

Type: String

The following values can be specified:

- *Directory path to place output files*

Default value: N/A

`-define servicename`

Defines a service.

Type: String

The following values can be specified:

- *Name of the service*

Default value: N/A

`-f:features | -features:features`

Enables the features specified.

Type: String

The following values can be specified:

- `datahandleronly`
- `documentliteral`
- `rpcliteral`
- `explicitcontext`
- `infix:<name>`
- `infix=<name>`
- `jaxbenumtype`
- `nodatabinding`
- `nonencodedtypes`
- `nomultirefs`
- `norpcstructures`
- `novalidation`
- `resolveidref`
- `searchschema`
- `serializeinterfaces`
- `strict`
- `useonewayoperations`
- `wsi`
- `unwrap`
- `donotoverride`
- `donotunwrap`

Default value: N/A

`-g`

Generates the debugging information.

`-gen`

Generates the client artifacts (for example, stubs).

`-gen:client`

Generates the client artifacts (for example, stubs).

`-gen:server`

Generates the server artifacts (for example, ties).

`-help`

Displays help.

`-httpproxy:`*`host`*`:`*`port`*

Specifies a HTTP proxy server (port defaults to `8080`).

Specify the host name of the proxy server for *host*. Specify the port number of the proxy server for *port*.

Type: Integer

The following values can be specified:

- `1` to `65535`

Default value: Port: `8080`

`-import`

Generates interfaces and value types only.

`-keep`

Stores the generated files.

`-model` *`file`*

Writes the internal model to the given file.

Type: String

The following values can be specified:

- *Name of the file*

Default value: N/A

`-nd` *`directory`*

Specifies where to place the non-class generated files.

Type: String

The following values can be specified:

- *Path to the directory*

Default value: N/A

`-O`

Optimizes the generated code.

`-s` *`directory`*

Specifies where to place the generated source files.

Type: String

The following values can be specified:

- *Path to the directory*

Default value: N/A

`-verbose={false|true}`

Outputs messages about the compiler actions.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-version`

Prints the version information.

`-mapping` *`file`*

Writes the JSR-109 mapping file to the given file.

Type: String

The following values can be specified:

- *Name of the file*

Default value: N/A

`-security` *`file`*

Displays the security configuration file.

Type: String

The following values can be specified:

- *Name of the file*

Default value: N/A

Examples

```
wscompile -gen -f:infix:Name -d generated config.xml
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully
1	error in executing the subcommand

2.2.7 wsdeploy

Generates a deployable WAR file for a service.

Synopsis

```
wsdeploy [OPTION] war file

    OPTION:
    [-classpath path]
    [-keep]
    [-o output war file]
```

```
    [-tmpdir directory]
    [-verbose={false|true}]
    [-version]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The wsdeploy command generates a deployable WAR file for a service. This tool takes a WAR file as an input for the service.

Execution permission

Superuser

Files

The tool takes as input a WAR file and generates a deployable WAR file.

Arguments

-classpath *path*

Specifies the path of input class files.

Type: String

The following values can be specified:

- *Path for the class files*

Default value: N/A

-keep

Stores the temporary files.

-o *output war file*

Specifies where to place the generated war file.

Type: String

The following values can be specified:

- *Path to place the WAR file*

Default value: N/A

-tmpdir *directory*

Specifies the temporary directory to use.

Type: String

The following values can be specified:

- *Directory Name with path*

Default value: N/A

-verbose={false|true}

Outputs messages about the compiler actions.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-version`

Prints version information.

Examples

```
wsdeploy -o target.war myapp.war
```

Exit Status

Exit Status	Explanation
`0`	subcommand executed successfully.
`1`	error in executing the subcommand.

2.2.8 wsgen

Generates the JAX-WS portable artifacts used in the JAX-WS web services.

Synopsis

```
wsgen [OPTION]... <SEI>

    OPTION:
    [-classpath path]
    [-cp path]
    [-d directory]
    [-extension={false|true}]
    [-help]
    [-keep={false|true}]
    [-r directory]
    [-s directory]
    [-encoding encoding]
    [-verbose={false|true}]
    [-version]
    [--fullversion]
    [-wsdl[:protocol]]
    [-servicename name]
    [-portname name]
    [-inlineSchemas={false|true}]
    [-x path]
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `wsgen` command generates the JAX-WS portable artifacts used in the JAX-WS Web services. The tool reads a Web service endpoint class and generates all the required artifacts for Web service deployment and invocation.

Execution permission

Superuser

Files

The tool reads a web service endpoint class and generates JAX-WS portable artifacts used in JAX-WS web services.

Arguments

`-classpath` *`path`* | `-cp` *`path`*

Specifies where to find the input class files.

Type: String

The following values can be specified:

- *Path of the input class files*

Default value: N/A

`-d` *`directory`*

Specifies where to place the generated output files.

Type: String

The following values can be specified:

- *Directory path to place output files*

Default value: N/A

`-extension={false|true}`

Allows vendor extensions. Use of extensions may result in applications that are not portable or may not interoperate with other implementations.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-help`

Displays help.

`-keep={false|true}`

Stores the generated files.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-r` *`directory`*

Specifies where to place the generated resource files such as WSDLs.

Used only in conjunction with the `-wsdl` option.

Type: String

The following values can be specified:

- *Directory path to place output files*

Default value: N/A

`-s` *`directory`*

Specifies where to place the generated source files.

Type: String

The following values can be specified:

- *Directory path to place output files*

Default value: *Current working directory*

`-encoding` *`encoding`*

Sets the encoding name for generated sources, such as, `EUC-JP` or `UTF-8`. If `-encoding` is not specified, the platform default encoding is used.

Type: String

The following values can be specified:

- `UTF-8`
- `EUC-JP`

Default value: N/A

`-verbose={false|true}`

Outputs messages about the compiler actions.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-version`

Prints the version information.

`-fullversion`

Prints the full version information.

`-wsdl[:`*`protocol`*`]`

By default, it does not generate a WSDL file.

This flag is optional and will cause `wsgen` to generate a WSDL file and is usually only used, so that the developers can look at the WSDL before the endpoint is deployed. The *protocol* is optional and is used to specify what *protocol* should be used in the `wsdl:binding`.

Valid *protocols* include:

- `soap1.1`
- `Xsoap1.2`

The default is `soap1.1`. `Xsoap1.2` is not standard and can only be used in conjunction with the `-extension` option.

Type: String

The following values can be specified:

- `soap1.1`

- `Xsoap1.2`

Default value: `soap1.1`

`-servicename` *`name`*

Specifies a particular `wsdl:service` name to be generated in the WSDL. For example: `-servicename "{http://mynamespace/}MyService"`

Used only in conjunction with the `-wsdl` option.

Type: String

The following values can be specified:

- *Service name*

Default value: N/A

`-portname` *`name`*

Specifies a particular `wsdl:port` name to be generated in the WSDL. For example: `-portname "{http://mynamespace/}MyPort"`

Used only in conjunction with the `-wsdl` option.

Type: String

The following values can be specified:

- *Name of the port*

Default value: N/A

`-inlineSchemas={false|true}`

Inlines the schemas in a generated wsdl.

Must be used in conjunction with the `-wsdl` option.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-x` *`path`*

Specifies the External Web Service Metadata xml descriptor to be used.

If there are more descriptor files then the option must be used before each descriptor file.

Type: String

The following values can be specified:

- *Path to the file*

Default value: N/A

Examples

This will generate the wrapper classes needed for `StockService` annotated with `@WebService` annotation inside the stock directory.

```
wsgen -d stock -cp myclasspath stock.StockService
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.2.9 wsimport

Generates JAX-WS portable artifacts.

Synopsis

```
wsimport [OPTION]... <WSDL>

    OPTION:
    [-d directory]
    [-b path]
    [-B jaxbOption]
    [-catalog catalogfile]
    [-extension={false|true}]
    [-help]
    [-httpproxy:host:port]
    [-keep={false|true}]
    [-p pkg]
    [-s directory]
    [-encoding encoding]
    [-verbose={false|true}]
    [-version]
    [-fullversion]
    [-clientjar jarfile]
    [-wsdllocation location]
    [-target version]
    [-quiet={false|true}]
    [-XadditionalHeaders={false|true}]
    [-Xauthfile authfile]
    [-Xdebug={false|true}]
    [-XdisableAuthenticator={false|true}]
    [-Xno-addressing-databinding={false|true}]
    [-Xnocompile={false|true}]
    [-XdisableSSLHostnameVerification={false|true}]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The wsimport command generates JAX-WS portable artifacts, such as:

- Service Endpoint Interface (SEI)
- Service
- Exception class mapped from wsdl:fault (if any)
- Async Response Bean derived from response wsdl:message (if any)

- JAXB generated value types (mapped java classes from schema types)

These artifacts can be packaged in a WAR file with the WSDL and schema documents along with the endpoint implementation to be deployed.

Execution permission

Superuser

Files

- This command takes the WSDL file as input.
- This command generates JAX-WS portable artifacts described in "Function".
- For `-b` option this command takes binding files/XML schema as input.

Arguments

`-d` *`directory`*

Specifies where to place the generated output files.

Type: String

The following values can be specified:

- *Name of the directory*

Default value: N/A

`-b` *`path`*

Specifies the external JAX-WS or JAXB binding files or additional schema files (Each `<file>` must have its own `-b`).

Type: String

The following values can be specified:

- *Name of the binding file with path*

Default value: N/A

`-B` *`jaxbOption`*

Pass this option to the JAXB schema compiler.

Type: String

The following values can be specified:

- *Name of the JAXB option*

Default value: N/A

`-catalog` *`catalogfile`*

Specifies the catalog file to resolve external entity references. It supports TR9401, XCatalog, and OASIS XML Catalog formats.

Type: String

The following values can be specified:

- `Path to the catalog file`

Default value: N/A

`-extension={false|true}`

Allows vendor extensions (functionality not specified by the specification). Use of extensions may result in applications that are not portable or may not interoperate with other implementations.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-help`

Displays help.

`-httpproxy:`*`host`*`:`*`port`*

Specifies an HTTP proxy server (port defaults to `8080`).

Type: Integer

The following values can be specified:

- `1` to `65535`

Default value: *port* defaults to `8080`

`-keep={false|true}`

Stores the generated source code files. It is enabled when the `-s` option is provided.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-p` *`pkg`*

Specifies a target package through this command-line option. Overrides any `wsdl` and schema binding customization for the package name and the default package name algorithm defined in the specification.

Type: String

The following values can be specified:

- *Name of the target package*

Default value: `WSDL/Schema target-Namespace` to package mapping as defined by the JAXB 2.1 spec.

`-s` *`directory`*

Specifies where to place the generated source code files. `-keep` is turned on with this option.

Type: String

The following values can be specified:

- *Directory path to place output files*

Default value: *Current working directory*

`-encoding` *`encoding`*

Sets the encoding name for generated sources, such as `EUC-JP` or `UTF-8`. If `-encoding` is not specified, the platform default encoding is used.

Type: String

The following values can be specified:

- `UTF-8`
- `EUC-JP`

Default value: *platform default encoding*

`-verbose={false|true}`

Outputs messages about what the compiler is doing.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-version`

Prints the version information.

`-fullversion`

Prints the full version information.

`-clientjar` *`jarfile`*

Creates the jar file of the generated artifacts along with the WSDL metadata required for invoking the Web service.

Type: String

The following values can be specified:

- *Name of the jar to be created*

Default value: N/A

`-wsdllocation` *`location`*

Specifies the `@WebServiceClient.wsdlLocation` value.

Type: String

The following values can be specified:

- *Mention the location*

Default value: *Defaults to the wsdl URL passed to wsdl attribute*

`-target` *`version`*

Generates the code as per the given JAX-WS specification version. For example, `-target 2.0` generates compliant code for JAX-WS 2.0 spec. The default value is `2.2`.

Type: Integer

The following values can be specified:

- *Till JAX-WS latest version (For example: 2.2.8)*

Default value: `2.2`

`-quiet={false|true}`

Suppresses the `wsimport` output.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

-XadditionalHeaders={false|true}

Maps the headers (that are not bound to request or response message) to Java method parameters.

Type: Boolean

The following values can be specified:

- true
- false

Default value: false

-Xauthfile *authfile*

Specifies the file to carry authorization information in the format http://*username*:*password*@example.org/stock?wsdl. The asterisk character (*) can be used to match multiple URL patterns. The default value is $HOME/.metro/auth.

Type: String

The following values can be specified:

- *URL to file carrying authorization information*

Default value: Defaults to $HOME/.metro/auth.

-Xdebug={false|true}

Prints the debug information.

Type: Boolean

The following values can be specified:

- true
- false

Default value: false

-XdisableAuthenticator={false|true}

Disables the Authenticator used by JAX-WS RI. -Xauthfile option will be ignored if -XdisableAuthenticator is set.

Type: Boolean

The following values can be specified:

- true
- false

Default value: false

-Xno-addressing-databinding={false|true}

Enables binding of W3C EndpointReference-Type to Java.

Type: Boolean

The following values can be specified:

- true
- false

Default value: false

-Xnocompile={false|true}

Does not compile the generated Java files.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`-XdisableSSLHostnameVerification={false|true}`

Disables the SSL Hostname verification while fetching the wsdls.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

Examples

```
wsimport -p stockquote http://stockquote.xyz/quote?wsdl
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.2.10 xjc

xml to java compiler

Synopsis

```
xjc [OPTION]... <schema file/URL/dir/jar> [-b <binding>...]

    OPTION:
    [-nv]
    [-extension]
    [-d dir]
    [-p pkg]
    [-httpproxy proxy]
    [-httpproxyfile file]
    [-classpath arg]
    [-catalog file]
    [-readOnly]
    [-npa]
    [-no-header]
    [-target 2.0|2.1]
    [-encoding encoding]
    [-enableIntrospection]
    [-disableXmlSecurity]
    [-contentForWildcard]
    [-xmlschema]
    [-verbose]
```

```
    [-quiet]
    [-help]
    [-version]
    [-Xinject-code]
    [-Xlocator]
    [-Xsync-methods]
    [-mark-generated]
    [-episode file]
    [-Xpropertyaccessors]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The XJC command produces a set of packages containing Java source files and also jaxb.properties files, depending on the binding options you used for compilation.

When generated, jaxb.properties files must be kept with the compiled source code and made available at runtime for the client applications.

Execution permission

Superuser

Files

- xsd schema file to be passed as input for the XJC command. The XJC command will generate java content classes.
- XJC produces a set of packages containing Java source files and also jaxb.properties files.

Arguments

-nv

Indicates whether to disable strict schema validation. This means that the command will perform less-strict validation.

-extension

Allows vendor extensions. Do not strictly follow the Compatibility Rules and App E.2 from the JAXB Spec.

-d *dir*

Generates the Java content classes in an alternate output directory.

For this option, the output directory must already exist since the XJC binding compiler will not create it.

Type: String

The following values can be specified:

- *Name of the output directory*

Default value: N/A

-p *pkg*

Specifies a target package via this command-line option to override any binding customization for the package name and the default package name algorithm defined in the specification.

Type: String

The following values can be specified:

- *Target package name*

`-httpproxy` *`proxy`*

Specifies the *HTTP/HTTPS proxy*.

The format is:

`[`*`user`*`[:`*`password`*`]@]`*`proxyHost`*`[:`*`proxyPort`*`]`

Type: Integer

The following values can be specified:

- *Specify HTTP/HTTPS proxy*

Default value: N/A

`-httpproxyfile` *`file`*

Specifies the HTTP/HTTPS proxy, but it takes the <proxy> parameter in a file, so that you can protect the password.

The format is:

[*user*[:*password*]@]*proxyHost*[:*proxyPort*]

Type: String

The following values can be specified:

- *Filename with path*

Default value: N/A

`-classpath` *`arg`*

Specifies where to find the client application class files used by the `<jxb:javaType>` and `<xjc:superClass>` customizations.

Type: String

The following values can be specified:

- *Classpath*

Default value: N/A

`-catalog` *`file`*

Specifies the catalog files to resolve external entity references. Supports TR9401, XCatalog, and OASIS XML Catalog formats.

Type: String

The following values can be specified:

- *Path to the catalog file*

Default value: N/A

`-readOnly`

Forces the XJC binding compiler to mark the generated Java sources as read-only.

`-npa`

Suppresses the generation of package level annotations into `**/package-info.java`. Using this switch causes the generated code to internalize those annotations into the other generated classes.

`-no-header`

Suppresses the generation of a file header comment that includes some note and timestamp.

Using this makes the generated code more diff-friendly.

`-target 2.0|2.1`

Avoids generating code that relies on any JAXB 2.1|2.2 features, thereby allowing the generated code to run with JAXB 2.0 runtime (such as JavaSE 6).

Type: Integer

The following values can be specified:

- *JAXB version number*

Default value: N/A

`-encoding` *`encoding`*

Specifies character encoding for the generated source files.

Type: String

The following values can be specified:

- *All the jdk supported Encoding formats*

Default value: *platform default encoding is used*

`-enableIntrospection`

Enables correct generation of `Boolean getters/setters` to enable `Introspection API`.

`-disableXmlSecurity`

Disables the XML security features when parsing XML documents.

`-contentForWildcard`

Generates content property for types with multiple `xs:any` derived elements.

`-xmlschema`

Treats input schemas as W3C XML Schema (default). If you do not specify this switch, your input schemas will be treated as W3C XML Schema.

Default value: W3C XML Schema

`-verbose`

Prints informational messages or displays stack traces when some errors occur. This information can be an extra verbose.

`-quiet`

Suppresses the compiler output, such as progress information and warnings.

`-help`

Displays a brief summary of the compiler switches.

`-version`

Displays the compiler version information.

`-Xinject-code`

Injects the specified Java code fragments into the generated code.

`-Xlocator`

Causes the generated code to expose SAX Locator information about the source XML in the Java bean instances after unmarshalling.

`-Xsync-methods`

Causes all the generated method signatures to include the synchronized keyword.

`-mark-generated`

Causes all the generated code to have the @Generated [`http://java.sun.com/javaee/5/docs/api/jaxb-2-0-overview? javax/annotation/Generated.html`] annotation.

`-episode file`

Generates an episode file from this compilation, so that, other schemas that rely on this schema can be compiled later and rely on classes that are generated from this compilation. The generated episode file is just a JAXB customization file (includes vendor extensions).

Type: String

The following values can be specified:

- *Name of the episode file to be created*

Default value: N/A

`-Xpropertyaccessors`

Annotates the `@XmlAccessorType` of generated classes with `XmlAccessType PROPERTY` instead of `FIELD`.

`schema file/URL/dir`

Specifies one or more schema files to compile. If you specify a directory, then `xjc` will scan it for all the schema files and compile them.

Type: String

The following values can be specified:

- *Path of schema files/directory*

Default value: N/A

`-b binding`

Specifies one or more external binding files to process.

Each binding file must have its own `-b` switch.

You may have a single binding file that contains customizations for multiple schemas or you can break the customizations into multiple binding files.

In addition, the ordering of the schema files and binding files on the command line does not matter.

Type: String

The following values can be specified:

- *Name of the binding file*

Default value: N/A

Examples

Example1:

```
xjc schema1.xsd schema2.xsd schema3.xsd -b bindings123.xjb
```

Example2:

```
xjc schema1.xsd schema2.xsd schema3.xsd -b bindings1.xjb -b bindings2.xjb
-b bindings3.xjb
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully
1	error in executing the subcommand

2.3 List of subcommands of asadmin command to be used in the Java EE server

The following table shows a list of subcommands of asadmin command to be used in the Java EE server.

Commands used for domain administration

Command Name	Classification	Summary
backup-domain	Creates a backup of the domain	Creates a backup of the files in a named domain.
change-admin-password	Changes the administrator password	Changes the administrator password.
create-domain	Creates Java EE Server domain	Creates Java EE Server domain.
create-service	Configures the starting of the DAS or server instance	Configures the starting of the DAS or server instance on an unattended boot on OS.
delete-domain	Deletes a domain	Deletes the specified domain.
list-backups	Lists all the backups	Displays information about the domain backups.
list-domains	Lists the domains in the specified directory	Lists the domains in the specified domains directory.
restart-domain	Restarts the DAS of the specified domain	Stops and then restarts the DAS of the specified domain.
restore-domain	Restores files from the backup	Restores files under the domain from a backup directory.
start-domain	Starts the DAS of the specified domain	Starts the DAS of the specified domain.
stop-domain	Stops the DAS of the specified domain	Stops the DAS of the specified domain.

Commands used for node administration

Command Name	Classification	Summary
create-node-config	Creates a node that is not enabled for remote communication	Creates a node that is not enabled for remote communication.
create-node-ssh	Creates a node that is enabled for communication over SSH	Creates a node that is enabled for communication over SSH.
delete-node-config	Deletes a node that is not enabled for remote communication	Deletes a node that is not enabled for remote communication.
delete-node-ssh	Deletes a node that is enabled for communication over SSH	Deletes a node that is enabled for communication over SSH from the domain.
list-nodes	Lists all Java EE Server nodes in a domain	Lists all Java EE Server nodes in a domain.
list-nodes-config	Lists all Java EE Server nodes that do not support remote communication in a domain	Lists all Java EE Server nodes that do not support remote communication in a domain.
list-nodes-ssh	Lists all Java EE Server nodes that support communication over SSH in a domain	Lists all Java EE Server nodes that support communication over SSH in a domain.
ping-node-ssh	Tests nodes enabled for communication over SSH to determine if they are usable	Validates the node to determine if the `asadmin` utility can run on the host which the node represents.
setup-ssh	Sets up an SSH key on specified hosts	Sets up an SSH key on specified hosts.

Command Name	Classification	Summary
update-node-config	Updates the configuration data of a node	Updates the configuration data of a node. This subcommand does not require the SSH to be configured to update the node.
update-node-ssh	Updates the configuration data of a node	Updates the configuration data of a node. This subcommand requires SSH to be configured on the machine where the DAS is running and on the machine where the node resides.

Commands used for server administration

Command Name	Classification	Summary
cleanup-prf	Cleaning PRF environment	Stops the PRF servers that were created on the host running the subcommand, and removes (cleans up) the PRF environment.
cleanup-webserver	Cleaning web server environment	Stops a web server that was set up on the host running the subcommand. This subcommand removes (cleans up) the web server environment.
copy-config	Copies an existing named configuration to create another configuration	Creates a configuration in the DAS by copying an existing configuration.
create-cluster	Creates Java EE Server cluster	Creates Java EE Server cluster.
create-instance	Creates a server instance	Creates a server instance.
create-prf	Creating PRF server in domain	Creates a PRF server in the domain.
create-webserver	Creating web server	Creates a web server in the domain.
delete-cluster	Deletes Java EE Server cluster	Deletes Java EE Server cluster.
delete-config	Deletes an existing named configuration	Deletes an existing named configuration from the configuration of the DAS.
delete-instance	Deletes a server instance	Deletes a server instance.
delete-prf	Deleting PRF server	Deletes the specified PRF server from the domain.
delete-webserver	Deleting web server	Deletes the specified web server from the domain.
export-sync-bundle	Exports the configuration data of a cluster or standalone instance to an archive file	Exports the configuration data of a cluster or standalone instance to an archive file.
get-health	Provides information on the cluster health	Gets information about the health of the cluster.
import-sync-bundle	Imports the configuration data of a clustered instance or standalone instance from an archive file	Imports the configuration data of a clustered instance or standalone instance from an archive file that was created by the `export-sync-bundle` subcommand.
list-clusters	Lists existing clusters in a domain	Lists existing clusters in a domain.
list-instances	Lists the server instances	Lists the server instances in a domain.
list-prfs	Listing PRF servers	Lists the PRF (Performance Tracer) servers that have been set up in the domain.
list-webservers	Listing all web servers	Lists all the web servers that have been set up in the domain.
start-cluster	Starts a cluster	Starts a cluster by starting all the server instances in the cluster that are not already running.
start-instance	Starts a server instance	Starts a server instance.
start-prf	Starting PRF server	Starts the specified PRF server.

Command Name	Classification	Summary
start-servers	Starting all the servers in batch	Starts, in a batch, all the servers in the domain.
start-webserver	Starting web server	Starts the specified web server.
stop-cluster	Stops Java EE Server cluster	Stops Java EE Server cluster by stopping all running server instances in the cluster.
stop-instance	Stops a running server instance	Stops a running server instance.
stop-prf	Stopping PRF server	Stops the specified PRF server.
stop-servers	Stopping all the servers in batch	Stops, in a batch, all the servers in the domain.
stop-webserver	Stopping web server	Stops the specified web server.
validate-multicast	Validates that the multicast transport is available for clusters	Validates that the multicast transport is available for clusters.

Commands used for setting server relations

Command Name	Classification	Summary
create-relation	Creating relation between servers	Creates a relation between servers.
delete-relation	Deleting relation between servers	Deletes a relation between servers from the domain.
list-relations	Listing relations between servers	Lists the relations between servers.

Commands used for application administration

Command Name	Classification	Summary
create-application-ref	Creates the reference to an application	Creates a reference from a clustered or an unclustered server instance to a previously deployed application element.
delete-application-ref	Removes the reference to an application	Removes a reference from a cluster or an unclustered server instance to an application.
deploy	Deploys the specified component	Deploys applications to the server.
disable	Disables the component	Disables the specified deployed component.
enable	Enables the component	Enables the specified deployed component.
get-client-stubs	Retrieves the application JAR files needed to launch the application client	Copies the required JAR files for an AppClient standalone module or each AppClient module in an application from the server machine to the local directory.
list-application-refs	Lists the existing application references	Lists all the application references in a cluster or an unclustered server instance.
list-applications	Lists the deployed applications	Lists the deployed Java EE applications and the type of each application that is listed.
list-containers	Lists the application containers	Lists the application containers.
list-modules	Lists Java EE Server modules	Displays a list of modules that are accessible to Java EE Server module subsystem.
list-sub-components	Lists EJB or servlet components either in a deployed module or in the module of a deployed application	Lists the EJB or servlet components in a deployed module or in a module of a deployed application.
list-timers	Lists all the persistent timers owned by the server instance(s)	Lists the persistent timers owned by a specific server instance or a cluster of server instances.

Command Name	Classification	Summary
list-web-context-param	Lists the servlet context-initialization parameters of a deployed Web application or module	Lists the servlet context-initialization parameters of a deployed Web application or module.
list-web-env-entry	Lists the environment entries for a deployed Web application or module	Lists the environment entries for a deployed Web application or module.
redeploy	Redeploys the specified component	Redeploys an enterprise application, Web application, module based on the Enterprise JavaBeans (EJB) specification (EJB module), connector module, or application client module that is already deployed or already exists.
set-web-context-param	Sets a servlet `context-initialization` parameter of a deployed Web application or module	Sets a servlet `context-initialization` parameter of a deployed Web application or module.
set-web-env-entry	Sets an environment entry for a deployed Web application or module	Sets an environment entry for a deployed Web application or module.
show-component-status	Displays the status of the deployed component	Gets the status (either enabled or disabled) of the deployed component.
undeploy	Removes a deployed component	Uninstalls a deployed application or module and removes it from the repository.
unset-web-context-param	Unsets a servlet context-initialization parameter of a deployed Web application or module	Unsets a servlet context-initialization parameter of a deployed Web application or module.
unset-web-env-entry	Unsets an environment entry for a deployed Web application or module	Unsets an environment entry for a deployed Web application or module.

Commands used for security administration

Command Name	Classification	Summary
change-master-password	Changes the master password	Changes the master password.
create-auth-realm	Adds the named authentication realm	Adds the named authentication realm.
create-file-user	Creates a new file user	Creates an entry in the keyfile
create-jacc-provider	Enables administrators to create a JACC provider that can be used by third-party authorization modules for applications running in Java EE Server	Creates a JSR-115-compliant Java Authorization Contract for Containers (JACC) provider that can be used for authorization of applications running in Java EE Server.
create-password-alias	Creates a password alias	Creates an alias for a password.
delete-auth-realm	Removes the named authentication realm	Removes the named authentication realm.
delete-file-user	Removes the named file user	Deletes the entry in the keyfile for the specified username.
delete-jacc-provider	Deletes JACC providers defined for a domain	Enables administrators to delete JACC providers defined for a domain.
delete-password-alias	Deletes a password alias	Deletes a password alias.
disable-secure-admin	Disables secure admin	Disables secure admin if it is already enabled.
enable-secure-admin	Enables secure admin	Enables secure admin if it is not already enabled.
list-auth-realms	Lists the authentication realms	Lists the authentication realms.

Command Name	Classification	Summary
list-file-groups	Lists the file groups	Lists the file users and groups supported by the file realm authentication.
list-file-users	Lists the file users	Displays a list of file users supported by file realm authentication.
list-jacc-providers	Lists JACC providers defined for a domain	Enables administrators to list the JACC providers defined for a domain.
list-password-aliases	Lists all the password aliases	Lists all the existing password aliases.
list-supported-cipher-suites	Lists the cipher suites that are supported and made available to a specific Java EE Server target	Enables administrators to list the cipher suites that are supported and made available to a specified Java EE Server target.
update-file-user	Updates a current `file user` as specified	Updates an existing entry in the keyfile using the specified user name, password and groups.
update-password-alias	Updates a password alias	Updates the password alias IDs in the named target.

Commands used for log administration

Command Name	Classification	Summary
collect-local-snapshot	Collecting system information when domain administration server stops	Collects system information when the domain administration server stops.
collect-snapshot	Collecting system information when domain administration server starts	Collects system information when the domain administration server starts.
list-log-attributes	Lists all the logging attributes defined for a specific target in a domain	Lists all the logging attributes currently defined for the specified Java EE Server domain or a target within a domain.
set-log-attributes	Sets the logging attributes for one or more loggers	Sets the logging attributes for one or more loggers.

Commands used for setting parameters

Command Name	Classification	Summary
create-jvm-options	Creates options for Java VM	Creates command-line options that are passed to Java VM when Java EE Server is started.
create-system-properties	Adds one or more system property elements that can be referenced elsewhere in the configuration	Adds or updates system properties that can be referenced elsewhere on the server.
delete-jvm-options	Removes one or more options for Java VM	Removes one or more command-line options for Java VM.
delete-system-property	Removes a system property of the domain, configuration, cluster, or server instance, one at a time	Deletes a system property of a domain, configuration, cluster, or server instance.
get	Retrieves the values of configurable attributes	Retrieves the values of configurable attributes.
list	Lists the configurable elements	Lists the configurable attributes of Java EE Server.
list-configs	Lists the named configurations in the configuration of the DAS	Lists named configurations in the configuration of the DAS.
list-jvm-options	Lists options for Java VM	Displays a list of command-line options that are passed to Java VM when the Java EE Server is started.

Command Name	Classification	Summary
list-system-properties	Lists the system properties of the domain, configuration, cluster, or a server instance	Lists the system properties of a domain, configuration, cluster, or server instance.
set	Sets the values of configurable attributes	Sets the values of configurable attributes.

Commands used for system administration

Command Name	Classification	Summary
add-library	Adds one or more library JAR files to Java EE Server	Adds one or more library archive files to Java EE Server.
create-profiler	Creates the profiler element	Creates the profiler element.
delete-profiler	Removes the profiler element	Deletes the profiler element in the Java configuration.
list-commands	Lists the available commands	Lists the `asadmin` subcommands.
list-libraries	Lists the library archive files present on Java EE Server	Lists library archive files present on Java EE Server.
login	Logs you into a domain	Enables you to log into a particular domain, thus simplifying domain administration.
remove-library	Removes one or more library JAR files from Java EE Server	Removes one or more library JAR files from Java EE Server.
version	Displays the version information for Java EE Server	Displays the version information for Java EE Server.

Commands used for internet connection administration

Command Name	Classification	Summary
create-protocol	Adds a new protocol	Adds a new protocol.
create-ssl	Creates and configures the SSL element in the selected HTTP listener, IIOP listener, or IIOP service	Creates and configures the SSL element in the selected HTTP listener, IIOP listener, or IIOP service to enable secure communication on that listener/service.
delete-protocol	Removes a protocol	Removes the specified protocol.
delete-ssl	Deletes the SSL element in the selected HTTP listener, IIOP listener, or IIOP service	Deletes the SSL element in the selected HTTP listener, IIOP listener, or IIOP service.
list-network-listeners	Lists the existing network listeners	Lists the existing network listeners.
list-protocols	Lists the existing protocols	Lists the existing protocols.
list-transports	Lists the existing transports	Lists the existing transports.
list-virtual-servers	Lists the existing virtual servers	Lists the existing virtual servers.

Commands used for ORB administration

Command Name	Classification	Summary
create-iiop-listener	Adds an IIOP listener	Adds an IIOP listener.
delete-iiop-listener	Removes an IIOP listener	Removes an IIOP listener.

Command Name	Classification	Summary
list-iiop-listeners	Lists the existing IIOP listeners	Lists the existing IIOP listeners.

Commands used for resource administration

Command Name	Classification	Summary
add-resources	Creates the resources named in a specified XML file	Creates the resources named in a specified XML file.
create-resource-ref	Creates a reference to a resource	Creates a reference from a cluster or an unclustered server instance to a previously created resource.
delete-resource-ref	Removes a reference to a resource	Removes from a cluster or an unclustered server instance a reference to a resource (for example, a JDBC resource).
list-resource-refs	Lists the existing resource references	Lists all the resource references in a cluster or an unclustered server instance.

Commands used for database connection administration

Command Name	Classification	Summary
create-jdbc-connection-pool	Registers a JDBC connection pool	Registers a new Java Database Connectivity ("JDBC") software connection pool with the specified JDBC connection pool name.
create-jdbc-resource	Creates a JDBC resource with the specified JNDI name	Creates a new JDBC resource.
delete-jdbc-connection-pool	Removes the specified JDBC connection pool	Deletes a JDBC connection pool.
delete-jdbc-resource	Removes a JDBC resource with the specified JNDI name	Removes a JDBC resource with the specified JNDI name.
list-jdbc-connection-pools	Lists all JDBC connection pools	Lists the current JDBC connection pools.
list-jdbc-resources	Lists all JDBC resources	Displays a list of all the existing JDBC resources.

Commands used for JavaMail service administration

Command Name	Classification	Summary
create-javamail-resource	Creates a JavaMail session resource	Creates a JavaMail session resource.
delete-javamail-resource	Removes a JavaMail session resource	Removes the specified JavaMail session resource.
list-javamail-resources	Lists the existing JavaMail session resources	Lists the existing JavaMail session resources.

Commands used for JNDI service administration

Command Name	Classification	Summary
create-custom-resource	Registers the connector resource with the specified JNDI name	Registers the connector resource with the specified JNDI name.
create-jndi-resource	Registers a JNDI resource	Registers a JNDI resource.
delete-custom-resource	Deletes a custom resource	Deletes a custom resource.

Command Name	Classification	Summary
delete-jndi-resource	Removes a JNDI resource	Removes the specified JNDI resource.
list-custom-resources	Gets all the custom resources	Gets all the custom resources.
list-jndi-resources	Lists all the existing JNDI resources	Identifies all the existing JNDI resources.

Commands used for JMS administration

Command Name	Classification	Summary
create-jms-host	Creates a JMS host	Creates a Java Message Service (JMS) host within the JMS service.
create-jms-resource	Creates a JMS resource	Creates a Java Message Service (JMS) connection factory resource or a JMS destination resource.
create-jmsdest	Creates a JMS physical destination	Creates a Java Message Service (JMS) physical destination.
delete-jms-host	Removes a JMS host	Removes a JMS host.
delete-jms-resource	Removes a JMS resource	Removes a JMS resource.
delete-jmsdest	Removes a JMS physical destination	Removes a JMS physical destination.
flush-jmsdest	Purges messages in a JMS destination	Purges the messages from a physical destination in the server's Java Message Service (JMS) configuration.
list-jms-hosts	Lists the existing JMS hosts	Lists the existing Java Message Service (JMS) hosts for the JMS service.
list-jms-resources	Lists the JMS resources	Lists the existing Java Message Service (JMS) resources (destination and connection factory resources).
list-jmsdest	Lists the existing JMS physical destinations	Lists the Java Message Service (JMS) physical destinations.

Commands used for EIS connection administration

Command Name	Classification	Summary
create-admin-object	Creates an administered object with the specified JNDI name for a resource adapter	Creates the administered object with the specified JNDI name and the interface definition for a resource adapter.
create-connector-connection-pool	Adds a connection pool with the specified connection pool name	Defines a pool of connections to an Enterprise Information System (EIS).
create-connector-resource	Registers the connector resource with the specified JNDI name	Registers the connector resource with the specified JNDI name.
create-connector-security-map	Creates a security map for the specified connector connection pool	Creates a security map for the specified connector connection pool.
create-connector-work-security-map	Creates a work security map for the specified resource adapter	Maps the caller identity of the work submitted by the resource adapter EIS principal or EIS user group to a suitable principal or user group in Java EE Server security domain.
create-resource-adapter-config	Creates the configuration information for the connector module	Creates configuration information for the connector module.
delete-admin-object	Removes the administered object with the specified JNDI name	Removes an administered object with the specified JNDI name.
delete-connector-connection-pool	Removes the specified connector connection pool	Removes the specified connector connection pool.

Command Name	Classification	Summary
delete-connector-resource	Removes the connector resource for the specified JNDI name	Removes the connector resource with the specified JNDI name.
delete-connector-security-map	Deletes a security map for the specified connector connection pool	Deletes a security map for the specified connector connection pool.
delete-connector-work-security-map	Deletes a work security map for the specified resource adapter	Deletes a security map associated with the specified resource adapter.
delete-resource-adapter-config	Deletes the resource adapter configuration	Deletes the configuration information for the connector module.
flush-connection-pool	Reinitializes all connections established in the specified connection pool	Resets a JDBC connection pool or a connector connection pool to its initial state.
list-admin-objects	Gets all the administered objects	Lists all the administered objects.
list-connector-connection-pools	Lists the existing connector connection pools	Lists the connector connection pools that have been created.
list-connector-resources	Lists all connector resources	Lists all the connector resources.
list-connector-security-maps	Lists the security maps belonging to the specified connector connection pool	Lists the security maps belonging to the specified connector connection pool.
list-connector-work-security-maps	Lists the work security maps belonging to the specified resource adapter	Lists the work security maps belonging to the specified resource adapter.
list-resource-adapter-configs	Lists the names of the current resource adapter configurations	Lists the configuration information in the`domain.xml` for the connector module.
ping-connection-pool	Tests a connection pool to determine if it is usable	Tests if an existing JDBC or connector connection pool is usable.
update-connector-security-map	Modifies a security map for the specified connector connection pool	Modifies a security map for the specified connector connection pool.
update-connector-work-security-map	Modifies a work security map for the specified resource adapter	Modifies a security map for the specified resource adapter.

Commands used for thread pool administration

Command Name	Classification	Summary
create-threadpool	Creates a thread pool	Creates a thread pool with the specified name.
delete-threadpool	Removes a thread pool	Removes the thread pool with the specified ID.
list-threadpools	Lists all the thread pools	Lists Java EE Server thread pools.

Commands used for transaction administration

Command Name	Classification	Summary
recover-transactions	Recovers pending transactions manually	Recovers transactions after a server failure.

Commands used for batch job administration

Command Name	Classification	Summary
list-batch-job-executions	Lists the batch job executions and execution details	Lists the batch job executions and execution details.

Command Name	Classification	Summary
list-batch-job-steps	Lists the steps for a specific batch job execution	Lists the steps for a specific batch job execution.
list-batch-jobs	Lists the batch jobs	Lists the batch jobs and job details.
list-batch-runtime-configuration	Displays the configuration of the batch runtime	Displays the configuration of the batch runtime.
set-batch-runtime-configuration	Configures the batch runtime	Configures the batch runtime.

Commands used for concurrent resource administration

Command Name	Classification	Summary
create-context-service	Creates a context service resource	Creates a context service resource.
create-managed-executor-service	Creates a managed executor service resource	Creates a managed executor service resource.
create-managed-scheduled-executor-service	Creates a managed scheduled executor service resource	Creates a managed scheduled executor service resource.
create-managed-thread-factory	Creates a managed thread factory resource	Creates a managed thread factory resource.
delete-context-service	Deletes a context service resource	Removes a context service resource with the specified JNDI name.
delete-managed-executor-service	Removes a managed executor service resource	Removes a managed executor service resource with the specified JNDI name.
delete-managed-scheduled-executor-service	Removes a managed scheduled executor service resource	Removes a managed scheduled executor service resource with the specified JNDI name.
delete-managed-thread-factory	Removes a managed thread factory resource	Removes a managed thread factory resource with the specified JNDI name.
list-context-services	Lists the context service resources	Lists the context service resources.
list-managed-executor-services	Lists the managed executor service resources	Lists the managed executor service resources.
list-managed-scheduled-executor-services	Lists the managed scheduled executor service resources	Lists the managed scheduled executor service resources.
list-managed-thread-factories	Lists the managed thread factory resources	Lists the managed thread factory resources.

Commands used for message security provider administration

Command Name	Classification	Summary
create-message-security-provider	Creates a message security provider	Creates a message security provider that specifies how SOAP messages will be secured.
delete-message-security-provider	Deletes a message security provider	Enables administrators to delete a message security provider.
list-message-security-providers	Lists all the security message providers for the given message layer	Enables administrators to list all security message providers (`provider-config` sub-elements) for the given message layer (`message-security-config` element of `domain.xml`).

Commands used to manage the event hook command

Command Name	Classification	Summary
create-event-hook	Registering event hook command	Registers an event hook command, which is executed automatically when an event occurs in the domain administration server.
delete-event-hook	Unregistering event hook command	Unregisters an event hook command.
list-event-hooks	Listing event hook commands	Lists the registered event hook commands.

Important note

- Specify the value which can be specified by each command as the argument of a command. Operation is not guaranteed when other values are specified.
- Even if a command is executed normally, errors may have occurred. Check the message log and stack trace log.

2.4 Commands used for domain administration

This section describes the syntax and functionality of the commands used for domain administration.

2.4.1 backup-domain

Creates a backup of the domain.

Synopsis

```
asadmin [asadmin-options] backup-domain [--help]
    [--long[={false|true}]] [--description description-text]
    [--domaindir domain-root-dir] [--backupdir backup-directory]
    [domain_name]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `backup-domain` subcommand creates a backup of the files in a named domain.

This command is supported in local mode only in Application Server. In Application Server, the domain to be backed up must be stopped.

Precondition

The domain to be backed up must be stopped.

Files

When the `backup-domain` subcommand is executed, a ZIP archive file, containing all the files of the domain being backed up, is created in the folder specified by the `--backupdir` option of this command.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--long[={false|true}] | -l`

Displays detailed information about the backup operation.

Type: Boolean

The following values can be specified:

- `true`
  Displays detail information.
- `false`
  Does not display detail information.

Default value: `false`

`--description` *description-text*

Specifies a description to store in the backup file. The description is displayed as part of the information about a backup file.

Type: String

The following values can be specified:

- *Text describing the backup*

Default value: Displays the default value in the following format:

domain-name backup created on *YYYY_MM_DD* by user *user-name*

`--domaindir` *domain-root-dir*

Specifies the domain root directory, the parent directory of the domain to back up.

Type: String

The following values can be specified:

- *Specify the domain directory path*

Default value: *Application Server installation directory*`/javaee/glassfish/domains`

`--backupdir` *backup-directory*

Specifies the directory under which the backup file is to be stored.

Type: String

The following values can be specified:

- *File path for backup*

Default value: *domain-dir*`/backups`

domain_name

Specifies the name of the domain to be backed up.

Type: String

The following values can be specified:

- *Name of a domain*

Default value: N/A

Examples

The following example creates a backup of the default domain, `domain1`, storing the backup file in `/net/backups.example.com/glassfish`.

```
asadmin backup-domain --backupdir /net/backups.example.com/glassfish
domain1
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.4.2 change-admin-password

Changes the administrator password.

Synopsis

```
asadmin [asadmin-options] change-admin-password [--help]
    [--domaindir domain-root-dir [--domain_name domain-name]]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `change-admin-password` subcommand of asadmin changes the administrator password. The `change-admin-password` subcommand is interactive, because it prompts the user for the old administrator password, the new administrator password, and for confirmation of the new password. The new password must contain at least `8` characters.

This subcommand fails in the following instances:

- If you are an anonymous user without a password.
- If a blank password is provided (only when secure administration is enabled).

If more than one administrator is configured on Java EE Server, you must run the `asadmin` command with the `--user` option to change the password for a specific user.

This subcommand is supported in both local and remote modes.

Precondition

If more than one administrator is configured for Java EE Server, the user must run the `asadmin` command with the `--user` option to change the password for that user.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--domaindir` *domain-root-dir*

Specifies the parent directory of the domain specified in the `--domain_name` option. When this option is used, the `change-admin-password` subcommand operates in local mode.

Type: String

The following values can be specified:

- *Path of a domain directory*

Default value: N/A

`--domain_name` *domain-name*

Specifies the domain of the admin user. This option is not required if the directory specified by the `--domaindir` option contains only one domain.

Type: String

The following values can be specified:

- *Path of a domain name*

Default value: N/A

Examples

The following example changes the Administrator Password for a single user in single mode.

```
asadmin --user admin change-admin-password
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.4.3 create-domain

Creates Java EE Server domain.

Synopsis

```
asadmin [asadmin-options] create-domain [--help]
    [--adminport adminport] [--instanceport instanceport]
    [--portbase portbase] [--template template-name]
    [--domaindir domaindir] [--savemasterpassword={false|true}]
    [--usemasterpassword={false|true}]
    [--domainproperties name=value[:name=value]...]
    [--keytooloptions name=value[:name=value]...]
    [--savelogin={false|true}] [--checkports={true|false}]
    [--nopassword={false|true}] domain-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-domain` subcommand creates Java EE Server domain. A domain in Java EE Server is an administrative namespace that complies with the Java Platform, Enterprise Edition (Java EE) standard.

Every domain has a configuration, which is stored in a set of files. Any number of domains, each of which has a distinct administrative identity, can be created in a given installation of Application Server. A domain can exist independently of other domains. Any user who has access to the `asadmin` utility on a given system can create a domain and store its configuration in a folder of the user's choice. By default, the domain configuration is created in the default directory for domains. You can override this location to store the configuration elsewhere.

This command creates a domain with a single administrative user specified by the `asadmin` utility option `--user`.

- If the `--user` option is not specified, and the `--nopassword` option is set to `true`, the default administrative user, `admin`, is used.

- If the `--nopassword` option is set to `false` (the default), a username is required. In this case, if you have not specified the user name by using the `--user` option, you are prompted to do so.

You choose an appropriate profile for the domain, depending on the applications that you want to run on your new domain. You can choose the developer, cluster, or enterprise profile for the domain you create.

This command is supported in local mode only.

Execution permission

On UNIX, creating sockets that listen on ports `1` to `1024` requires superuser privileges.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--adminport` *`adminport`*

Specifies the HTTP port or the HTTPS port for administration. This port is the port number in the URL that you specify in your web browser to manage the domain, for example, `http://localhost:4949`. The `--adminport` option cannot be used with the `--portbase` option. The default value is `4848`.

The `--adminport` option overrides the `domain.adminPort` property of the `--domainproperties` option.

Type: Integer

The following values can be specified:

- `1` to `65535`

Default value: `4848`

`--instanceport` *`instanceport`*

Specifies the domain that provides services so that applications can run when deployed. This HTTP port specifies where the web application context roots are available for a web browser to connect to it. This port is a positive integer and must be available while creating the domain. The `--instanceport` option cannot be used with the `--portbase` option. The default value is `8080`. The `--instanceport` option overrides the `domain.instancePort` property of the `--domainproperties` option.

Type: Integer

The following values can be specified:

- `1` to `65535`

Default value: `8080`

`--portbase` *`portbase`*

Determines the number with which port assignments should start. A domain uses a certain number of ports that are statically assigned. The *portbase* value determines where the assignment should start. The values for the ports are calculated as follows:

- Administration port: portbase + 48
- HTTP listener port: portbase + 80
- HTTPS listener port: portbase + 81
- JMS port: portbase + 76
- IIOP listener port: portbase + 37

- Secure IIOP listener port: portbase + 38
- Secure IIOP with mutual authentication port: portbase + 39
- JMX port: portbase + 86
- JPDA debugger port: portbase + 9
- Felix shell service port for OSGi module management: portbase + 66

When the `--portbase` option is specified, the output of this subcommand includes a complete list of used ports.

The `--portbase` option cannot be used with the `--adminport`, `--instanceport`, or the `--domainproperties` option.

Type: Integer

The following values can be specified:

- `-8` to `65449`

Default value: N/A

`--template` *`template-name`*

Specifies the file name, including a relative or absolute path, of a domain configuration template to use for creating the domain.

If a relative path is specified, the subcommand appends the path to the *`Application Server installation directory`*`/javaee/glassfish/lib/templates` directory to locate the file. If it is an absolute pathname, the subcommand locates the file in the specified path. This option enables you to create different types of domains and define custom domain templates. The jar file is the input file for this option which contains the domain configuration file.

Type: String

The following values can be specified:

- *Specify file name of domain configuration*

Default value: N/A

`--domaindir` *`domaindir`*

Specifies the directory where the domain is to be created. If specified, the path must be accessible in the file system. If not specified, the domain is created in the default domain directory, *`Application Server installation directory`*`/javaee/glassfish/domains`.

Type: String

The following values can be specified:

- *Domain directory path*

Default value: *`Application Server installation directory`*`/javaee/glassfish/domains`

`--savemasterpassword={false|true}`

Setting this option to `true` allows the master password to be written to the file system. The default value is `false`.

A master password is a password for the secure key store. A domain is designed to keep its own certificate (created while creating a domain) in a safe place in the configuration location. This certificate is called the domain's SSL server certificate. When the domain is contacted by a web browser over a secure channel (HTTPS), this certificate is presented by the domain. The master password protects the store (a file) that contains this certificate. This file is called `keystore.jks` and is created in the configuration directory of the created domain. If this option is chosen, then the master password is saved on the disk in the configuration location of the domain. The master password is stored in the `master-password` file, which is a Java JCEKS type keystore. The `--savemasterpassword` option is used during an unattended system restart. In this case, the master password is not prompted when the domain starts because the password is extracted from this file.

It is a best practice to create a master password when creating a domain, because the master password is used by the `start-domain` subcommand. For security purposes, the default setting should be set to `false`, because saving the master password on the disk is not secure, unless file system permissions are set properly. If the master password is saved, then the `start-domain` subcommand does not prompt for it. The master password gives an extra level of security to the environment.

If this option is set to `true`, then the `--usemasterpassword` option is also set to `true`, irrespective of the value that is specified on the command line.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--usemasterpassword={false|true}`

Specifies whether the key store is encrypted with a master password that is built into the system or a user-defined master password. If the value is set to `false` (default), then the keystore is encrypted with a well-known password that is built into the system. Encrypting the keystore with a password that is built into the system does not provide additional security. If this value is set to `true`, then the subcommand obtains the master password from the `AS_ADMIN_ MASTERPASSWORD` entry in the password file or prompts for the master password. The password file is specified in the `--passwordfile` option of the asadmin utility command. If the `--savemasterpassword` option is set to `true`, then this option is also set to `true`, irrespective of the value that is specified in the command line.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--domainproperties` *`name=value`*`[:`*`name=value`*`]...`

Specifies the optional *name*/*value* pairs that override the default values for the properties of the domain to be created. The list must be separated by a colon (`:`). The `--portbase` options cannot be used with the `--domainproperties` option.

The `--portbase` options cannot be used with the `--domainproperties` option.

On UNIX, creating sockets that listen on ports `1` to `1024` requires superuser privileges.

Type: String

The following values can be specified:

- `domain.adminPort=`*`value`*

  Specifies the port number of the HTTP port or the HTTPS port for administration. This port is the port in the URL that you specify in your web browser to manage the instance, for example, `http://localhost:4949`. Valid values are `1-65535`.

  The `domain.adminPort` property is overridden by the `--adminport` option.

  Type: Integer

  Default value: N/A

  Range Value: `1` to `65535`

- `domain.instancePort=`*`value`*

  Specifies the port number of the port that is used to listen for HTTP requests. Valid values are `1-65535`.

The `domain.instancePort` property is overridden by `--instanceport` option.

Type: Integer

Default value: N/A

Range Value: `1` to `65535`

- `domain.jmxPort=`*`value`*

  Specifies the port number on which the JMX connector listens. Valid values are `1-65535`.

  Type: Integer

  Default value: N/A

  Range Value: `1` to `65535`

- `http.ssl.port=`*`value`*

  Specifies the port number of the port that is used to listen for HTTPS requests. Valid values are `1-65535`.

  Type: Integer

  Default value: N/A

  Range Value: `1` to `65535`

- `java.debugger.port=`*`value`*

  Specifies the port number of the port that is used for connections to the Java Platform Debugger Architecture (JPDA) (`http://java.sun.com/javase/technologies/core/toolsapis/jpda/`) debugger. Valid values are `1-65535`.

  Type: Integer

  Default value: N/A

  Range Value: `1` to `65535`

- `jms.port=`*`value`*

  Specifies the port number for the Java Message Service provider. Valid values are `1-65535`.

  Type: Integer

  Default value: N/A

  Range Value: `1` to `65535`

- `orb.listener.port=`*`value`*

  Specifies the port number of the port that is used for IIOP connections. Valid values are `1-65535`.

  Type: Integer

  Default value: N/A

  Range Value: `1` to `65535`

- `orb.mutualauth.port=`*`value`*

  Specifies the port number of the port that is used for secure IIOP connections with client authentication. Valid values are `1-65535`.

  Type: Integer

  Default value: N/A

  Range Value: `1` to `65535`

- `orb.ssl.port=`*`value`*

  Specifies the port number of the port that is used for secure IIOP connections. Valid values are `1-65535`.

  Type: Integer

  Default value: N/A

  Range Value: `1` to `65535`

- `osgi.shell.Telnet.port=`*`value`*

  Specifies the port number of the port that is used for connections to the Apache Felix Remote Shell (`http://felix.apache.org/site/apache-felix-remote-shell.html`). This shell uses the Felix shell service to interact with the OSGi module management subsystem. Valid values are `1-65535`.

  Type: Integer

  Default value: N/A

  Range Value: `1` to `65535`

Default value: N/A

The `--portbase` options cannot be used with the `--domainproperties` option.

`--keytooloptions` *`name=value`*`[:`*`name=value`*`]...`

Specifies an optional list of *name-value* pairs of keytool options for a self-signed server certificate. The certificate is generated while creating the domain. Each pair in the list must be separated by a colon (`:`).

Type: String

The following values can be specified:

- CN=*value*

  Specifies the common name of the host that is to be used for the self-signed certificate. This option name is case insensitive.

  Default value: Fully qualified name of the host where the `create-domain` subcommand is run.

  Range Value: N/A

  * CN property value is case insensitive.

Default value: N/A

`--savelogin={false|true}`

Saves the administration user name and password. The default value is `false`. The username and password are stored in the `.asadminpass` file of the user's home directory. A domain can only be created locally. Therefore, when using the `--savelogin` option, the host name saved in the `.asadminpass` property/entry is always localhost. If the user has specified default administration port while creating the domain, there is no need to specify `--user`, `--passwordfile`, `--host`, or `--port` in any of the subsequent `asadmin` remote commands. These values are obtained automatically.

When the same user creates multiple domains that have the same administration port number on the same or different host (where the home directory is NFS mounted), the subcommand does not ask if the password must be overwritten. The password is always overwritten.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--checkports={true|false}`

Specifies whether to check for the availability of the administration of the HTTP, JMS, JMX, and IIOP ports. The default value is `true`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--nopassword={false|true}`

Specifies whether the administrative user will have a password. If the value is set to `false` (default value), the password is specified using the `AS_ADMIN_PASSWORD` entry in the `asadmin password` file (set using the `--passwordfile` option). If the value is set to `false` and the `AS_ADMIN_PASSWORD` is not set, you are prompted for the password. If the value is set to `true`, the administrative user is created without a password. If a user name for the domain is not specified by using the `--user` option, and the `--nopassword` option is set to `true`, the default user name, `admin`, is used.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`domain-name`

Specifies the name of the domain to be created. The name can contain only ASCII characters and must be a valid directory name for the operating system on the host where the domain is created.

Type: String

The following values can be specified:

- *domain name*

Default value: N/A

Examples

The following example creates a domain.

```
asadmin create-domain --adminport 4848 domain4
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.4.4 create-service

Configures the starting of the DAS or server instance on an unattended boot on OS.

Synopsis

```
asadmin [asadmin-options] create-service [--help]
    [--serviceproperties service-properties] [--dry-run={false|true}]
    [--force={false|true}] [--serviceuser service-user]
    [--domaindir domain-dir] [--nodedir node-dir] [--node node]
    [domain-or-instance-name]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The create-service subcommand configures the starting of a Domain Administration Server (DAS) or a server instance on an unattended boot on the OS.

If no operand is specified and the domains directory contains only one domain, the subcommand configures the starting of the DAS for the default domain. If no operand is specified and the domains directory contains multiple domains, an error occurs.

If the operand specifies an instance, the create-service subcommand does not contact the domain administration server (DAS) to determine the node on which the instance resides. To determine the node on which the instance resides, the subcommand searches the directory that contains the node directories. If multiple node directories exist, the node must be specified as an option of the subcommand.

This subcommand is supported in local mode only

Execution permission

Superuser

Files

On UNIX systems, the create-service subcommand creates a System-V-style initialization script /etc/init.d/*domain-or-instance-name* and installs a link to this script in the /etc/rc?.d directories.

Arguments

--help | -?

Displays the help text for the subcommand.

--serviceproperties *service-properties*

Specifies a colon(**:**)-separated list of various properties that are specific to the service.

Type: String

The following values can be specified:

- *Name of the service property*

Default value: If you do not specify the DISPLAY_NAME property, then the display name of the service in the list of Windows Services is *Name of the instance or domain* Hitachi Java EE Server.

--dry-run={false|true} | -n

Previews your attempt to create a service. Indicates issues and the outcome that will occur if you run the command without using the --dry-run option.

Type: Boolean

The following values can be specified:

- true
- false

Default value: false

`--force={false|true}`

Specifies whether the service is created even if validation of the service fails.

Type: Boolean

The following values can be specified:

- `true`

  The service is created even if validation of the service fails.

- `false`

  The service is not created. This is the default value.

Default value: `false`

`--serviceuser` *`service-user`*

Specifies the user that is to run Application Server software when the service is started. Specify this option if Application Server software is to be run by a user other than the root user.

Type: String

The following values can be specified:

- *Name of the service user*

Default value: *The default is the user that is running the subcommand.*

`--domaindir` *`domain-dir`*

Specifies the absolute path of the directory on the disk that contains the configuration of the domain. If this option is specified, the operand must specify a domain.

Type: File

The following values can be specified:

- *Absolute path of the directory*

Default value: *`Application Server installation directory`*`/javaee/glassfish/domains`

`--nodedir` *`node-dir`*

Specifies the directory that contains the instance's node directory. If this option is specified, the *domain-or-instance-name* must specify a instance.

Type: String

The following values can be specified:

- *Instance nodes directory*

Default value: *`Application Server installation directory`*`/javaee/glassfish/nodes`

`--node` *`node`*

Specifies the node on which the instance resides. This option may be omitted only if the directory specified by the `--nodedir` option contains only one node directory. Otherwise, this option is required. If this option is specified, the *domain-or-instance-name* must specify an instance.

Type: String

The following values can be specified:

- *Name of the node*

Default value: N/A

`domain-or-instance-name`

Specifies the name of the domain or instance to configure. If no operand is specified, the default domain is used.

Type: String

The following values can be specified:

- *Name of the domain or instance*

Default value: N/A

Examples

The following example creates a service for the default domain.

```
asadmin create-service
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.4.5 delete-domain

Deletes a domain.

Synopsis

```
asadmin [asadmin-options] delete-domain [--help]
    [--domaindir domaindir] domain-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-domain` subcommand deletes the specified domain. The domain must already exist and must be stopped.

This subcommand is supported in local mode only.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--domaindir` *domaindir*

Specifies the directory where the domain to be deleted is located.

Type: String

The following values can be specified:

- *Name of the directory where the domain is located*

Default value: *Application Server installation directory*/javaee/glassfish/domains

domain-name

Specifies the unique name of the domain.

Type: String

The following values can be specified:

- *Name of the domain*

Default value: N/A

Examples

The following example deletes a domain named `mydomain4` from the default domains directory.

```
asadmin delete-domain mydomain4
```

Exit Status

Exit Status	Explanation
`0`	command executed successfully.
`1`	error in executing the command.

2.4.6 list-backups

Lists all the backups.

Synopsis

```
asadmin [asadmin-options] list-backups [--help]
    [--long[={false|true}]]
    [--domaindir domain-root-dir]
    [--backupdir backup-directory]
    [domain-name]
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `list-backups` subcommand displays information about the domain backups. This subcommand is supported in local mode only.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--long={false|true} | -l={false|true}`

Displays detailed information about each backup. The default value is `false`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--domaindir` *`domain-root-dir`*

Specifies the domain root directory and the parent directory of the domain upon which the command will operate.

The default value is *`Application Server installation directory`*`/javaee/glassfish/domains`.

Type: String

The following values can be specified:

- *Name of the directory*

Default value: *`Application Server installation directory`*`/javaee/glassfish/domains`

`--backupdir` *`backup-directory`*

Specifies the directory under which backup files are stored.

Type: String

The following values can be specified:

- *Name of the directory*

Default value: *`Application Server installation directory`*`/javaee/glassfish/domains/`*`domain-dir`*`/backups`

If the domain is not in the default location, the location is *`domain-dir`*`/backups`.

`domain-name`

Specifies the domain for which backups are listed.

This operand is optional if only one domain exists in Java EE Server installation.

Type: String

The following values can be specified:

- *Name of the domain*

Default value: N/A

Examples

The following example lists the backups of the default domain, `domain1`, that are stored in the `/net/backups.example.com/glassfish` directory.

```
asadmin list-backups --backupdir /net/backups.example.com/glassfish domain1
```

Exit Status

Exit Status	Explanation
`0`	subcommand executed successfully.
`1`	error in executing the subcommand.

2.4.7 list-domains

Lists the domains in the specified directory.

Synopsis

```
asadmin [asadmin-options] list-domains [--help]
    [--domaindir domaindir]
    [--long={false|true}]
    [--header={false|true}]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-domains` subcommand lists the domains in the specified domains directory. If the domains directory is not specified, the domains in the default directory are listed. This subcommand is supported in local mode only.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--domaindir` *domaindir*

Specifies the directory where the domains are to be listed. If specified, the path must be accessible in the files system. If not specified, the domains in the *Application Server installation directory*/javaee/glassfish/domains domain root directory are listed.

Type: String

The following values can be specified:

- *Name of the directory*

Default value: *Application Server installation directory*/javaee/glassfish/domains

`--long={false|true} | -l={false|true}`

Displays detailed information about the administration servers in the listed domains, including host names and port numbers.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--header={false|true} | -h={false|true}`

Specifies whether a header is displayed when the `--long` option is used.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

Examples

This example lists the domains in the default directory.

```
asadmin list-domains
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.4.8 restart-domain

Restarts the DAS of the specified domain.

Synopsis

```
asadmin [asadmin-options] restart-domain [--help]
    [--debug={true|false}]
    [--domaindir domaindir]
    [--force={true|false}] [--kill={false|true}]
    [domain-name]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `restart-domain` subcommand of `asadmin` stops and then restarts the DAS of the specified domain. If a domain is not specified, the default domain is assumed. If the `domains` directory contains two or more domains, the `domain-name` operand must be specified. If the DAS is not already running, the command attempts to start it.

This subcommand is supported in local or remote mode.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--debug={true|false}`

Specifies whether the domain is restarted with Java Platform Debugger Architecture (JPDA) debugging enabled. The default is the current setting of this option for the domain that is being restarted.

Type: Boolean

The following values can be specified:

- `true`

  The domain is restarted with JPDA debugging enabled and the port number for JPDA debugging is displayed.

- `false`

The domain is restarted with JPDA debugging disabled (default). The default is the current setting of this option for the domain that is being restarted.

Default value: `false`

`--domaindir` *`domaindir`*

Specifies the domain root directory, which contains the directory of the domain that is to be restarted.

Type: String

The following values can be specified:

- *Path of the domain root directory*

Default value: *`Application Server installation directory`*`/javaee/glassfish/domains`

`--force={true|false}`

Specifies whether the domain must be forcibly stopped just before it is restarted.

Type: Boolean

The following values can be specified:

- `true`

  The domain is forcibly stopped immediately. This is the default value.

- `false`

  The subcommand waits until all threads that are associated with the domain are exited before stopping the domain.

Default value: `true`

`--kill={false|true}`

Specifies whether the domain must be killed before it is restarted using the functionality of the operating system to terminate the domain process.

Type: Boolean

The following values can be specified:

- `true`

  The domain is killed. The subcommand uses functionality of the operating system to terminate the domain process.

- `false`

  The domain is not killed. The subcommand uses the functionality of the Java platform to terminate the domain process. This is the default value.

Default value: `false`

`domain-name`

Specifies the name of the domain you want to restart. Default is the name specified during installation, usually `domain1`.

Type: String

The following values can be specified:

- *Name of the domain*

Default value: `domain1`

Examples

The following example restarts `mydomain4` in the default `domains` directory.

```
asadmin restart-domain mydomain4
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.4.9 restore-domain

Restores files from the backup.

Synopsis

```
asadmin [asadmin-options] restore-domain [--help]
    [--long[={false|true}]] [--filename backup-filename]
    [--domaindir domain-root-dir] [--backupdir backup-directory]
    [--force[={false|true}]] [domain-name]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `restore-domain` subcommand of `asadmin` restores files under the domain from a backup directory. This subcommand is supported in local mode.

Additionally, the backup file that was backed up using a different operating system (OS), cannot be restored.

Precondition

The domain must be stopped.

Files

Files and subdirectories in a specified domain's directory.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--long[={false|true}] | -l`

Displays detailed information about the restore operation.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--filename` *`backup-filename`*

Specifies the name of the backup file to be used as the source.

Type: String

The following values can be specified:

- *File name*

Default value: N/A

`--domaindir` *`domain-root-dir`*

Specifies the domain root directory, the parent directory of the domain to restore.

Type: String

The following values can be specified:

- *Directory path*

Default value: *`Application Server installation directory`*`/javaee/glassfish/domains`

`--backupdir` *`backup-directory`*

Specifies the directory under which the backup file is stored.

Type: String

The following values can be specified:

- *Directory path*

Default value: *`Application Server installation directory`*`/javaee/glassfish/domains/domain-dir/backups`

If the domain is not in the default location, the location is `domain-dir/backups`.

`--force[={false|true}]`

Enables the restore operation to continue even when the name of the domain to restore does not match the name of the domain stored in the backup file.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`domain-name`

Specifies the name of the domain to restore. This operand is optional if only one domain exists in Java EE Server installation.

If the specified *domain name* does not match the domain name stored in the backup file, an error occurs unless the `--force` option is specified.

Type: String

The following values can be specified:

- *Name of the domain*

Default value: single domain available in Java EE Server installation

Examples

The following example restores files for the default domain, `domain1`, from the most recent backup stored in a specified backup directory.

```
asadmin restore-domain --backupdir /net/backups.example.com/glassfish domain1
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.4.10 start-domain

Starts the DAS of the specified domain.

Synopsis

```
asadmin [asadmin-options] start-domain
    [--help] [--debug={true|false}] [--domaindir domain-dir]
    [--dry-run={true|false}] [--upgrade={true|false}]
    [--verbose={true|false}] [--watchdog={true|false}] [domain-name]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `start-domain` subcommand of `asadmin` starts the DAS of the specified domain. If a domain is not specified, the default domain is assumed. If the `domains` directory contains two or more domains, the `domain-name` operand must be specified. This subcommand is supported in local mode only.

The `start-domain` subcommand prompts for a new admin user password if no password has been set for the admin user. Additionally, the admin user password must not be blank if secure administration is enabled; else, the `start-domain` subcommand fails.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`{--debug|-d}={true|false}`

Specifies whether the domain is started with Java Platform Debugger Architecture (JPDA) debugging enabled.

Type: Boolean

The following values can be specified:

- `true`

  The instance is started with JPDA debugging enabled and the port number for JPDA debugging is displayed.

- `false`

  The instance is started with JPDA debugging disabled. This is the default value.

Default value: `false`

`--domaindir` *`domain-dir`*

Specifies the domain root directory, which contains the directory of the domain that is to be restarted. If specified, the path must be accessible in the file system.

Type: String

The following values can be specified:

- *Path of the root directory of the domain*

Default value: *`Application Server installation directory`*`/javaee/glassfish/domains`

`{--dry-run|-n}={true|false}`

Suppresses the actual starting of the domain. Instead, `start-domain` displays the full Java command that would be used to start the domain, including all options. Reviewing this command can be useful to confirm JVM options and when troubleshooting startup issues.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--upgrade={true|false}`

Specifies whether the configuration of the domain administration server (DAS) is upgraded to the current release.

Type: Boolean

The following values can be specified:

- `true`

  When the domain is started, the configuration is modified to be compatible with this release of DAS, and DAS process stops.

- `false`

  The configuration of the DAS is not updated. This is the default value.

Default value: `false`

`{--verbose|-v}={true|false}`

Specifies whether detailed information about the domain is displayed in the console window where the subcommand is run.

Type: Boolean

The following values can be specified:

- `true`

  Detailed startup messages and log messages about the domain are displayed in the console window where the subcommand is run. If the domain is later restarted by running the `restart-domain` subcommand from a different console window, messages continue to be displayed in the original console window. You can kill Java EE Server process by typing **CTRL**+**C** in the console window. You can kill Java EE Server process and obtain a thread dump for the server by typing one of the following key combinations in the console window:

  CTRL+\ on UNIX systems

- `false`

  Detailed information is not displayed. This is the default value.

Default value: `false`

`{--watchdog|-w}={true|false}`

Specifies whether limited information about the domain is displayed in the console window where the subcommand is run.

The `--watchdog` option is similar to `--verbose` but does not display the detailed startup messages and log messages. This option is useful when running the `asadmin` utility in the background or with no attached console.

Type: Boolean

The following values can be specified:

- `true`

  Limited information is displayed in the console window.

- `false`

  Limited information is not displayed in the console window. This is the default value.

Default value: `false`

`domain-name`

Specifies the unique name of the domain you want to start.

Type: String

The following values can be specified:

- *Name of the domain*

Default value: N/A

Examples

The following example starts `mydomain4` in the default `domains` directory.

```
asadmin start-domain mydomain4
```

Exit Status

Exit Status	Explanation
`0`	subcommand executed successfully.
`1`	error in executing the subcommand.

2.4.11 stop-domain

Stops the Domain Administration Server of the specified domain.

Synopsis

```
asadmin [asadmin-options] stop-domain [--help]
    [--domaindir domaindir]
    [--kill={false|true}] [domain-name]
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `stop-domain` subcommand of `asadmin` stops the Domain Administration Server (DAS) of the specified domain. If the domain directory is not specified, the domain in the default `domains` directory is stopped. If there are two or more domains in the `domains` directory, the *`domain-name`* operand must be specified.

This subcommand is supported in local or remote mode.

When stopping a server, the message `KDKD45000-E` may be stored in the standard error and `logger.log` files. However, the termination processing is not affected.

If you specify a host name, the subcommand assumes you are operating in remote mode, which means you must correctly authenticate to the remote server. In local mode, you normally do not need to authenticate to the server as long as you are running the subcommand as the same user who started the server.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--domaindir={true|false}`

Specifies the directory of the domain that is to be stopped. If specified, the path must be accessible in the file system. If not specified, the domain in the default *`Application Server installation directory`*`/javaee/glassfish/domains` directory is stopped.

Type: String

The following values can be specified:

- *Path of the root directory of the domain*

Default value: *`Application Server installation directory`*`/javaee/glassfish/domains`

`--kill={false|true}`

Specifies whether the domain is killed by using the functionality of the operating system to terminate the domain process.

Type: Boolean

The following values can be specified:

- `true`

  The domain is killed. The subcommand uses functionality of the operating system to terminate the domain process.

- `false`

  The domain is not killed. The subcommand uses functionality of the Java platform to terminate the domain process. This is the default value.

Default value: `false`

`domain-name`

Specifies the name of the domain you want to stop.

Type: String

The following values can be specified:

- *Name of the domain*

Default value: `domain1`

Examples

The following example stops the domain named `sampleDomain` in the default `domains` directory.

```
asadmin stop-domain sampleDomain
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.5 Commands used for node administration

This section describes the syntax and functionality of the commands used for node administration.

2.5.1 create-node-config

Creates a node that is not enabled for remote communication.

Synopsis

```
asadmin [asadmin-options] create-node-config [--help]
    [--nodehost node-host] [--installdir as-install-parent]
    [--nodedir node-dir] node-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-node-config` subcommand creates a node that is not enabled for remote communication.

This command does not require the secure shell (SSH) to be configured to create the node.

A node represents a host on which Application Server software is installed. A node must exist for every host on which server instances reside. All administration of instances on a node that is not enabled for remote communication must be performed on the host that the node represents. The domain administration server (DAS) on a remote host cannot contact the node. This command is supported in remote mode only.

Precondition

DAS has to be in a running state.

A node must exist for every host on which server instances reside.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--nodehost` *node-host*

Specifies the name of the host that the node represents. If this option is omitted, no host is specified for the node.

Type: String

The following values can be specified:

- *Specify the name of the host*

Default value: N/A

`--installdir` *as-install-parent*

Specifies the full path to *Application Server installation directory*/javaee. If this option is omitted, no *Application Server installation directory*/javaee is specified for the node.

Type: String

The following values can be specified:

- The full path to *Application Server installation directory*/javaee

Default value: N/A

--nodedir *node-dir*

Specifies the path to the directory that is to contain server instances that are created on the node. If a relative path is specified, the path is relative to the *Application Server installation directory* directory. If this option is omitted, no directory for instances is specified for the node.

Type: String

The following values can be specified:

- *File path of a directory where the node resides*

Default value: N/A

node-name

Specifies the name of the node.

The name must meet the following requirements:

The name may contain only ASCII characters.

The name must start with a letter, a number, or an underscore.

The name may contain only the following characters:

- Lowercase letters
- Uppercase letters
- Numbers
- Hyphen
- Period
- Underscore

The name must be unique in the domain and must not be the name of another node, a cluster, a named configuration, or a server instance.

The name must not be domain, server, or any other keyword that is reserved by Java EE Server.

Type: String

The following values can be specified:

- *Specify the node name*

Default value: N/A

Examples

The following example creates a node config:

```
asadmin create-node-config--nodehost sj03.example.com sj03
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.5.2 create-node-ssh

Creates a node that is enabled for communication over SSH.

Synopsis

```
asadmin [asadmin-options] create-node-ssh [--help]
    --nodehost node-host [--installdir as-install-parent]
    [--nodedir node-dir] [--sshport ssh-port] [--sshuser ssh-user]
    [--sshkeyfile ssh-keyfile] [--force={false|true}]
    node-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-node-ssh` subcommand creates a node that is enabled for communication over the secure shell (SSH) protocol. A node represents a host on which Application Server software is installed. A node must exist for every host on which server instances reside. The domain administration server (DAS) contacts an SSH node's host through the SSH connector to manage server instances that reside on the node. However, the DAS does not use the SSH connector to contact the host where the DAS is running because the DAS can run all `asadmin` subcommands locally. This command is supported in remote mode only.

To force the node to be created in the DAS configuration even if the host cannot be contacted through SSH, set the `--force` option to `true`.

Precondition

Domain Administration Server (DAS) is running.

A node must exist for every host on which server instances reside

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--nodehost` *node-host*

Specifies the name of the host that the node represents. The name of the host must be specified. Otherwise, an error occurs.

Type: String

The following values can be specified:

- *Specify the nodehost name*

Default value: N/A

`--installdir` *as-install-parent*

Specifies the full path to *Application Server installation directory*/javaee.

Type: String

The following values can be specified:

- The full path to *Application Server installation directory*/javaee

Default value: *Application Server installation directory*/javaee directory for the DAS.

--nodedir *node-dir*

Specifies the path to the directory that is to contain server instances that are created on the node. If a relative path is specified, the path is relative to *Application Server installation directory* directory.

Type: String

The following values can be specified:

- *Specify the path to the directory that contain server instance*

Default value: *Application Server installation directory*/javaee/glassfish/nodes

--sshport *ssh-port*

Specifies the port to use for SSH connections to this node's host. If the --nodehost option is set to localhost-domain, the --sshport option is ignored.

Type: Integer

The following values can be specified:

- 1 to 65535

Default value: 22

--sshuser *ssh-user*

Specifies the user on this node's host that is to run the process for connecting to the host through the SSH protocol. To ensure that the DAS can read this user's SSH private key file, specify the user that is running the DAS process. If the --nodehost option is set to localhost domain, the --sshuser option is ignored.

Type: String

The following values can be specified:

- *Specify the sshuser name*

Default value: User that is running the DAS process

--sshkeyfile *ssh-keyfile*

Specifies the absolute path to the SSH private key file for user that the --sshuser option specifies. This file is used for authentication to the SSHd daemon on the node's host.

Type: String

The following values can be specified:

- *Path to the SSH private key file for user*

Default value:

Key file in the user's .ssh directory on the host where the subcommand is run. If multiple key files are found in user's .ssh directory, the subcommand uses the following order of preference:

1. id_rsa
2. id_dsa
3. identity

--force={false|true}

Specifies whether the node is created in the DAS configuration even if validation of the node's parameters fails.

To validate a node's parameters, the DAS must be able to contact the node's host through SSH.

Type: Boolean

The following values can be specified:

- false

The node is not created if validation of the node's parameters fails (default).

- `true`

  The node is created even if validation of the node's parameters fails.

Default value: `false`

`node-name`

Specifies the name of the node.

The name must meet the following requirements:

The name may contain only ASCII characters.

The name must start with a letter, a number, or an underscore.

The name may contain only the following characters:

- Lowercase letters
- Uppercase letters
- Numbers
- Hyphen
- Period
- Underscore

The name must be unique in the domain and must not be the name of another node, a cluster, a named configuration, or a server instance.

The name must not be domain, server, or any other keyword that is reserved by Java EE Server.

Type: String

The following values can be specified:

- *Specify the node name*

Default value: N/A

Examples

The following example creates a node.

```
asadmin create-node-ssh --nodehost adc.example.com
--installdir /export/glassfish3 adc
```

Exit Status

Exit Status	Explanation
`0`	command executed successfully.
`1`	error in executing the command.

2.5.3 delete-node-config

Deletes a node that is not enabled for remote communication.

Synopsis

```
asadmin [asadmin-options] delete-node-config [--help]
    node-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-node-config` subcommand deletes a node that is not enabled for remote communication from the domain.

This subcommand does not require the secure shell (SSH) to be configured.

This subcommand can delete only a node that is not enabled for remote communication. No server instances must reside on the node that is being deleted. Otherwise, the subcommand fails. Before running this subcommand, delete any instances that reside on the node.

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

node-name

Specifies the name of the node.

The node must not be enabled for communication over SSH. Otherwise, an error occurs.

Type: String

The following values can be specified:

- *Name of the node*

Default value: N/A

Examples

The following example deletes the node `sj03`, which is not enabled for remote communication.

```
asadmin delete-node-config sj03
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.5.4 delete-node-ssh

Deletes a node that is enabled for communication over SSH.

Synopsis

```
asadmin [asadmin-options] delete-node-ssh [--help]
    node-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-node-ssh` subcommand deletes a node that is enabled for communication over secure shell (SSH) from the domain. This subcommand does not require SSH to be configured.

This subcommand can delete only a node that is enabled for communication over SSH.

No server instances must reside on the node that is being deleted. Otherwise, the subcommand fails. Before running this subcommand, delete any instances that reside on the node.

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

node-name

Specifies the name of the node.

Type: String

The following values can be specified:

- *Name of the node*

Default value: N/A

Examples

The following example deletes the node `eg1`, which is enabled for communication over SSH.

```
asadmin delete-node-ssh eg1
```

Exit Status

Exit Status	Explanation
0	command executed successfully.

Exit Status	Explanation
1	error in executing the command.

2.5.5 list-nodes

Lists all Java EE Server nodes in a domain.

Synopsis

```
asadmin [asadmin-options] list-nodes [--help]
    [--long={false|true}]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-nodes` subcommand of asadmin lists all Java EE Server nodes in a domain.

By default, the command displays the following information for each node that is listed:

- The name of the node.
- The type of the node, which is one of the following types:
  `CONFIG`
  The node does not support remote communication.
  `SSH`
  The node supports communication over secure shell (SSH).
- The name of the host that the node represents.

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`
Displays the help text for the subcommand.

`--long={false|true} | -l={false|true}`
Specifies whether the nodes are listed in long format.
Type: Boolean
The following values can be specified:

- `true`
  The nodes are listed in long format.
- `false`

The nodes are listed in short format. This is the default value.

Default value: `false`

Examples

The following example lists Java EE Server nodes in long format.

```
asadmin list-nodes --long=true
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.5.6 list-nodes-config

Lists all Java EE Server nodes that do not support remote communication in a domain.

Synopsis

```
asadmin [asadmin-options] list-nodes-config [--help]
    [--long={false|true}]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-nodes-config` subcommand of asadmin lists all Java EE Server nodes that do not support remote communication in a domain.

By default, the command displays the following information for each listed node:

- The name of the node.
- The type of the node (`CONFIG`).
- The name of the host that the node represents.

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--long={false|true} | -l={false|true}`

Specifies whether the nodes are listed in long format.

Type: Boolean

The following values can be specified:

- `true`

  The nodes are listed in long format.

- `false`

  The nodes are listed in short format. This is the default value.

Default value: `false`

Examples

The following example lists Java EE Server nodes that do not support remote communication in the domain `domain1` in long format.

```
asadmin list-nodes-config --long=true
```

Exit Status

Exit Status	Explanation
`0`	subcommand executed successfully.
`1`	error in executing the subcommand.

2.5.7 list-nodes-ssh

Lists all Java EE Server nodes that support communication over SSH in a domain.

Synopsis

```
asadmin [asadmin-options] list-nodes-ssh [--help]
    [--long={false|true}]
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `list-nodes-ssh` subcommand of asadmin lists all Java EE Server nodes that support communication over a secure shell (SSH) in a domain.

By default, the command displays the following information for each node that is listed:

- The name of the node.
- The type of the node (`SSH`).
- The name of the host that the node represents.

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--long={false|true} | -l={false|true}`

Specifies whether the nodes are listed in long format.

Type: Boolean

The following values can be specified:

- `true`
  The nodes are listed in long format.
- `false`
  The nodes are listed in short format. This is the default value.

Default value: `false`

Examples

The following example displays the name, type, and host of all Java EE Server nodes that support communication over SSH in a domain.

```
asadmin list-nodes-ssh
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.5.8 ping-node-ssh

Tests nodes enabled for communication over SSH to determine if they are usable.

Synopsis

```
asadmin [asadmin-options] ping-node-ssh [--help]
    [--validate={false|true}] node-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `ping-node-ssh` subcommand of `asadmin` validates the node to determine if the `asadmin` utility can run on the host which the node represents. This subcommand is supported in remote mode only.

Precondition

- Domain Administration Server (DAS) is running.
- This command requires secure shell (SSH) to be configured on the machine where the domain administration server (DAS) is running, and on the machine where the node resides.
- The node must be enabled for communication over SSH. Otherwise, an error occurs.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--validate={false|true}`

Specifies whether the subcommand validates the node.

Type: Boolean

The following values can be specified:

- `true`

  The node is validated.

- `false`

  The node is not validated. This is the default value.

Default value: `false`

`node_name`

Specifies the name of the node to test.

Type: String

The following values can be specified:

- *Name of the node*

Default value: N/A

Examples

The following example tests if the SSH-enabled node `sj03-node` is usable.

```
asadmin ping-node-ssh sj03-node
```

Exit Status

Exit Status	Explanation
`0`	subcommand executed successfully.
`1`	error in executing the subcommand.

2.5.9 setup-ssh

Sets up an SSH key on specified hosts.

Synopsis

```
asadmin [asadmin-options] setup-ssh [--help]
    [--sshport ssh-port] [--sshuser ssh-user]
    [--sshkeyfile ssh-keyfile] [--sshpublickeyfile ssh-public-keyfile]
    [--generatekey={false|true}] host-list
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `setup-ssh` subcommand of `asadmin` sets up a secure shell (SSH) key on the hosts that are specified as the operand of the subcommand. This key enables Java EE Server to use public-key authentication for authentication of the user's SSH login on remote hosts. SSH ensures that Java EE Server clusters that span multiple hosts can be administered centrally. SSH provides confidentiality and security for data that is exchanged between the DAS and remote hosts.

Public-key authentication uses an SSH key pair that comprises the following keys:

- A private key, which is stored in a secure location on the DAS host and which may be protected with a passphrase.
- The public key, which is stored on all the remote hosts with which the DAS communicates.

The subcommand does not require any configuration information from the DAS and does not modify the configuration of the DAS. This subcommand is supported in local mode only.

Precondition

- The ssh client is installed on the DAS host and is accessible through the DAS user's path.
- The sshd daemon is installed and running on all hosts where an SSH key is to be set up.
- The `ssh-keygen` utility is installed on the DAS host either at the default location or in a location that is defined in the DAS user's path.
- The user specified in `--sshuser` has an SSH login on all hosts where an SSH key is to be set up.

Files

Few or all the file namely:

- `authorized_keys`
- `id_rsa`
- `id_rsa.pub`
- `id_dsa`
- `id_dsa.pub`
- `identity`
- `identity.pub`

files at *user-home*/.ssh/ of both DAS host and the remote hosts are affected.

Arguments

--help | -?

Displays the help text for the subcommand.

--sshport *ssh-port*

Specifies the port to use for SSH connections to the host where SSH is being set up.

Type: Integer

The following values can be specified:

- 1 to 65535

Default value: 22

--sshuser *ssh-user*

Specifies the SSH user on the remote host, which is to run the process for setting up SSH on that host.

To ensure that the DAS can read this user's SSH private key file, specify the user, which is running the DAS process.

Type: String

The following values can be specified:

- *Name of the user*

Default value: User running the command.

--sshkeyfile *ssh-keyfile*

Specifies the absolute path to the SSH private key file for the user that the --sshuser option specifies. This file is used for authentication to the sshd daemon on the host. The user that is running this subcommand must be able to reach the path to the key file and read the key file.

Type: String

The following values can be specified:

- *Path to the SSH private key file for user*

Default value: Key file in the user's .ssh directory on the host where the subcommand is run. If multiple key files are found in user's .ssh directory, the subcommand uses the following order of preference:

- id_rsa
- id_dsa
- identity

--sshpublickeyfile *ssh-public-keyfile*]

Specifies the absolute path to the SSH public key file for the user that the --sshuser option specifies. The content of the public key file is appended to the user's .ssh/authorized_keys file on each host where SSH is being set up.

If the .ssh/authorized_keys file does not exist on a host, the subcommand creates the file. The user that is running this subcommand must be able to reach the path to the key file and read the key file.

Type: String

The following values can be specified:

- *Path to the SSH public key file for user*

Default value: Key file in the user's .ssh directory on the host where the subcommand is run. If multiple key files are found in user's .ssh directory, the subcommand uses the following order of preference:

- `id_rsa.pub`
- `id_dsa.pub`
- `identity.pub`

`--generatekey={false|true}`

Specifies whether the subcommand generates the SSH key files without prompting the user.

Type: Boolean

The following values can be specified:

- `true`

  The subcommand generates the SSH key files without prompting the user.

- `false`

  The behavior of the subcommand depends on whether the SSH key files exist:

  If the SSH key files exist, the subcommand does not generate the files.

  If the SSH key files do not exist, the behavior of the subcommand depends on the value of the `--interactive` option of the `asadmin` utility.

  If the `--interactive` option is `true`, the subcommand prompts the user to create the files.

  If the `--interactive` option is `false`, the subcommand fails.

  This is the default value.

Default value: `false`

`host-list`

Generates a space-separated list of the names of the hosts where an SSH key is to be set up.

Type: String

The following values can be specified:

- *Name of the remote hosts*

Default value: N/A

Examples

The following example sets up an SSH key for the user `gfuser` on the hosts `sj03` and `sj04`. The key file is not generated but is copied from the user's `.ssh` directory on the host where the subcommand is running.

```
asadmin setup-ssh sj03 sj04
```

Exit Status

Exit Status	Explanation
`0`	command executed successfully.
`1`	error in executing the command.

2.5.10 update-node-config

Updates the configuration data of a node.

Synopsis

```
asadmin [asadmin-options] update-node-config [--help]
    [--nodehost node-host] [--installdir as-install-parent]
    [--nodedir node-dir] node-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The update-node-config subcommand of asadmin updates the configuration data of a node. This subcommand can update any node, regardless of whether the node is enabled for remote communication. If a node that is enabled for remote communication is updated, the node is not enabled for remote communication after the update.

This subcommand does not require the secure shell (SSH) to be configured to update the node. You may run this subcommand from any host that can contact the DAS.

This subcommand is supported in remote mode only.

Precondition

- Node should be existing.
- Domain Administration Server (DAS) is running.

Arguments

--help | -?

Displays the help text for the subcommand.

--nodehost *node-host*

Specifies the name of the host that the node is to represent after the node is updated.

Type: String

The following values can be specified:

- *Host name*

Default value: N/A

--installdir *as-install-parent*

Specifies the full path to *Application Server installation directory*/javaee.

Type: String

The following values can be specified:

- The full path to *Application Server installation directory*/javaee

Default value: N/A

--nodedir *node-dir*

Specifies the path to the directory that is to contain server instances that are created on the node. If a relative path is specified, the path is relative to the *Application Server installation directory*/javaee/glassfish directory, where *Application Server installation directory*/javaee/glassfish is the base installation directory of Application Server software on the host.

Type: String

The following values can be specified:

- *Path to the directory containing the server instances of the node*

Default value: N/A

node-name

Specifies the name of the node to update.

Type: String

The following values can be specified:

- *Name of the node*

Default value: N/A

Examples

The following example updates the host that the node `sj04` represents to `hsj04`.

```
asadmin update-node-config --nodehost hsj04 sj04
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.5.11 update-node-ssh

Updates the configuration data of a node.

Synopsis

```
asadmin [asadmin-options] update-node-ssh [--help]
    [--nodehost node-host] [--installdir as-install-parent]
    [--nodedir node-dir] [--sshport ssh-port]
    [--sshuser ssh-user] [--sshkeyfile ssh-keyfile]
    [--force={false|true}] node-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `update-node-ssh` subcommand of `asadmin` updates the configuration data of a node. This subcommand requires secure shell (SSH) to be configured on the machine where the domain administration server (DAS) is running and on the machine where the node resides. You may run this subcommand from any machine that can contact the DAS. This subcommand can update any node, regardless of whether the node is enabled for remote communication. If the node is not enabled for remote communication, the subcommand enables SSH communication for the node and updates any other specified configuration data. If this subcommand is run to enable SSH communication for a node, default values are applied if `--sshport`, `--sshuser` or `--sshkeyfile` is omitted. This subcommand is supported in remote mode only.

Precondition

- SSH is configured on the host where the DAS is running and on the host that the node represents.
- The node that you are updating exists.
- Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--nodehost` *`node-host`*

Specifies the name of the host that the node is to represent after the node is updated.

Type: String

The following values can be specified:

- *Host name*

Default value: N/A

`--installdir` *`as-install-parent`*

Specifies the full path to *`Application Server installation directory`*`/javaee`.

Type: String

The following values can be specified:

- The full path to *`Application Server installation directory`*`/javaee`

Default value: N/A

`--nodedir` *`node-dir`*

Specifies the path to the directory that is to contain server instances that are created on the node. If a relative path is specified, the path is relative to the *`Application Server installation directory`*`/javaee/glassfish` directory, where *`Application Server installation directory`*`/javaee/glassfish` is the base installation directory of Application Server software on the host.

Type: String

The following values can be specified:

- *Path to the directory containing the server instances of the node*

Default value: N/A

`--sshport` *`ssh-port`*

Specifies the port to use for SSH connections to this node's host.

The default depends on whether this subcommand is run to enable SSH communication for the node:

- If the node is already enabled for communication over SSH, the default is to leave the port unchanged.
- If this subcommand is run to enable SSH communication for the node, the default port is `22`. If the `--nodehost` is set to `localhost`, the `--sshport` option is ignored.

Type: Integer

The following values can be specified:

- `1` to `65535`

Default value: `22`

`--sshuser` *`ssh-user`*

Specifies the user on this node's host that is to run the process for connecting to the host, through SSH.

The default depends on whether this subcommand is run to enable SSH communication for the node:

- If the node is already enabled for communication over SSH, the default is to leave the user unchanged.
- If this subcommand is run to enable SSH communication for the node, the default is the user that is running the DAS process. If the `--nodehost` option is set to `localhost`, the `--sshuser` option is ignored.

Type: String

The following values can be specified:

- *User name*

Default value: *User running the DAS process*

`--sshkeyfile` *`ssh-keyfile`*

Specifies the absolute path to the SSH private key file for the user that the `--sshuser` option specifies. This file is used for authentication to the `sshd` daemon on the node's host.

If the SSH private key file is protected by a passphrase, the password file must contain the `AS_ADMIN_SSHKEYPASSPHRASE` entry. The path to the key file must be reachable by the DAS and the key file must be readable by the DAS. The default depends on whether this subcommand is run to enable SSH communication for the node:

- If the node is already enabled for communication over SSH, the default is to leave the key file unchanged.
- If this subcommand is run to enable SSH communication for the node, the default is the key file in the user's `.ssh` directory. If multiple key files are found, the subcommand uses the following order of `id_rsa`, `id_dsa`, `identity`.

Type: String

The following values can be specified:

- *Path to the directory containing SSH private key file for user*

Default value: Key file in the user's `.ssh` directory if subcommand is run to enable SSH communication.

`--force={false|true}`

Specifies whether the node is updated even if validation of the node's parameters fails. To validate a node's parameters, the DAS must be able to contact the node's host through SSH.

Type: Boolean

The following values can be specified:

- `false`

  The node is not updated if validation of the node's parameters fails. This is the default value.

- `true`

  The node is updated even if validation of the node's parameters fails.

Default value: `false`

`node-name`

Specifies the name of the node to update.

Type: String

The following values can be specified:

- *Node name*

Default value: N/A

Examples

The following example updates the host that the node `lssh` represents to `sj04`.

```
asadmin update-node-ssh --nodehost sj04 lssh
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.6 Commands used for server administration

This section describes the syntax and functionality of the commands used for server administration.

2.6.1 cleanup-prf

The `cleanup-prf` subcommand cleans up the PRF server environment.

Synopsis

```
asadmin [asadmin-options] cleanup-prf [--nodedir node-dir]
    [--node node-name] prf-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `cleanup-prf` subcommand stops the PRF servers that were created on the host running the subcommand, and removes (cleans up) the PRF environment. However, the subcommand does not remove the PRF information from the domain.

Use this command for the PRF servers that are not managed by the domain administration server in situations where the PRF environment remains on the remote node because a domain was recreated when the domain administration server stopped.

This subcommand is supported only in the local mode.

Execution permission

General user

Precondition

The domain administration server does not manage the PRF server of the target node.

Arguments

`--nodedir` *node_dir*

Specifies the node directory where the PRF server was created.

Type: String

You can specify the following values:

- Path of the existing node directory.

Default value: *Application Server installation directory*/nodes

`--node` *node_name*

Specifies the name of a node where a PRF server has been created.

Type: String

You can specify the following values:

- Existing node name.

Default value: N/A

This option can be omitted if only one node directory exists in the directory specified by the `--nodedir` option.

prf_name

Specifies the name of the PRF server to be deleted.

Type: String

You can specify the following values:

- Name of the server where the PRF server exists.

Default value: (None. You must specify a value.)

Examples

The following example removes the PRF environment named `PRF1`, which belongs to the `localhost-domain1` node, and which was built in the `/temp/nodes` directory.

```
asadmin cleanup-prf --nodedir /temp/nodes --node localhost-domain1 PRF1
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

Notes

- Do not use this command in an environment where the `stop-prf` subcommand can stop the PRF server. If you do use this command in such an environment, the domain administration server might detect a PRF server failure.
- Do not use this command in cases where the PRF server can be removed from the domain administration server. If you do use this command in such a situation and therefore PRF information remains in the domain, remove the PRF information by using the `--force` option of the `delete-prf` subcommand. Also, if you execute `start-prf` when PRF information remains in the domain, PRF information might be created on the node.

2.6.2 cleanup-webserver

The `cleanup-webserver` subcommand cleans up the web server environment.

Synopsis

```
asadmin [asadmin_options] cleanup-webserver [--graceful={false|true}]
    [--nodedir node_dir] [--node node_name] webserver_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `cleanup-webserver` subcommand stops a web server that was set up on the host running the subcommand. This subcommand removes (cleans up) the web server environment. However, the subcommand does not remove the web server information from the domain.

Use this command for web servers that are not managed by the domain administration server in situations such as where the web environment remains on the remote node because a domain was recreated when the domain administration server stopped.

This subcommand is supported only in the local mode.

Execution permission

General user

Precondition

The web server of the target node is not managed by the domain administration server.

Arguments

`--graceful={false|true}`

Specifies whether to stop the web server in a planned termination. This value is ignored if the process is already stopped.

Type: Boolean

You can specify the following values:

- `true`

  Stops the web server in a planned termination.

- `false`

  Stops the web server normally.

Default value: `true`

`--nodedir` *`node_dir`*

Specifies the node directory where the web server is created.

Type: String

You can specify the following values:

- Path of the existing node directory

Default value: *`Application Server installation directory`*`/nodes`

`--node` *`node_name`*

Specifies the node name where the web server must be configured.

Type: String

You can specify the following values:

- Existing node name

Default value: N/A

This option can be omitted if only one node directory exists in the directory specified by the `--nodedir` option.

`webserver_name`

Specifies the name of the web server to be deleted.

Type: String

You can specify the following values:

- Name of the server where the web server exists.

Default value: (None. You must specify a value.)

Examples

The following example cleans up a web server that has the name `Web1`:

```
asadmin cleanup-webserver --nodedir /temp/nodes --node localhost-domain1 Web1
```

The following example cleans up a web server that has the name `Web1`, which belongs to the `localhost-domain1` domain specified in the `/temp/nodes` node:

```
asadmin cleanup-webserver --nodedir /temp/nodes --node localhost-domain1 Web1
```

The following example cleans up a web server that has the name `Web1` after a planned termination:

```
asadmin cleanup-webserver --graceful true --nodedir /temp/nodes
--node localhost-domain1 Web1
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

Notes

- Do not use this command in an environment where the `stop-webserver` subcommand can stop the web server. If you do use this command in such an environment, the domain administration server might detect a web server failure.
- Do not use this command in cases where the web server can be removed from the domain administration server. If you used this command in such a situation and therefore web server information remains in the domain, remove the web server information by using the `--force` option of the `delete-webserver` subcommand. Also, if you execute `start-webserver` when web server information remains in the domain, web server information might be created on the node.

2.6.3 copy-config

Copies an existing named configuration to create another configuration.

Synopsis

```
asadmin [asadmin-options] copy-config [--help]
    [--systemproperties name=value[:name=value]...]
    source-configuration-name destination-configuration-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The copy-config subcommand of asadmin creates a configuration in the DAS by copying an existing configuration.

The new configuration is identical to the copied configuration, except for any properties that are specified in the --systemproperties option. The default-config configuration is copied when either a standalone sever instance or a standalone cluster is created.

This command is supported in remote mode only.

Precondition

- The DAS whose configuration is to copied must be running.
- The source configuration must exist before executing the copy-config subcommand.

Arguments

--help | -?

Displays the help text for the subcommand.

--systemproperties *name*=*value*[:*name*=*value*]...

Displays the optional attribute name-value pairs for the configuration. These properties override the port settings in the configuration. The specification format is name=*value*. Use a colon (:) to separate property names. If you specify the same property name more than once, the property value specified last is valid.

On UNIX, creating sockets that listen on ports in the range of 1 to 1024 requires superuser privileges.

Type: String

The following values can be specified:

- ASADMIN_LISTENER_PORT=*value*

  This property specifies the port number of the HTTP or HTTPS ports through which the DAS connects to the instance to manage the instance. The valid values are in the range of 1 - 65535.

  Type: Integer

  Default value: N/A

  Range Value: 1 to 65535

- HTTP_LISTENER_PORT=*value*

  This property specifies the port number of the port that is used to listen for HTTP requests. The valid values are in the range of 1 - 65535.

  Type: Integer

  Default value: N/A

  Range Value: 1 to 65535

- HTTP_SSL_LISTENER_PORT=*value*

  This property specifies the port number of the port that is used to listen for HTTPS requests. The valid values are in the range of 1 - 65535.

  Type: Integer

  Default value: N/A

Range Value: 1 to 65535

- IIOP_LISTENER_PORT=*value*

  This property specifies the port number of the port that is used for IIOP connections. The valid values are in the range of 1 - 65535.

  Type: Integer

  Default value: N/A

  Range Value: 1 to 65535

- IIOP_SSL_LISTENER_PORT=*value*

  This property specifies the port number of the port that is used for secure IIOP connections. The valid values are in the range of 1 - 65535.

  Type: Integer

  Default value: N/A

  Range Value: 1 to 65535

- IIOP_SSL_MUTUALAUTH_PORT=*value*

  This property specifies the port number of the port that is used for secure IIOP connections with client authentication. The valid values are in the range of 1 - 65535.

  Type: Integer

  Default value: N/A

  Range Value: 1 to 65535

- JAVA_DEBUGGER_PORT=*value*

  This property specifies the port number of the port that is used for connections to the Java Platform Debugger Architecture (JPDA) http://java.sun.com/javase/technologies/core/toolsapis/jpda/ debugger. The valid values are in the range of 1 - 65535.

  Type: Integer

  Default value: N/A

  Range Value: 1 to 65535

- JMS_PROVIDER_PORT=*value*

  This property specifies the port number for the Java Message Service provider. The valid values are in the range of 1 - 65535.

  Type: Integer

  Default value: N/A

  Range Value: 1 to 65535

- JMX_SYSTEM_CONNECTOR_PORT=*value*

  This property specifies the port number on which the JMX connector listens. The valid values are in the range of 1 to 65535.

  Type: Integer

  Default value: N/A

  Range Value: 1 to 65535

Default value: N/A

source-configuration-name

Displays the name of the configuration that you are copying.

Type: String

The following values can be specified:

- *Name of a source configuration to be copied*

Default value: N/A

destination-configuration-name

Displays the name of the configuration that you are creating by copying the source configuration.

The value for destination-configuration-name must meet the following conditions:

The name may contain only ASCII characters.

The name must start with a letter, a number, or an underscore.

The name may contain only the following characters:

- Lowercase letters
- Uppercase letters
- Numbers
- Hyphen
- Period
- Underscore

The name must be unique in the domain and must not be the name of another node, a cluster, a named configuration, or a server instance.

The name must not be domain, server, or any other keyword that is reserved by Java EE Server.

Default value: None

Type: String

The following values can be specified:

- *Name of a destination configuration to be created*

Examples

The following example creates the named configuration `clusterpresets-config` by copying the default configuration.

```
asadmin copy-config default-config clusterpresets-config
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.6.4 create-cluster

Creates Java EE Server cluster.

Synopsis

```
asadmin [asadmin-options] create-cluster [--help]
    [--config config-name]
    [--systemproperties name=value[:name=value]...]
```

```
    [--properties name=value[:name=value]...]
    [--gmsenabled={true|false}] [--multicastport multicast-port]
    [--multicastaddress multicast-address]
    [--bindaddress bind-address] cluster-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-cluster` subcommand creates Java EE Server cluster. A cluster requires a reference to the named configuration that defines the configuration of all instances that are added to that cluster.

The cluster that is created is a standalone cluster because the cluster's configuration is not shared with any other clusters or standalone instances.

This command is supported in remote mode only.

Execution permission

On UNIX, creating sockets that listen on ports `1` to `1024` requires superuser privileges.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--config` *config-name*

Specifies the named configuration that the cluster references. Specifying the `--config` option creates a shared cluster. If this option is omitted, a standalone cluster is created.

Type: String

The following values can be specified:

- *Specify the configuration name*

Default value: N/A

`--systemproperties` *name*=*value*[:*name*=*value*]...

Defines system properties for the configuration that is created for the cluster. These properties override the property values in the `default-config` configuration. The specification format is `name=`*value*. Use a colon (`:`) to separate property names. If you specify the same property name more than once, the property value specified last is valid.

On UNIX, creating sockets that listen on ports `1` to `1024` requires superuser privileges.

Type: String

The following values can be specified:

- `ASADMIN_LISTENER_PORT=`*value*

  Specifies the port number of the HTTP port or HTTPS port through which the DAS connects to the instance to manage the instance. The valid values are in the range of `1-65535`.

  Type: Integer

Default value: N/A
Range Value: 1 to 65535

- HTTP_LISTENER_PORT=*value*
  Specifies the port number that is used to listen HTTP requests. The valid values are in the range of 1-65535.
  Type: Integer
  Default value: N/A
  Range Value: 1 to 65535
- HTTP_SSL_LISTENER_PORT=*value*
  Specifies the port number that is used to listen HTTPS requests. The valid values are in the range of 1-65535.
  Type: Integer
  Default value: N/A
  Range Value: 1 to 65535
- IIOP_LISTENER_PORT=*value*
  Specifies the port number that is used for IIOP connections. The valid values are in the range of 1 to 65535.
  Type: Integer
  Default value: N/A
  Range Value: 1 to 65535
- IIOP_SSL_LISTENER_PORT=*value*
  Specifies the port number that is used for secure IIOP connections. The valid values are in the range of 1 to 65535.
  Type: Integer
  Default value: N/A
  Range Value: 1 to 65535
- IIOP_SSL_MUTUALAUTH_PORT=*value*
  Specifies the port number that is used for secure IIOP connections with client authentication. The valid values are in the range of 1 to 65535.
  Type: Integer
  Default value: N/A
  Range Value: 1 to 65535
- JAVA_DEBUGGER_PORT=*value*
  Specifies the port number that is used for connections to the Java Platform Debugger Architecture (JPDA) (http://java.sun.com/javase/technologies/core/toolsapis/jpda/) debugger. The valid values are in the range of 1 to 65535.
  Type: Integer
  Default value: N/A
  Range Value: 1 to 65535
- JMS_PROVIDER_PORT=*value*
  Specifies the port number for the Java Message Service provider. The valid values are in the range of 1 to 65535.
  Type: Integer
  Default value: N/A
  Range Value: 1 to 65535
- JMX_SYSTEM_CONNECTOR_PORT=*value*

Specifies the port number on which the JMX connector listens. The valid values are in the range of 1 to 65535.

Type: Integer

Default value: N/A

Range Value: 1 to 65535

Default value: N/A

--properties *name*=*value*[:*name*=*value*]...

Defines properties for the cluster. The specification format is name=*value*. Use a colon (:) to separate property names. If you specify the same property name more than once, the property value specified last is valid.

Type: String

The following values can be specified:

- GMS_DISCOVERY_URI_LIST=*value*

  Lists the locations of server instances in the cluster to be used for discovering the cluster. This property is required only if the Group Management Service (GMS) is not using multicast for broadcasting messages.

  The following values are valid for this property:

  A comma-separated list of Uniform Resource Identifiers (URIs). Each URL must locate a server instance or the DAS. This format is required if multiple server instances are running on the same host.

  The format of each URL in the list is as follows:

  scheme://host-name-or -IP-address:port

  scheme is the URI scheme, which is tcp.

  host-name-or -IP-address is the host name or IP address of the host on which the instance is running.

  port is the port number of the port on which the instance listens for messages from GMS. The system property GMS_LISTENER_PORT-cluster-name must be set for the instance.

  A comma-separated list of IP addresses or host names on which the DAS or the instances are running. The list can contain a mixture of IP addresses and host names. This format can be used only if one clustered instance is running on each host. The value of the GMS_LISTENER_PORT property must be unique for each cluster in a domain.

  The keyword generate. This format can be used only if one instance in a cluster is running on each host and the DAS is running on a separate host. Multiple instances on the same host cannot be members of the same cluster. The GMS_LISTENER_PORT value property must be unique for each cluster in a domain.

  The following values are valid for this property:

  Type: String

  Default Value: N/A

  Range: N/A

- GMS_LISTENER_PORT=*value*

  Specifies the port number of the port on which the cluster listens for messages from GMS. The default value is a reference to the GMS_LISTENER_PORT-cluster-name system property. By default, this system property is not set. In this situation, GMS selects a free port from the range that is defined by the properties GMS_TCPSTARTPORT and GMS_TCPENDPORT. By default, this range is 9090-9200. In most situations, the default behavior should suffice. However, if GMS is not using multicast for broadcasting messages, the GMS_LISTENER_PORT property must specify a port number that is valid for all server instances in the cluster. To use the default value to meet this requirement, use a system property to set the port number individually for each instance.

  For example, use the create-system-properties subcommand to create the system property GMS_LISTENER_PORT-cluster-name for the DAS. Then, for each instance in the cluster, set the GMS_LISTENER_PORT-cluster-name system property to the port number on which the instance listens

for messages from GMS. The default value of the GMS_LISTENER_PORT property for the cluster references this system property.

Type: Integer

Default Value: N/A

Range: 1 to 65535

- GMS_LOOPBACK=*value*

  Specifies whether an instance may receive from itself application-level messages that the instance broadcasts to the cluster.

  The possible values are as follows:

  true

  The instance may receive messages from itself. Use this setting for testing an instance when the instance is the only instance in a cluster.

  false

  The instance may not receive messages from itself. This is the default value.

  Type: Boolean

  Default Value: false

  Range:

  true

  false

- GMS_MULTICAST_TIME_TO_LIVE=*value*

  Specifies the maximum number of iterations or transmissions that a multicast message for the following types of events can experience before the message is discarded:

  Group discovery

  Member heartbeats

  Membership changes

  To match the configuration of the network on which the DAS and clustered instances are deployed, set this value as low as possible. To determine the lowest possible value for your system, use the validate-multicast subcommand. A value of 0 ensures that multicast messages never leave the host from which they are broadcast.

  A value of 1 might prevent the broadcast of messages between hosts on same subnet that are connected by a switch or a router. The default value is 4, which ensures that messages are successfully broadcasted to all cluster members in networks where hosts are connected by switches or routers.

  Type: Integer

  Default Value: 4

  Range: 0 to 255

- GMS_TCPENDPORT=*value*

  Specifies the highest port number in the range from which GMS selects a free port if the GMS_LISTENER_PORT-cluster-name system property is not set. The default value is 9200.

  Type: Integer

  Default Value: 9200

  Range: 1 to 65535

- GMS_TCPSTARTPORT=*value*

  Specifies the lowest port number in the range from which GMS selects a free port if the GMS_LISTENER_PORT-cluster-name system property is not set. The default value is 9090.

  Type: Integer

Default Value: `9090`

Range: `1` to `65535`

`--gmsenabled={true|false}`

Specifies whether GMS is enabled for the cluster.

Type: Boolean

The following values can be specified:

- `true`

  GMS is enabled for the cluster (default). When GMS is enabled for a cluster, GMS is started on each server instance in the cluster and in the DAS. The DAS participates in each cluster for which this option is set to true.

- `false`

  GMS is disabled for the cluster.

Default value: `true`

`--multicastaddress` *`multicast-address`*

Specifies the address on which GMS listens for group events.

Type: String

The following values can be specified:

- `224.0.0.0` to `239.255.255.255`

Default value: `228.9.XX.YY` where `XX` and `YY` are automatically-generated independent values between `0` and `255`.

`--multicastport` *`multicast-port`*

Specifies the port number of the communication port on which GMS listens for group events.

Type: Integer

The following values can be specified:

- `2048` to `49151`

Default value: The default value is an automatically generated value in the given range.

`--bindaddress` *`bind-address`*

Specifies the Internet Protocol (IP) address of the network interface to which GMS binds.

Default value: The default value is all the public network interface addresses.

`cluster-name`

Specifies the name of the cluster.

Type: String

The following values can be specified:

- *Specify the cluster name*

Default value: N/A

Examples

The following example creates a cluster.

```
asadmin create-cluster --systemproperties IIOP_SSL_LISTENER_PORT=1169 ltscluster
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.6.5 create-instance

Creates a server instance.

Synopsis

```
asadmin [asadmin-options] create-instance [--help] --node node-name
    [--config config-name | --cluster cluster-name]
    [--portbase=port-number] [--checkports={true|false}]
    [--systemproperties name=value[:name=value]...]
    [--prf prf-name] instance-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-instance` subcommand creates a server instance.

This command requires the secure shell (SSH) protocol to be configured on the host where the Domain Administration Server (DAS) is running and on the host that is represented by the node where the instance must reside.

A server instance is a single Java VM on a single node in which Java EE Server is running. A node defines the host where the server instance resides. Java VM must be compatible with the Java Platform, Enterprise Edition (Java EE).

This subcommand is supported in remote mode only.

SSH is not required if the instance must reside on a node, of the type CONFIG, which represents the local host. A node of type CONFIG is not enabled for remote communication over SSH.

Execution permission

The information given below is for `--systemproperties`:

On UNIX, creating sockets that listen on ports in the range of `1-1024` requires superuser privileges.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--node` *node-name*

Specifies the name of the node that defines the host where the instance is to be created.

Type: String

The following values can be specified:

- *Name of the node*

Default value: `localhost-domain`

`--config` *config-name*

Specifies the named configuration that the instance references. The configuration must exist and must not be named `default-config` or `server-config`.

Type: String

The following values can be specified:

- *Name of the configuration*

Default value: N/A

`--cluster` *cluster-name*

Specifies the cluster from which the instance inherits its configuration.

Type: String

The following values can be specified:

- *Name of cluster*

Default value: N/A

`--portbase=`*port-number*

Confirms the number with which the port assignment should start.

Type: Integer

The following values can be specified:

- `-8` to `65449`

Default value: N/A

`--checkports={true|false}`

Specifies whether to check for the availability of the administration, HTTP, JMS, JMX, and IIOP ports.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--systemproperties` *name*`=`*value*`[:`*name*`=`*value*`]...`

Defines system properties for the instance. These properties override property definitions for port settings in the instance's configuration. The specification format is `name=`*value*. Use a colon (`:`) to separate property names. If you specify the same property name more than once, the property value specified last is valid.

Predefined port settings must be overridden if two clustered instances reside on the same host. In this situation, port settings for one instance must be overridden because both instances share the same configuration.

Type: String

The following values can be specified:

- `ASADMIN_LISTENER_PORT=`*value*

Specifies the port number of the HTTP port or HTTPS port through which the DAS connects to the instance to manage the instance. The valid values are in the range of 1 to 65535.

Type: Integer

Default value: N/A

Range Value: 1 to 65535

- HTTP_LISTENER_PORT=*value*

  Specifies the port number of the port that is used to listen for HTTP requests. The valid values are in the range of 1 to 65535.

  Type: Integer

  Default value: N/A

  Range Value: 1 to 65535

- HTTP_SSL_LISTENER_PORT=*value*

  Specifies the port number of the port that is used to listen for HTTPS requests. The valid values are in the range of 1 to 65535.

  Type: Integer

  Default value: N/A

  Range Value: 1 to 65535

- IIOP_LISTENER_PORT=*value*

  Specifies the port number of the port that is used for IIOP connections. The valid values are in the range of 1 to 65535.

  Type: Integer

  Default value: N/A

  Range Value: 1 to 65535

- IIOP_SSL_LISTENER_PORT=*value*

  Specifies the port number of the port that is used for secure IIOP connections. The valid values are in the range of 1 to 65535.

  Type: Integer

  Default value: N/A

  Range Value: 1 to 65535

- IIOP_SSL_MUTUALAUTH_PORT=*value*

  Specifies the port number of the port that is used for secure IIOP connections with client authentication. The valid values are in the range of 1 to 65535.

  Type: Integer

  Default value: N/A

  Range Value: 1 to 65535

- JAVA_DEBUGGER_PORT=*value*

  Specifies the port number of the port that is used for connections to the Java Platform Debugger Architecture (JPDA) http://java.sun.com/javase/technologies/core/toolsapis/jpda/ debugger. The valid values are in the range of 1 to 65535.

  Type: Integer

  Default value: N/A

  Range Value: 1 to 65535

- JMS_PROVIDER_PORT=*value*

Specifies the port number for the Java Message Service provider. The valid values are in the range of `1` to `65535`.

Type: Integer

Default value: N/A

Range Value: `1` to `65535`

- `JMX_SYSTEM_CONNECTOR_PORT=`*`value`*

  Specifies the port number on which the JMX connector listens. The valid values are in the range of `1` to `65535`.

  Type: Integer

  Default value: N/A

  Range Value: `1` to `65535`

Default value: N/A

`--prf` *`prf-name`*

Specifies the server name of the PRF used by the Java EE server. Specify the PRF server that exists in the node specified in the `--node` option.

Type: String

The following values can be specified:

- *PRF name*

Default value: N/A

`instance-name`

Specifies the name of the instance that is being created.

Type: String

The following values can be specified:

- *Name of the instance*

Default value: N/A

Examples

The following example creates an instance node.

```
asadmin create-instance --node localhost-domain1 pmdsainst
```

Exit Status

Exit Status	Explanation
`0`	command executed successfully.
`1`	error in executing the command.

2.6.6 create-prf

The `create-prf` subcommand creates a Performance Tracer (PRF) server in the domain.

Synopsis

```
asadmin [asadmin_options] create-prf --node node_name
    [--properties name=value[:name=value]...]
    [--template prf_templatedir] prf_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-prf` subcommand creates a PRF server in the domain.

This subcommand is supported only in the remote mode.

Execution permission

General user

Precondition

The node has been created.

Files

Server template file for the PRF server (available for the `--template` option)

domain_root_directory/*domain_name* /server_templates/prf

Arguments

`--node` *node_name*

Specifies the node in which the PRF server is to be created.

Type: String

You can specify the following values:

- Name of a node that exists in the domain.

Default value: (None. You must specify a value.)

`--properties` *name*=*value*[:*name*=*value*]...

Specifies the property names and values in pairs with the standard properties or extended properties of the PRF server. The syntax is *name*=*value*. You can specify multiple values separated by a colon (`:`). If the same property name is specified multiple times, then the later-specified property name is used.

name

Type: String

You can specify the following values:

- A PRF standard property or extended property that begins with `ex_`.

value

You can specify the following values:

- A value in the range of properties defined for the `name` option.

Default value: For a standard property, which is not specified for this option, the value in the PRF default configuration (the configuration name `default-prf-config`) is applied.

`--template` *`prf_templatedir`*

Specifies the absolute path of the directory storing the PRF server template.

Type: String

Default value: The path set in the following attribute is applied: `hitachi-prf-configs.hitachi-prf-config.default-prf-config.hitachi-manage-info.template-path`.

The following path is the default value of this attribute:

`domain_root_directory``/`*`domain_name`*`/server_templates/prf`

`prf_name`

Specifies the name of the PRF server to be created.

Type: String

You can specify a value based on the following conventions:

- Within 31 ASCII characters.
- The name can be contain the following characters:

  Lowercase letters: `a` to `z`

  Uppercase letters: `A` to `Z`

  Numbers: `0` to `9`

  Underscore: _
- The first character can either be an uppercase or a lowercase letter. However, you cannot specify the following values with a combination of both uppercase and lowercase letters:

  TSC

  CTM
- The following names that exist in the domain cannot be specified:

  Node name

  Java EE server name

  Web server name

  PRF server name

  Cluster name

  Name of a dependency relation between servers

  Configuration name of the Java EE server
- The following names cannot be specified because they are either used by the system or are reserved words:

  `domain`

  `server`

  `default`

  `server-config`

  `default-config`

  `default-webserver-config`

  `default-prf-config`

  `javaee`

  `webserver`

  `prf`

  `cluster`

redirect
prf-relation
Name that begins with HJES_
pre_built_web_server_name -config
pre_built_PRF_name -config

Default value: (None. You must specify a value.)

Examples

The following example creates a PRF server named PRF1:

```
asadmin create-prf --node localhost-domain1 PRF1
```

The following example creates a PRF server by specifying a standard property:

```
asadmin create-prf --node localhost-domain1 --properties trace-file-size=65536
PRF1
```

The following example creates a PRF server by specifying the directory of the server template:

```
asadmin create-prf --node localhost-domain1 --template C:\temp\PRF1 PRF1
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

Notes

- If a PRF server has already been created on the host of the configuration destination node by using another domain, do not create the PRF with the same name.

2.6.7 create-webserver

The create-webserver subcommand creates a web server.

Synopsis

```
asadmin [asadmin_options] create-webserver --node node_name
    [--properties name=value[:name=value]...]
    [--template webserver_templatedir]
    [--prf prf_name] webserver_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-webserver` subcommand creates a web server in the domain.

This subcommand is supported only in the remote mode.

Execution permission

General user

Precondition

The node has been created.

Files

Server template file for the web server (available for the `--template` option)

domain_root_directory/*domain_name*/server_templates/webserver

Arguments

`--node` *node_name*

Specifies the node in which the web server to be created.

Type: String

You can specify the following values:

- Name of a node that exists in the domain

Default value: (None. You must specify a value.)

`--properties` *name*=*value*[:*name*=*value*]...

Specifies the property name and value in pairs for the standard properties or extended properties of the web server.

The specified syntax is *name*=*value*. You can specify multiple values separated by a colon (`:`). If the same property name is specified multiple times, then the property name that is specified later is used.

name

Type: String

You can specify the following values:

- Web standard property or extended property that begins with `ex_`.

Default value: For a standard property, which is not specified in this option, the value of the web server default configuration (the configuration name `default-webserver-config`) is applied.

value

You can specify the following values:

- A value in the range of properties defined for the `name` option.

`--template` *webserver_templatedir*

Specifies the absolute path of the directory where the web server template is stored.

Type: String

Default value: The path that is set in the following attribute is applied: `hitachi-webserver-configs.hitachi-webserver-config.default-webserver-config.hitachi-manage-info.template-path`. The following path is the default value of this attribute:

domain_root_directory/*domain_name*/server_templates/webserver

`--prf` *`prf_name`*

Specifies the name of the PRF server used by the web server.

Type: String

You can specify the following values:

- Name of a PRF server that exists in the node specified by the `--node` option.

Default value: (None. No default value is specified.)

`webserver_name`

Specifies the name of the web server to be created.

Type: String

You can specify a value based on the following conventions.

- Within 128 ASCII characters.
- The name can contain the following characters:
  Lowercase letters: `a` to `z`
  Uppercase letters: `A` to `Z`
  Numbers: `0` to `9`
  Hyphen: `-`
  Underscore: `_`
- The first character can either be an uppercase or a lowercase letter.
- The following names that exist in the domain cannot be specified:
  Node name
  Java EE server name
  Web server name
  PRF server name
  Cluster name
  Name of a dependency relation between servers
  Configuration name of Java EE servers
- The following names cannot be specified because they are either used by the system or are reserved words:
  `domain`
  `server`
  `default`
  `server-config`
  `default-config`
  `default-webserver-config`
  `default-prf-config`
  `javaee`
  `webserver`
  `prf`
  `cluster`
  `redirect`
  `prf-relation`
  Name that begins with `HJES_`
  `pre_built_web_server_name``-config`

pre_built_PRF_name-config

Default value: (None. You must specify a value.)

Examples

The following example creates a web server with the name Web1, related to the PRF server PRF1:

```
asadmin create-webserver --node localhost-domain1 --prf PRF1 Web1
```

The following example specifies a standard property that needs to be changed from the default, and then creates a web server:

```
asadmin create-webserver --node localhost-domain1 --properties
 listen-port=88:server-name=www.webserver.com --prf PRF1 Web1
```

The following example specifies a server template, and then creates a web server:

```
asadmin create-webserver --node localhost-domain1 --template /user_template/Web1
 --prf PRF1 Web1
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

Notes

- If a web server is already created on the host of the configuration destination node by using another domain, do not create a web server with the same name.

2.6.8 delete-cluster

Deletes Java EE Server cluster.

Synopsis

```
asadmin [asadmin-options] delete-cluster [--help] cluster-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The delete-cluster subcommand deletes Java EE Server cluster. A cluster can be deleted only if the cluster contains no server instances. If a cluster that you are deleting contains any instances, stop and delete the instances before deleting the cluster.

If the cluster's named configuration was created automatically for the cluster and no other clusters or unclustered instances refer to the configuration, the configuration is deleted when the cluster is deleted. A configuration that is created automatically for a cluster is named `cluster-name-config`, where `cluster-name` is the name of the cluster.

This command is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`cluster-name`

Specifies the name of the cluster.

Type: String

The following values can be specified:

- *Name of the cluster*

Default value: N/A

Examples

The following example deletes Java EE Server cluster `adccluster`.

```
asadmin delete-cluster adccluster
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.6.9 delete-config

Deletes an existing named configuration.

Synopsis

```
asadmin [asadmin-options] delete-config [--help] configuration-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-config` subcommand deletes an existing named configuration from the configuration of the domain administration server (DAS). You can delete a configuration only if no server instances or clusters refer to the configuration. A standalone configuration is automatically deleted when the instance or cluster that refers to it is deleted. You cannot delete the `default-config` configuration that is copied to create standalone configurations.

This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`configuration-name`

Specifies the name of the configuration.

Type: String

The following values can be specified:

- *Name of the configuration*

Default value: N/A

Examples

The following example deletes the named configuration `pmdconfig`.

```
asadmin delete-config pmdconfig
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.6.10 delete-instance

Deletes a server instance.

Synopsis

```
asadmin [asadmin-options] delete-instance [--help] instance-name
```

Storage location

`Application Server installation directory``/javaee/glassfish/bin`

Function

The `delete-instance` subcommand deletes a server instance.

This subcommand requires the secure shell (SSH) protocol to be configured on the host where the domain administration server (DAS) is running and on the host that is represented by the node where the instance resides. However, if the instance to be deleted is configured on a config-type node, you do not need to specify SSH.

You may run this subcommand from any host that can contact the DAS.

The subcommand can delete any server instance, regardless of how the instance was created.

The instance that is being deleted must not be running. Otherwise, an error occurs.

The subcommand deletes an instance by performing the following actions:

- Removing the instance from the configuration of the domain administration server (DAS)
- Deleting the instance's files from file system

If the instance that is being deleted is the only instance that is using the node directory, that directory is also removed.

If a standalone instance is deleted, the instance's standalone configuration is also deleted. A standalone instance refers to a configuration that is named `instance-name-config` to which no other clusters or unclustered instances refer.

This subcommand is supported in remote mode only

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`instance-name`

Specifies the name of the instance.

Type: String

The following values can be specified:

- *Name of the instance*

Default value: N/A

Examples

The following example deletes the server instance `pmdsainst`.

```
asadmin delete-instance pmdsainst
```

Exit Status

Exit Status	Explanation
0	command executed successfully.

Exit Status	Explanation
1	error in executing the command.

2.6.11 delete-prf

The `delete-prf` subcommand deletes a PRF server.

Synopsis

```
asadmin [asadmin_options] delete-prf [--force={false|true}] prf_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-prf` subcommand deletes the specified PRF server from the domain. If the PRF server is in a dependency relation with another server (the PRF server is the dependent source server), that relation is also deleted.

This subcommand is supported only in the remote mode.

Execution permission

General user

Precondition

The PRF server has been created.

The PRF server has been stopped.

Arguments

`--force={false|true}`

Specifies whether to forcibly delete the PRF server.

Type: Boolean

You can specify the following values:

- `true`

  Forces deletion of the PRF server. If the PRF server on the node is not stopped or if deletion failed, the PRF server information is deleted from the domain.

- `false`

  Does not force the deletion of the PRF server. If the deletion of the PRF server on the node failed, the subcommand terminates with an error without deleting the PRF server information from the domain.

Default value: `false`

prf_name

Specifies the name of the PRF server to be deleted.

Type: String

You can specify the following values:

- Name of a PRF server that exists in the domain

Default value: (None. You must specify a value.)

Examples

The following example deletes the PRF server named `PRF1`:

```
asadmin delete-prf PRF1
```

The following example forcibly deletes the PRF server named `PRF1`:

```
asadmin delete-prf --force true PRF1
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

2.6.12 delete-webserver

The `delete-webserver` subcommand deletes a web server.

Synopsis

```
asadmin [asadmin_options] delete-webserver [--force={false|true}]
    webserver_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-webserver` subcommand deletes the specified web server from the domain. If the web server has a dependency relation with another server (the web server is the dependent source server), that relation is also deleted.

This subcommand is supported only in the remote mode.

Execution permission

General user

Precondition

The web server has been created.

The web server has been stopped.

Arguments

`--force={false|true}`

Specifies whether to forcibly delete the web server.

Type: Boolean

You can specify the following values:

- `true`

  Forces deletion of the web server. If the web server on the node is not stopped or if the deletion failed, the web server information is deleted from the domain.

- `false`

  Does not force deletion of the web server. If the deletion of the web server on the node failed, the subcommand terminates with an error without deleting the web server information from the domain.

Default value: `false`

`webserver_name`

Specifies the name of the web server to be deleted.

Type: String

You can specify the following values:

- Name of a web server that exists in the domain

Default value: (None. You must specify a value.)

Examples

The following example deletes the web server named `Web1`:

```
asadmin delete-webserver Web1
```

The following example forcibly deletes the web server named `Web1`:

```
asadmin delete-webserver --force true Web1
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

2.6.13 export-sync-bundle

Exports the configuration data of a cluster or standalone instance to an archive file.

Synopsis

```
asadmin [asadmin-options] export-sync-bundle [--help]
    --target target
```

```
    [--retrieve={false|true}]
    [file-name]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `export-sync-bundle` subcommand exports the configuration data of a cluster or standalone instance to an archive file. The archive file can then be used with the `import-sync-bundle` subcommand to restore the configuration data.

Importing an instance's configuration data transfers the data to a host for an instance without the need for the instance to be able to communicate with the Domain Administration Server (DAS). This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Files

The subcommand creates an archive that contains the following files and directories in the current domain directory:

- All the files in the following directories:
  `config`
  `docroot`
- The entire contents of the following directories and their subdirectories:
  `applications`
  `config/target`, where `target` is the cluster or standalone instance for which configuration data is being exported.
  `generated`
  `lib`

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the cluster or standalone instance for which configuration data is to be exported. The `--target` option is required.

This option must not specify a clustered server instance. If this option specifies a clustered instance, an error occurs. To export configuration data for a clustered instance, specify the name of the cluster of which the instance is a member, not the instance.

Type: String

The following values can be specified:

- *cluster_name*
- *standalone_instance_name*

Default value: N/A

`--retrieve={false|true}`

Specifies whether the archive file is downloaded from the DAS host to the host where the subcommand is run.

Type: Boolean

The following values can be specified:

- `true`
  The archive file is downloaded to the host where the subcommand is run.
- `false`
  The archive file is not downloaded and remains on the DAS host (default).

Default value: `false`

`file-name`

Specifies the file name and location of the archive file to which the data is to be exported.

Type: String

The following values can be specified:

- *Name of the file*

Default value:

- If `--retrieve` is `false`
  `sync/target-sync-bundle.zip`
- If `--retrieve` is `true`
  `target-sync-bundle.zip` in the current working directory

Examples

The following example exports the configuration data of the cluster `pmdcluster`.

```
asadmin export-sync-bundle --target pmdcluster
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.6.14 get-health

Provides information on the cluster health.

Synopsis

```
asadmin [asadmin-options] get-health [--help]
    cluster_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The get-health subcommand gets information about the health of the cluster. Note that if the Group Management Service (GMS) is not enabled, the basic information about whether the server instances in this cluster are running or not running is not returned. For each server instance, one of the following states is reported:

- not started
- started
- stopped
- rejoined
- failed

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

--help | -?

Displays the help text for the subcommand.

cluster_name

Specifies the name of the cluster for which you want the health information. This subcommand prompts you for the cluster name if you do not specify it.

A Cluster must be present in the DAS.

Type: String

The following values can be specified:

- *Name of the cluster*

Default value: N/A

Examples

The following example checks the health of server instances in a cluster.

```
asadmin get-health cluster1
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.6.15 import-sync-bundle

Imports the configuration data of a clustered instance or standalone instance from an archive file.

Synopsis

```
asadmin [asadmin-options] import-sync-bundle [--help]
    --instance instance-name
    [--nodedir node-dir] [--node node-name]
    file-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `import-sync-bundle` subcommand imports the configuration data of a clustered instance or standalone instance from an archive file that was created by the `export-sync-bundle` subcommand. Importing an instance's configuration data transfers the data to a host for an instance without the need for the instance to be able to communicate with the DAS. This subcommand is supported in local mode only.

Subcommand must run on the host where the instance resides. To contact the Domain Administration Server (DAS), this subcommand requires the name of the host where the DAS is running. If a non-default port is used for administration, this subcommand also requires the port number.

Precondition

For `--instance` option below is the pre-condition:

The instance must already exist in the DAS configuration. The archive file from which the data is being imported must contain data for the specified instance.

Files

The `import-sync-bundle` subcommand imports the configuration data of a clustered instance or standalone instance from an archive file (sync-bundle archive file) that was created by the `export-sync-bundle` subcommand.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--instance` *instance-name*

Specifies the instance for which configuration data is being imported. The instance must already exist in the DAS configuration. The archive file from which the data is being imported must contain data for the specified instance.

Type: String

The following values can be specified:

- *Name of the instance*

Default value: N/A

`--nodedir` *`node-dir`*

Specifies the directory that contains the instance's node directory. The instance's files are stored in the instance's node directory.

Type: String

The following values can be specified:

- *Path of the node directory*

Default value: *`Application Server installation directory`*`/javaee/glassfish/nodes`

`--node` *`node-name`*

Specifies the node on which the instance resides. If this option is omitted, the subcommand determines the node from the DAS configuration in the archive file.

Type: String

The following values can be specified:

- *Name of the node*

Default value: *node of the DAS configuration*

`file-name`

Specifies the name of the file, including the path that contains the archive file to be imported. This operand is required.

Type: String

The following values can be specified:

- *Name of the file including the path*

Default value: N/A

Examples

The following example imports the configuration for the clustered instance `ymli2` on the node `sj02` from the archive file `/export/glassfish3/glassfish/domains/domain1/sync/ymlcluster-sync-bundle.zip`.

The command is run on the host `sj02`, which is the host that the node `sj02` represents. The DAS is running on the host `sr04` and uses the default HTTP port for administration.

```
sj02# asadmin --host sr04 import-sync-bundle --node sj02 --instance ymli2
/export/glassfish3/glassfish/domains/domain1/sync/ymlcluster-sync-bundle.zip
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.6.16 list-clusters

Lists existing clusters in a domain.

Synopsis

```
asadmin [asadmin-options] list-clusters [--help]
    [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-clusters` subcommand lists the existing clusters in a domain. The list can be filtered by cluster, instance, node, or a configuration. For each cluster that is listed, the subcommand indicates whether the cluster is running. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

target

Filters the list of clusters by specifying the target for which the clusters are to be listed.

Type: String

The following values can be specified:

- `domain`
  Lists all clusters in the domain. This is the default value.
- *cluster-name*
  Lists only the specified cluster.
- *instance-name*
  Lists the cluster of which the specified instance is a member.
- *node-name*
  Lists the clusters that contain an instance that resides on the specified node. For example, if instance `pmdi1` in cluster `pmdc` and instance `ymli1` in cluster `ymlc` reside on node `n1`, `pmdc` and `ymlc` are listed.
- *configuration-name*
  Lists all clusters that contain instances whose configuration is defined by the named configuration.

Default value: `domain`

Examples

The following example lists all clusters in the current domain.

```
asadmin list-clusters
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.6.17 list-instances

Lists the server instances in a domain.

Synopsis

```
asadmin [asadmin-options] list-instances [--help]
    [--timeoutmsec timeout]
    [--long={false|true} | --nostatus={false|true}]
    [--standaloneonly={false|true} | target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-instances` subcommand lists server instances in a domain. The list can be filtered by cluster, instance, node, or configuration.

The subcommand displays every server instance in the specified target, regardless of how each instance was created. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--timeoutmsec` *timeout*

Specifies the time limit in milliseconds for determining the status of instances. The default is `60,000`, which is equivalent to 60 seconds.

Type: Integer

The following values can be specified:

- `-2147483648` to `2147483647`

Default value: `60000` milliseconds

`--long={false|true} | -l={false|true}`

Specifies whether detailed information is displayed for each instance that is listed.

The `--long` option and `--nostatus` option are mutually exclusive. If both options are specified in the same command, an error occurs.

Type: Boolean

The following values can be specified:

- `true`

  The following details are displayed for each instance that is listed:

  The name of the instance

  The name of the host where the instance's node resides

  The HTTP port on which the instance listens for administration requests

  The process identifier (PID) of the instance process or `-1` if the instance is not running

  The name of the cluster of which the instance is a member, if any

  The state of the instance, which is `running` or `not running`

  When an instance is listed, some configuration changes in the domain administration server (DAS) for the instance might not have been applied to the instance itself. In this situation, the commands that are required to apply the changes are listed adjacent to the state of the instance. The maximum number of commands that are listed for an instance is 10.

- `false`

  Only the name of the instance and an indication of whether the instance is running are displayed (default). The length of time that the instance has been running is not displayed.

Default value: `false`

`--nostatus={false|true}`

Specifies whether information about whether instances are running is suppressed.

The `--long` option and `--nostatus` option are mutually exclusive. If both options are specified in the same command, an error occurs.

Type: Boolean

The following values can be specified:

- `true`

  Information about whether instances are running is suppressed. Only the name of each instance is displayed.

- `false`

  Information about whether instances are running is displayed. This is the default value.

Default value: `false`

`--standaloneonly={false|true}`

Specifies whether only standalone instances are listed.

The `--standaloneonly` option and the *target* operand and are mutually exclusive. If both the `--standaloneonly` option and the *target* operand are specified in the same command, an error occurs.

Type: Boolean

The following values can be specified:

- `true`

  Only standalone instances are listed.

- `false`

  All instances in the specified target are listed. This is the default value.

Default value: `false`

target

Filters the list of server instances by specifying the target for which instances are listed.

The *target* operand and the `--standaloneonly` option are mutually exclusive. If both the *target* operand and the `--standaloneonly` option are specified in the same command, an error occurs.

Type: String

The following values can be specified:

- `domain`
  Lists all instances in the domain (default).
- *cluster-name*
  Lists the instances that are members of the specified cluster.
- *instance-name*
  Lists only the specified instance.
- *node-name*
  Lists the instances that reside on the specified node.
- *configuration-name*
  Lists all instances whose configuration is defined by the specified named configuration.

Default value: `domain`

Examples

This example lists detailed information about all server instances in the current domain.

```
asadmin list-instances --long=true
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.6.18 list-prfs

The `list-prfs` subcommand lists the PRF servers.

Synopsis

```
asadmin [asadmin_options] list-prfs [--long={false|true} |
    --nostatus={false|true}] [target]
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `list-prfs` subcommand lists the PRF (Performance Tracer) servers that have been set up in the domain. You can use this command to check the PRF server configuration information and operation status.

This subcommand is supported only in the remote mode.

Execution permission

General user

Arguments

`--long={false|true}`

Specifies whether to display the detailed information, in a list, about the PRF server.

Type: Boolean

You can specify the following values:

- `true`

  Displays the PRF server name and its detailed information (host name, process ID, and status). The header information is also displayed.

- `false`

  Displays only the PRF name and its status. The header information is not displayed.

Default value: `false`

`--nostatus={false|true}`

Specifies whether to display the status of a PRF server.

Type: Boolean

You can specify the following values:

- `true`

  Does not display the status.

- `false`

  Displays the status.

Default value: `false`

`target`

Specifies the name of the object for which information is displayed. Based on the specified name, the PRF servers are filtered and displayed.

Type: String

You can specify the following values:

- `domain`

  Displays information about all the PRF servers in a domain.

- *PRF_server_name*

  Displays information about the specified PRF server.

- *node_name*

  Displays information about the PRF servers that belong to the specified node.

Default value: `domain`

Output Format

The first line is the header of the output information and displays a fixed character string, which indicates the output format.

Between columns, the longest character string within a column is separated from the beginning of the next column by two spaces.

The status is displayed with one space at the beginning of the word.

```
NAME          HOST     PID         STATE
PRF_server_name  host_name  process_ID   status
```

PRF_server_name: Displays the name of the PRF server.

host_name: Displays the name of the host on which the PRF server is created.

process_ID: Displays the process ID of a PRF server that is running. This column displays -- if the PRF server is not running.

status: Displays the status of the PRF server. This column displays the following values:

- not running (Stopped)
- running (Running)

Examples

The following example displays the names and statuses of PRF servers:

```
asadmin list-prfs
```

Example of output:

```
PRF1   running
PRF2   not running
PRF4   running
PRF3   not running
```

The following example displays the complete information about a PRF server:

```
asadmin list-prfs --long true
```

Example of output:

```
NAME  HOST       PID  STATE
PRF1  localhost  258   running
PRF2  localhost  --    not running
PRF4  WINHOST1   547   running
PRF3  WINHOST1   --    not running
```

The following example does not display the PRF status:

```
asadmin list-prfs --nostatus true
```

Example of output:

```
PRF1
PRF2
```

```
PRF4
PRF3
```

The following example displays information corresponding to the PRF server `PRF1`:

```
asadmin list-prfs PRF1
```

Example of output:

```
PRF1   running
```

The following example displays information about the PRF servers that belong to the node `winhost1-domain2` (host `WINHOST1`):

```
asadmin list-prfs --long true winhost1-domain2
```

Example of output:

```
NAME  HOST      PID  STATE
PRF4  WINHOST1  547   running
PRF3  WINHOST1  --    not running
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

2.6.19 list-webservers

The `list-webservers` subcommand lists the web servers.

Synopsis

```
asadmin [asadmin_options] list-webservers [--long={false|true} |
    --nostatus={false|true}] [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-webservers` subcommand lists all the web servers that have been set up in the domain. You can use this subcommand to check the web server configuration information and operation status.

This subcommand is supported only in the remote mode.

You can simultaneously specify the `--long` and `--nostatus` options for the format only if one of these options is explicitly set to `false`.

Execution permission

General user

Arguments

`--long={false|true}`

Specifies whether to display the detailed information, in a list, about the web server.

Type: Boolean

You can specify the following values:

- `true`

  Displays the web server name and its detailed information (host name, process ID, and status). The header information is also displayed.

- `false`

  Displays only the web server name and its status. The header information is not displayed.

Default value: `false`

`--nostatus={false|true}`

Specifies whether to display the status of a web server.

Type: Boolean

You can specify the following values:

- `true`

  Does not display the status.

- `false`

  Displays the status.

Default value: `false`

`target`

Specifies the name of the object for which information is displayed. Based on the specified name, the web servers are filtered and displayed.

Type: String

You can specify the following values:

- `domain`

  Displays information about all the web servers in a domain.

- *web_server_name*

  Displays information about the specified web server.

- *node_name*

  Displays information about the web servers that belong to the specified node.

Default value: `domain`

Output Format

The first line is the header of the output information and displays a fixed character string, which indicates the output format.

Between columns, the longest character string within a column is separated from the beginning of the next column by two spaces.

The status is displayed with one space at the beginning of the word.

```
NAME       HOST    PID       STATE
web_server_name host_name process_ID status
```

web_server_name: Displays the name of the web server.

host_name: Displays the name of the host on which the web server is created.

process_ID: Displays the process ID of the web server that is running. This column displays -- if the web server is not running.

status: Displays the status of the web server. This column displays the following values:

- not running (Stopped)
- running (Running)

Examples

The following example displays the names and statuses of web servers:

```
asadmin list-webservers
```

Example of output:

```
Web1 running
Web2 not running
Web4 running
Web3 not running
```

The following example displays the complete information about a web server:

```
asadmin list-webservers --long true
```

Example of output:

```
NAME  HOST       PID  STATE
Web1  localhost  258   running
Web2  localhost  --    not running
Web4  WINHOST1   526   running
Web3  WINHOST1   --    not running
```

The following example does not display the status of web servers:

```
asadmin list-webservers --nostatus true
```

Example of output:

```
Web1
Web2
Web4
Web3
```

The following example displays information corresponding to the web server `Web1`:

```
asadmin list-webservers Web1
```

Example of output:

```
Web1   running
```

The following example displays information about the web servers that belong to the node `winhost1-domain2` (host `WINHOST1`):

```
asadmin list-webservers winhost1-domain2
```

Example of output:

```
Web4   running
Web3   not running
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

2.6.20 start-cluster

Starts a cluster.

Synopsis

```
asadmin [asadmin-options] start-cluster [--help]
    [--verbose={false|true}] cluster-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `start-cluster` subcommand of `asadmin` starts a cluster by starting all the server instances in the cluster that are not already running. This subcommand is supported in remote mode only.

When this subcommand is used, there is a timeout period for starting the cluster. If a timeout occurs while starting the cluster or reading the `asadmin` command, then this subcommand fails to run. However, the start processing of the cluster continues. You can verify whether the cluster has been started successfully, by viewing the KDKD20031-I log or the server instance status in the Administration Console.

Precondition

Secure shell (SSH) must be configured on the host where the Domain Administration Server (DAS) is running and on all hosts where instances in the cluster reside. However, if the instances in the cluster are all configured on `config-type` nodes, then you need not specify SSH.

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--verbose={false|true}`

Specifies whether additional status information is displayed when the cluster is started.

Type: Boolean

The following values can be specified:

- `true`

  Displays the command to start each instance in the cluster and whether the attempt to start each instance succeeded.

- `false`

  Displays no additional status information. This is the default value.

Default value: `false`

`cluster-name`

Specifies the name of the cluster to start.

Type: String

The following values can be specified:

- *Name of the cluster*

Default value: N/A

Examples

The following example starts the cluster `ymlcluster`. Additional status information is displayed when the cluster is started.

```
asadmin start-cluster --verbose ymlcluster
```

Exit Status

Exit Status	Explanation
`0`	command executed successfully.
`1`	error in executing the command.

2.6.21 start-instance

Starts a server instance.

Synopsis

```
asadmin [asadmin-options] start-instance [--help]
    [--debug={false|true}] [--sync={normal|full|none}] instance-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `start-instance` subcommand of `asadmin` starts a server instance. You may run this subcommand from any machine that can contact the DAS. The subcommand can start any server instance, regardless of how the instance was created.

This subcommand is supported in remote mode only.

When the log identifier is changed from default and the `--sync=full` option is specified in the `start-instance` subcommand, the following files which are displayed in *Node root directory*/*Node name*/*Java EE server name*/`config` are deleted because Java EE server restarts. Therefore, create a backup of these files before running this subcommand.

- The thread dump log of server instance (Default log identifier is `javacore`)
- Memory dump (Default log identifiers are `core`, `.core`, and `*.dmp`)
- Error report file (Default log identifier is `hs_err_pid`)
- Compiler replay file (Default log identifier is `replay_pid`

When this subcommand is used, there is a timeout period for starting the server instance. If a timeout occurs while requesting to the Domain Administration Server, the start processing continues. If a timeout occurs while starting the server instance or reading the `asadmin` command, this subcommand fails to run. However, the start processing of the server instance continues. You can verify whether the server instance has been started successfully, by viewing the KDKD20031-I log or the server instance status in the Administration Console.

Precondition

The remote node where the instance resides must be enabled for remote communication.

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--debug={false|true}`

Specifies whether the instance is started with Java Platform Debugger Architecture (JPDA) debugging enabled.

Type: Boolean

The following values can be specified:

- `true`

  The instance is started with JPDA debugging enabled and the port number for JPDA debugging is displayed.

- `false`

  The instance is started with JPDA debugging disabled. This is the default value.

Default value: `false`

`--sync={normal|full|none}`

Specifies the type of synchronization between the DAS and the instance's files when the instance is started.

Type: String

The following values can be specified:

- `none`

  The DAS does not synchronize the instance's files with any changes. This type of synchronization minimizes the time that is required to start the instance.

- `normal`

  The DAS synchronizes the instance with changes since the last synchronization as follows:

  For the `config` directory, the DAS synchronizes the instance with all changes.

  For the `applications` directory and `docroot` directory, only a change to a top-level subdirectory causes the DAS to synchronize all files under that subdirectory.

  If a file below a top level subdirectory is changed without a change to a file in the top level subdirectory, full synchronization is required. In normal operation, files below the top level subdirectories of these directories are not changed. If an application is deployed and undeployed, full synchronization is not necessary to update the instance with the change. This is the default value.

- `full`

  The DAS synchronizes the instance with all the instance files, regardless of whether the files have changed since the last synchronization. This type of synchronization might delay the startup of the instance while the DAS updates all files in the instance directories.

Default value: `normal`

`instance-name`

Specifies the name of the server instance to start.

Type: String

The following values can be specified:

- *Name of the server instance*

Default value: N/A

Examples

The following example starts the server instance `pmdsa1`.

```
asadmin start-instance pmdsa1
```

Exit Status

Exit Status	Explanation
`0`	command executed successfully.
`1`	error in executing the command.

2.6.22 start-prf

The `start-prf` subcommand starts a PRF server.

Synopsis

```
asadmin [asadmin_options] start-prf prf_name
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `start-prf` subcommand starts the specified PRF server.

This subcommand is supported only in the remote mode.

When this subcommand is used, there is a timeout period for starting PRF server. If a timeout occurs while requesting to the Domain Administration Server, the start processing continues. If a timeout occurs while starting PRF server or reading the `asadmin` command, this subcommand fails to run. However, the start processing of PRF server continues. You can verify whether PRF server has been started successfully, by viewing `list-prfs` subcommand or PRF status in the Administration Console.

Execution permission

General user

Precondition

The PRF server has been created.

Arguments

prf_name

Specifies the name of the PRF server to be started.

Type: String

You can specify the following values:

- Name of a PRF server that exists in the domain

Default value: (None. You must specify a value.)

Examples

The following example starts the PRF server `PRF1`:

```
asadmin start-prf PRF1
```

Exit Status

Exit Status	Explanation
0	Normal termination.

Exit Status	Explanation
1	Abnormal termination.

2.6.23 start-servers

The `start-servers` subcommand starts all the servers in a batch.

Synopsis

```
asadmin [asadmin_options] start-servers
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `start-servers` subcommand starts, in a batch, all the servers in the domain. If there is a dependency relation set between servers, the depended-on destination server is started first. If there are many dependency relations, the servers are started in order from the server at the highest-level of the dependency tree. If no dependency is set between the servers, the subcommand processes all the servers simultaneously.

The processing to start the server is not performed for a server that is already running.

This subcommand is supported only in the remote mode.

When the log identifier is changed from default and the `--sync=full` option is specified in the `start-instance` subcommand, the following files which are displayed in *Node root directory*/*Node name*/*Java EE server name*/config are deleted because Java EE server restarts. Therefore, create a backup of these files before running this subcommand.

- The thread dump log of server instance (Default log identifier is `javacore`)
- Memory dump (Default log identifiers are `core`, `.core`, and `*.dmp`)
- Error report file (Default log identifier is `hs_err_pid`)
- Compiler replay file (Default log identifier is `replay_pid`

Execution permission

General user

Precondition

Various servers have been set up.

Examples

The following example starts all the servers in the domain:

```
asadmin start-servers
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

2.6.24 start-webserver

The `start-webserver` subcommand starts a web server.

Synopsis

```
asadmin [asadmin_options] start-webserver webserver_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `start-webserver` subcommand starts the specified web server.

This subcommand is supported only in the remote mode.

When this subcommand is used, there is a timeout period for starting web server. If a timeout occurs while requesting to the Domain Administration Server, the start processing continues. If a timeout occurs while starting web server or reading the `asadmin` command, this subcommand fails to run. However, the start processing of web server continues. You can verify whether web server has been started successfully, by viewing `list-webservers` subcommand or web server status in the Administration Console.

Execution permission

General user

Precondition

The web server has been created.

Arguments

webserver_name

Specifies the name of the web server to be started.

Type: String

You can specify the following values:

- Name of a web server that exists in the domain

Default value: (None. You must specify a value.)

Examples

The following example starts the web server `Web1`:

```
asadmin start-webserver Web1
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

2.6.25 stop-cluster

Stops Java EE Server cluster.

Synopsis

```
asadmin [asadmin-options] stop-cluster [--help]
    [--verbose={false|true}] [--kill={false|true}] cluster-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `stop-cluster` subcommand of `asadmin` stops Java EE Server cluster by stopping all running server instances in the cluster. This subcommand is supported in remote mode only.

When this subcommand is used, there is a timeout period for stopping the cluster. If a timeout occurs while stopping the cluster or reading the `asadmin` command, then this subcommand fails to run. However, the stop processing of the cluster continues. You can verify whether the cluster has been stopped successfully, by viewing the KDKD20032-I log or the server instance status in the Administration Console.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--verbose={false|true}`

Specifies whether additional progress messages about the status of instances in the cluster are displayed while the cluster is being stopped.

Type: Boolean

The following values can be specified:

- `true`

Additional progress messages about the status of instances in the cluster are displayed.

- `false`

  No messages about the status of instances in the cluster are displayed.

Default value: `false`

`--kill={false|true}`

Specifies whether each instance in the cluster is killed using functionality of the operating system to terminate the instance process.

Type: String

The following values can be specified:

- `true`

  Each instance is killed. The subcommand uses functionality of the operating system to terminate each instance process.

- `false`

  No instances are killed. The subcommand uses functionality of the Java platform to terminate each instance process. This is the default value.

Default value: `false`

`cluster-name`

Specifies the name of the cluster to stop.

Type: String

The following values can be specified:

- *Name of the cluster*

Default value: N/A

Examples

The following example stops the cluster `pmdcluster`.

```
asadmin stop-cluster pmdcluster
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.6.26 stop-instance

Stops a running server instance.

Synopsis

```
asadmin [asadmin-options] stop-instance [--help]
     [--kill={false|true}] instance-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The stop-instance subcommand of asadmin stops a running server instance. The subcommand can stop any server instance, regardless of how the instance was created.

Before stopping a server instance, stop from the instances on the front-end server, such as the load-balancing device and the Web server. If you do not stop from the front-end server and requests are left in the server instance, an unexpected error might occur.

When stopping a server, the message KDKD45000-E might be output to the standard error and logger.log files, but the termination processing is not affected.

This subcommand is supported in remote mode only.

When this subcommand is used, there is a timeout period for stopping the server instance. If a timeout occurs while requesting to the Domain Administration Server, the stop processing continues. If a timeout occurs while stopping the server instance or reading the asadmin command, this subcommand fails to run. However, the stop processing of the server instance continues. You can verify whether the server instance has been stopped successfully, by viewing the KDKD20032-I log or the server instance status in the Administration Console.

Precondition

Domain Administration Server (DAS) is running.

Arguments

--help | -?

Displays the help text for the subcommand.

--kill={false|true}

Specifies whether the instance is killed by using functionality of the operating system to terminate the instance process.

Type: Boolean

The following values can be specified:

- true: The instance is killed. The subcommand uses functionality of the operating system to terminate the instance process.
- false: The instance is not killed. The subcommand uses functionality of the Java platform to terminate the instance process. This is the default value.

Default value: false

instance-name

Specifies the name of the server instance to stop.

Type: String

The following values can be specified:

- *Name of the server instance*

Default value: N/A

Examples

The following example stops the server instance `yml-i-sj01`.

```
asadmin stop-instance yml-i-sj01
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.6.27 stop-prf

The `stop-prf` subcommand stops a PRF server.

Synopsis

```
asadmin [asadmin_options] stop-prf prf_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `stop-prf` subcommand stops the specified PRF server.

This subcommand is supported only in the remote mode.

When this subcommand is used, there is a timeout period for stopping the PRF server. If a timeout occurs while requesting to the Domain Administration Server, the stop processing continues. If a timeout occurs while stopping the PRF server or reading the `asadmin` command, this subcommand fails to run. However, the stop processing of the PRF server continues. You can verify whether the PRF server has been stopped successfully, by viewing `list-prfs` subcommand or the PRF status in the Administration Console.

Execution permission

General user

Precondition

The PRF server has been started.

Arguments

prf_name

Specifies the name of the PRF server to be stopped.

Type: String

You can specify the following values:

- Name of a PRF server that exists in the domain

Default value: (None. You must specify a value.)

Examples

The following example stops the PRF server `PRF1`:

```
asadmin stop-prf PRF1
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

2.6.28 stop-servers

The `stop-servers` subcommand stops all the servers in a batch.

Synopsis

```
asadmin [asadmin_options] stop-servers [--graceful {false|true}]
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `stop-servers` subcommand stops, in a batch, all the servers in the domain. If a dependency relation is set between servers, the dependent server is stopped first. If there are many dependencies, the servers are stopped in order beginning from the server at the lowest-level of the dependency tree. If no dependency is set between the servers, the subcommand processes all the servers simultaneously.

The processing to stop the server is not performed for a server that is already stopped.

This subcommand is supported only in the remote mode.

Execution permission

General user

Precondition

Various servers have been started.

Arguments

`--graceful {false|true}`

Specifies whether to stop the web servers in a planned termination. This argument is applicable only to web servers.

Type: Boolean

You can specify the following values:

- `true`
  Stops the web server in a planned termination.
- `false`
  Stops the web server normally.

Default value: `true`

Examples

The following example stops all the servers in the domain:

```
asadmin stop-servers
```

Exit Status

Exit Status	Explanation
`0`	Normal termination.
`1`	Abnormal termination.

2.6.29 stop-webserver

The `stop-webserver` subcommand stops a web server.

Synopsis

```
asadmin [asadmin_options] stop-webserver [--graceful {false|true}]
    webserver_name
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `stop-webserver` subcommand stops the specified web server.

This subcommand is supported only in the remote mode.

When this subcommand is used, there is a timeout period for stopping web server. If a timeout occurs while requesting to the Domain Administration Server, the stop processing continues. If a timeout occurs while stopping web server or reading the `asadmin` command, this subcommand fails to run. However, the stop processing of web server continues. You can verify whether web server has been stopped successfully, by viewing `list-webservers` subcommand or web server status in the Administration Console.

Execution permission

General user

Precondition

The web server has been started.

Arguments

`--graceful {false|true}`

Specifies whether to stop the web server in a planned termination.

Type: Boolean

You can specify the following values:

- `true`

  Stops the web server in a planned termination.

- `false`

  Stops the web server normally.

Default value: `true`

`webserver_name`

Specifies the name of the web server to be stopped.

Type: String

You can specify the following values:

- Name of a web server that exists in the domain

Default value: (None. You must specify a value.)

Examples

```
asadmin stop-webserver Web1
```

```
asadmin stop-webserver --graceful true Web1
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

2.6.30 validate-multicast

Validates that the multicast transport is available for clusters.

Synopsis

```
asadmin [asadmin-options] validate-multicast
    [--help] [--multicastport multicastport]
    [--multicastaddress multicastaddress] [--bindaddress bindaddress]
    [--sendperiod sendperiod] [--timeout timeout]
    [--timetolive timetolive] [--verbose={false|true}]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The validate-multicast subcommand of asadmin validates that the multicast transport is available for clusters. You must run this subcommand at the same time on each of the hosts to be validated. As long as all machines see each other, multicast is validated to be working properly across the machines. If the machines are not seeing each other, set the --bindaddress option explicitly to ensure that all machines are using interface on same subnet, or increase the --timetolive option. If these changes fail to resolve the multicast issues, ask the network administrator to verify that the network is configured so the multicast messages can be seen between all the machines used to run the cluster. This subcommand is supported in local mode only.

Do not run the validate-multicast subcommand using the DAS and cluster's multicast address and port values while the DAS and cluster are running. Doing so results in an error.

Arguments

--help | -?

Displays the help text for the subcommand.

--multicastport *multicastport*

Specifies the port for the multicast socket on which the Group Management Service (GMS) listens for group events. Specify a standard UDP port number in the range 2048 to 32000.

Type: Integer

The following values can be specified:

- 2048 to 32000

Default value: 2048

--multicastaddress *multicastaddress*

Specifies the address for the multicast socket on which the GMS listens for group events. Specify a class D IP address. Class D IP addresses are in the range 224.0.0.0 to 239.255.255.255, inclusive. The address 224.0.0.0 is reserved and must not be used.

Type: IP

The following values can be specified:

- 224.0.0.0 to 239.255.255.255

Default value: 228.9.3.1

--bindaddress *bindaddress*

Specifies the local interface to receive multicast datagram packets for the GMS.

Type: String

The following values can be specified:

- *Local network interface IP address*

Default value: all available binding interfaces

--sendperiod *sendperiod*

Specifies the number of milliseconds between test messages sent between nodes.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `2000`

`--timeout` *`timeout`*

Specifies the number of seconds before the subcommand times out and exits. The default is `20`. You can also exit this subcommand using **Ctrl**+**C**.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `20`

`--timetolive` *`timetolive`*

Specifies the default `time-to-live` for multicast packets sent out on the multicast socket in order to control the scope of the multicasts. The `time-to-live` value must be between `0` and `255` inclusive. The default is the JDK default or a minimum defined by a constant in the GMS subsystem, whichever is lower.

Type: Integer

The following values can be specified:

- `0` to `255`

Default value: lower of JDK default or a constant in the GMS subsystem.

`--verbose={false|true}`

Provides additional debugging information (If used without a value or set to `true`).

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

Examples

The following example checks whether multicast transport is available for a cluster named `c1`.

Run from host `sr1`:

```
asadmin validate-multicast
```

Exit Status

Exit Status	Explanation
`0`	command executed successfully.
`1`	error in executing the command.

2.7 Commands used for setting server relations

This section describes the syntax and functionality of the commands used for setting server relatations.

2.7.1 create-relation

The `create-relation` subcommand creates a relation between servers.

Synopsis

```
asadmin [asadmin_options] create-relation
    --relationtype relation_type_name --from server_name
    --to server_name [--order order]
    [--properties name=value[:name=value]...] relation_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-relation` subcommand creates a relation between servers. You can create a relation by specifying the names of the relation source server and destination server, which correspond to the relation type. This command creates the relation information as relation information of the server specified in the `--from` option.

This subcommand is supported only in the remote mode.

Execution permission

General user

Precondition

Various servers have been setup.

Arguments

`--relationtype` *relation_type_name*

Specifies the type of the relation to be made between the relation source server and relation destination server.

Type: String

You can specify the following values:

- `prf-relation`
  PRF server relation
- `redirect`
  Redirect relation

Default value: (None. You must specify a value.)

`--from` *server_name*

Specifies the name of the server that will become the relation source. Specify a server of the server type that is set for the dependent source server of the relation type.

Type: String

You can specify the following values:

- If `--relationtype` is `prf-relation`, specify a following value:

  Name of a Java EE server existing in the domain.

  Name of a web server existing in domain.

- If `--relationtype` is `redirect`, specify the

  Name of a web server existing in the domain.

Default value: (None. You must specify a value.)

`--to` *server_name*

Specifies the name of the server that will become the relation destination. Specify a server of the server type that is set for the depended-on destination server, of the relation type.

Type: String

You can specify the following values:

- If `--relationtype` is `prf-relation`, specify the

  Name of a PRF server existing in the domain.

- If `--relationtype` is `redirect`, specify the

  Name of a Java EE server existing in the domain.

Default value: (None. You must specify a value.)

`--order` *order*

Specifies the sequence number of the relation when multiple dependency relations of the same relation type are created for the same relation source server.

Type: Integer

You can specify the following values:

- Integer: `0` to `65535`

Default value: `0`

`--properties` *name*`=`*value*`[:`*name*`=`*value*`]...`

name

Specifies the property names and values in pairs, in the properties that apply to the relation information between servers. The syntax is *name*`=`*value*. You can specify multiple values separated by a colon (`:`). If the same property name is specified multiple times, then the property name that is specified later is used.

Type: String

You can specify the following values:

- PRF standard property or extension property that begins with `ex_`.

value

You can specify the following values:

- A value in the range of properties defined for the `name` option.

Default value:

If `redirect` is specified for `--relationtype`, the default value is: `hitachi-relation-types.hitachi-relation-type.redirect.property.`*`value_of_the_standard_property_key`*.

If `prf-relation` is specified for `--relationtype`: (None)

`relation_name`

Specifies the name of the relation between the related servers.

Type: String

You can specify the following values:

- ASCII characters
- The name can contain the following characters:
  Lowercase letters: `a` to `z`
  Uppercase letters: `A` to `Z`
  Numbers: `0` to `9`
  Hyphens: `-`
  Underscores: `_`
- The first character can either be an uppercase or a lowercase letter.
- The following names that exist in the domain cannot be specified:
  Node name
  Java EE server name
  Web server name
  PRF server name
  Cluster name
  Name of a relation between servers
  Name of the configuration of the Java EE server
- The following names that are listed cannot be specified because they are either used by the system or are reserved words:
  `domain`
  `server`
  `default`
  `server-config`
  `default-config`
  `default-webserver-config`
  `default-prf-config`
  `javaee`
  `webserver`
  `prf`
  `cluster`
  `redirect`
  `prf-relation`
  Name that begins with `HJES_`
  `prebuilt_web_server_name` `-config`
  `prebuilt_PRF_name` `-config`

Default value: (None. You must specify a value.)

Examples

The following example creates a PRF related server (the relation type is `prf-relation` and the relation name is `redirect1`) between the Java EE server `JavaEE1` and the PRF server `PRF1` :

```
asadmin create-relation --relationtype prf-relation --from JavaEE1
--to PRF1 redirect1
```

The following example creates a redirect relation server (the relation type is `redirect` and the relation name is `redirect3`) between the web server `Web1` and the Java EE server `JavaEE1` :

```
asadmin create-relation --relationtype redirect --from Web1 --to JavaEE1
--properties path=/aaa/:network-listener=http-listener-1 redirect3
```

The following example creates a relation redirect server by specifying the order number:

```
asadmin create-relation --relationtype redirect --from Web1 --to JavaEE1
--order 20 --properties path=/aaa/:network-listener=http-listener-1 redirect3
```

```
asadmin create-relation --relationtype redirect --from Web1 --to JavaEE2
--order 10 --properties path=/:network-listener=http-listener-1 redirect4
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

2.7.2 delete-relation

The `delete-relation` subcommand deletes a relation between servers.

Synopsis

```
asadmin [asadmin_options] delete-relation relation_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-relation` subcommand deletes a relation between servers from the domain.

This subcommand is supported only in the remote mode.

Execution permission

General user

Precondition

The relation between the servers has been created.

Arguments

relation_name

Specifies the name of the relation between servers that is to be deleted.

Type: String

You can specify the following values:

- Name of an existing relation between servers.

Default value: (None. You must specify a value.)

Examples

The following example deletes the associated `redirect1` relation.

```
asadmin delete-relation redirect1
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

2.7.3 list-relations

The `list-relations` subcommand lists relations between servers.

Synopsis

```
asadmin [asadmin_options] list-relations [{--long={false|true} |
    --output output} [--header={false|true}]]
    [--relationtype relation_type_name] [--from server_name]
    [--to server_name] [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-relations` subcommand lists the relations between servers.

This subcommand is supported only in the remote mode.

Execution permission

General user

Arguments

`--long={false|true}`

Specifies whether to display detailed information, in a list, about the relation between the servers.

Type: Boolean

You can specify the following values:

- `true`

  Displays the relation name and its detailed information (relation type, dependent source server name, depended-on destination server name, and order number). If you specify `true` for this argument, you cannot specify the `--output` option.

- `false`

  Follows the specified `--output` and the `--header` options.

Default value: `false`

`--output` *`output`*

Specifies the items to be displayed in a list and their order, which is specified by a comma separated list. Uppercase and lowercase letters are not distinguished for the specified values.

The items to be displayed in the list are displayed from the left, in the order specified by this option. The list is sorted and displays the left most item. In order to sort by the character strings, if you specify `order` for the leftmost item, sorting is not in the ascending order of the numerical values.

If this option is omitted, the relation name, relation type, dependent source server name, and the depended-on destination server name are displayed. However, if the `--long` option is set to `true`, this specification is followed.

Type: String

You can specify the following values:

- `name`

  Relation name

- `type`

  Relation type name

- `from`

  Dependent source server name

- `to`

  Depended-on destination server name

- `order`

  Order number

Default value: (None)

`--header={false|true}`

Specifies whether to display the header information. If the `--long` option is set to `true`, this option is ignored.

Type: Boolean

You can specify the following values:

- `true`

  Displays the header information.

- `false`

  Does not display the header information.

Default value: (None)

`--relationtype` *relation_type_name*

Specifies the relation type to filter dependency relation information.

Type: String

You can specify the following values:

- `prf-relation`
  PRF server relation
- `redirect`
  Redirect relation

Default value: (None)

If this option is omitted, the dependency relation information of the all relation types is displayed.

`--from` *server_name*

Specify the name of dependent source server to filter dependency relation information.

Type: String

You can specify the following values:

- Name of a dependent source server existing in the domain.

Default value: (None)

If this option is omitted, the dependency relation information of the all dependent source servers is displayed.

`--to` *server_name*

Specify the name of depended-on destination server to filter dependency relation information.

Type: String

You can specify the following values:

- Name of a depended-on destination server existing in the domain.

Default value: (None)

If this option is omitted, the dependency relation information of all the depended-on destination servers is displayed.

target

Specifies the name of the object for which information is displayed.

Type: String

You can specify the following values:

- `domain`
  Displays information about all the dependency relations in a domain.
- *relation_name*
  Displays information about the specified dependency relation.

Default value: `domain`

Output Format

The first line is the header of the output information and displays a fixed character string, which indicates the output format.

Between columns, the longest character string within a column is separated from the beginning of the next column by two spaces.

```
NAME    TYPE      FROM          TO            ORDER
relation_name  relation_type_name  dependent_source_server_name  depended-
on_destination_server_name  order_number
```

relation_name: Displays the relation name.

relation_type_name: Displays the relation type name.

dependent_source_server_name: Displays the name of the dependent source server.

depended-on_destination_server_name: Displays the name of the depended-on destination server.

order_number: Displays the relation order number.

Examples

The following example displays the relation between all servers.

```
asadmin list-relations
```

Example of output:

```
relation1  prf-relation       JavaEE1       PRF1
relation2  prf-relation       Web1          PRF1
relation3  redirect           Web1          JavaEE1
relation4  redirect           Web1          JavaEE2
relation5  ex_relation_type1  custom_type1  Web1
```

The following example displays the relation between all the servers, with the header.

```
asadmin list-relations --long=true
```

Example of output:

```
NAME       TYPE               FROM          TO       ORDER
relation1  prf-relation       JavaEE1       PRF1     0
relation2  prf-relation       Web1          PRF1     0
relation3  redirect           Web1          JavaEE1  20
relation4  redirect           Web1          JavaEE2  10
relation5  ex_relation_type1  custom_type1  Web1     0
```

The following example displays the relation between all the servers, without the header.

```
asadmin list-relations --long=true --header=false
```

Example of output:

```
relation1  prf-relation       JavaEE1       PRF1     0
relation2  prf-relation       Web1          PRF1     0
relation3  redirect           Web1          JavaEE1  20
relation4  redirect           Web1          JavaEE2  10
relation5  ex_relation_type1  custom_type1  Web1     0
```

The following example displays only the relation of the relation type `redirect`.

```
asadmin list-relations --relationtype  redirect
```

Example of output:

```
relation3  redirect  Web1  JavaEE1
relation4  redirect  Web1  JavaEE2
```

The following example displays only the relation of the dependent source server named `Web1`.

```
asadmin list-relations --from Web1
```

Example of output:

```
relation2  prf-relation  Web1  PRF1
relation3  redirect      Web1  JavaEE1
relation4  redirect      Web1  JavaEE2
```

The following example displays only the relation of the depended-on destination server named `PRF1`.

```
asadmin list-relations --to PRF1
```

Example of output:

```
relation1  prf-relation  JavaEE1  PRF1
relation2  prf-relation  Web1     PRF1
```

The following example displays only the relation of the relation name `relation3`.

```
asadmin list-relations relation3
```

Example of output:

```
relation3  redirect  Web1  JavaEE1
```

The following example displays only the order numbers, relation names, and depended-on destination servers of the dependency relations whose dependent source server is `Web1` and dependency relation type is `redirect`, sorted by order number.

```
asadmin list-relations  --relationtype  redirect  --from Web1
--output order,name,to
```

Example of output:

```
10    relation4  JavaEE2
20    relation3  JavaEE1
```

Exit Status

Exit Status	Explanation
0	Normal termination.

Exit Status	Explanation
1	Abnormal termination.

2.8 Commands used for application administration

This section describes the syntax and functionality of the commands used for application administration.

2.8.1 create-application-ref

Creates a reference from a clustered or an unclustered server instance to a previously deployed application element (for example, a Java EE application, a Web module, or an enterprise bean module).

Synopsis

```
asadmin [asadmin-options] create-application-ref [--help]
    [--target target] [--virtualservers virtual_servers]
    [--enabled={true|false}] reference_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-application-ref` subcommand creates a reference from a cluster or an unclustered server instance to a previously deployed application element. For example:

- A Java EE application
- A Web module
- An enterprise bean module

This effectively results in the application element being deployed and made available on the targeted instance or cluster. The target instance or instances making up the cluster need not be running or available for this command to succeed. If one or more instances are not available, they will receive the new application element the next time they start.

This command is supported in remote mode only.

Precondition

- DAS has to be in a running state.
- The application or module has to be pre-deployed. To complete the predeployment task a reference is created using this command.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target for which you are creating the application reference.

Type: String

The following values can be specified:

- `server`

Specifies the default server instance as the target for creating the application reference. Server is the name of the default server instance. This is the default value for this option.

- *cluster_name*

  Specifies a particular cluster as the target for creating the application reference.

- *instance_name*

  instance_name: Specifies a particular stand-alone server instance as the target for creating the application reference.

Default value: `server`

`--virtualservers` *virtual_servers*

Specifies a comma-separated list of virtual server IDs on which to deploy. This option applies only to Web modules (either standalone or in a Java EE application). If this option is not specified, the application is deployed to all virtual servers except the administrative server, `__asadmin`.

Type: String

The following values can be specified:

- *Name of specified comma-separated list of virtual server IDs*

Default value: N/A

`--enabled={true|false}`

Indicates whether the application should be enabled (that is, loaded). This value will take effect only if the application is enabled at the global level. The default is `true`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

reference_name

Specifies the name of the application or module. This can include the following:

- *JavaEE application*
- *Web module*
- *EJB module*
- *Connector module*
- *Application client module*
- *Lifecycle module*

The name can include an optional version identifier, which follows the name and is separated from the name by a colon (**:**).

The value for reference_name must meet the following conditions:

The version identifier only contains alphanumeric characters, underscores (_), hyphens (-), and periods (**.**).

The first character is an alphanumeric character.

Type: String

The following values can be specified:

- *Name of an application or a module whose reference is to be created*

Default value: N/A

Examples

The following example creates a reference to the Web module `MyWebApp` on the unclustered server instance `NewServer`.

```
asadmin create-application-ref --target NewServer MyWebApp
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.8.2 delete-application-ref

Removes the reference to an application.

Synopsis

```
asadmin [asadmin-options] delete-application-ref [--help]
    [--target target] [--cascade={true|false}] reference_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-application-ref` subcommand removes a reference from a cluster or a nonclustered server instance to an application. This effectively results in the application element being removed and no longer available on the targeted instance or the cluster. The target instance or instances making up the cluster need not be running or available for this subcommand to succeed. If one or more instances are not available, they will no longer load the application the next time they start. Removal of the reference does not result in removal of the application from the domain. The bits are removed only by the `undeploy` subcommand. This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target from which you are removing the application reference.

Type: String

The following values can be specified:

- `server`

Specifies the default server instance as the target. The server is the name of the default server instance and is the default value.

- *cluster-name*
  Specifies a certain cluster as the target.
- *instance_name*
  Specifies a certain stand-alone server instance as the target.

Default value: `server`

`--cascade={false|true}`

Indicates whether the resources dependent on the module should also be recursively deleted for a connector module.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`reference_name`

Specifies the name of the application or module that can be one of the following:

- Java EE application module
- Web module
- EJB module
- Connector module
- Application client module
- Lifecycle module

Type: String

The following values can be specified:

- *Name of the application or module*

Default value: N/A

Examples

The following example removes a reference to the Web module `MyWebApp` from the nonclustered server instance `NewServer`.

```
asadmin delete-application-ref --target NewServer MyWebApp
```

Exit Status

Exit Status	Explanation
`0`	command executed successfully.
`1`	error in executing the command.

2.8.3 deploy

Deploys the specified component.

Synopsis

```
asadmin [asadmin-options] deploy [--help]
    [--force={false|true}]
    [--virtualservers virtual_servers]
    [--contextroot context_root]
    [--precompilejsp={false|true}]
    [--name component_name]
    [--upload={true|false}]
    [--retrieve local_dirpath]
    [--deploymentplan deployment_plan]
    [--altdd alternate_deploymentdescriptor]
    [--runtimealtdd runtime_alternate_deploymentdescriptor]
    [--deploymentorder deployment_order]
    [--enabled={true|false}]
    [--generatermistubs={false|true}]
    [--availabilityenabled={false|true}]
    [--libraries jar_file[,jar_file]...]
    [--target target]
    [--type pkg-type]
    [--properties name=value[:name=value]...]
    file_archive|filepath
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `deploy` subcommand deploys applications to the server. Applications can be enterprise applications, web applications, Enterprise JavaBeans (EJB) modules, connector modules, and application client modules. If the component is already deployed or already exists, it is forcibly redeployed if the `--force` option is set to `true` (the default value is `false`).

If OutOfMemoryError occurs in the Domain Administration Server (DAS) during application deployment, the error may be caused by insufficient Java heap of the DAS. When specifying the size of Java heap for the DAS, consider the size of the application archive to be deployed.

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Files

- *file_archive*|*filepath* operand takes an input file which will be deployed by the command.
- Use of `--retrieve` option generates the client stub JAR files in the file path specifies by `--retrieve` option.
- `--deploymentplan` option takes an input JAR file which contains Java EE RI DD.
- `--altdd` option takes an input JAR file which contains Java EE standard DD.
- `--runtimealtdd` option takes an input JAR file which contains Java EE RI DD.

- Use of `--generatermistubs` option generates static RMI-IIOP stubs and put into the `client.jar`.
- `--libraries` option takes input library JAR files which are made available for the deployed application.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--force={false|true}`

Specifies whether to redeploy the component even if the specified component has already been deployed or already exists.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--virtualservers` *`virtual_servers`*

Specifies one or more virtual server IDs.

Type: String

The following values can be specified:

- *One or more virtual server IDs*

Default value: N/A

`--contextroot` *`context_root`*

Set the context root of the specified application. Valid only if the archive is a web module.

Type: String

The following values can be specified:

- *Name of the context root for the web module*

Default value: File name without extension having archive as web module.

`--precompilejsp={false|true}`

Specifies whether to allow the JSP to be precompiled during deployment.

When there is a compilation error, the file name of the cause may not be displayed depending on the contents of the error. In this case, compile the .jsp file individually using the `jspc` command, to identify the file where the error occurred.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--name` *`component_name`*

Specifies the name of the deployable component.

The name can include an optional version identifier, which follows the name and is separated from the name by a colon (**:**). The version identifier must begin with a letter or number. It can contain alphanumeric characters, underscore (_), hyphen (-), and period (**.**).

Type: String

The following values can be specified:

- *Name of the deployable component*

Default value: N/A

`--upload={true|false}`

Specifies whether the subcommand uploads the file to the DAS.

Type: Boolean

The following values can be specified:

If a directory *filepath* is specified, this option is ignored.

- `false`

  The subcommand does not upload the file and attempts to access the file through the specified file name. If the DAS cannot access the file, then the subcommand fails. For example, the DAS might be running as a different user than the administration user and does not have read access to the file. In this situation, the subcommand fails if the `--upload` option is set to `false`.

- `true`

  The subcommand uploads the file to the DAS over the network connection.

Default value:

- If the DAS is on the host where the subcommand is run, the default is `false`.
- If the DAS is on a remote host, the default is `true`.

`--retrieve` *`local_dirpath`*

Retrieves the client stub JAR file from the server machine to the local directory.

Type: String

The following values can be specified:

- *File path where client JAR file are stored*

Default value: N/A

`--deploymentplan` *`deployment_plan`*

Deploys the deployment plan, which is a JAR file that contains Java EE RI DD.

Type: String

The following values can be specified:

- *Name of the JAR file*

Default value: N/A

`--altdd` *`alternate_deploymentdescriptor`*

Deploys the application using a Java EE standard DD that resides outside of the application archive.

The alternate DD overrides the top-level DD packaged in the archive. For example, for an EAR, the `--altdd` option overrides the `application.xml` file. For a standalone module, the `--altdd` option overrides the top-level module descriptor such as `web.xml`.

Type: String

The following values can be specified:

- *Path to the DD file*

Default value: N/A

`--runtimealtdd` *`runtime_alternate_deploymentdescriptor`*

Deploys the application using Java EE RI DD that reside outside the application archive.

The alternate DD overrides the top-level DD packaged in the archive. For a standalone module, the `--runtimealtdd` option overrides the top-level module descriptor such as `glassfish-web.xml`. This applies to Java EE Server DD only (`glassfish-*.xml`); the name of the alternate DD file must begin with `glassfish-`. This does not apply to `sun-*.xml` DD, which are deprecated.

Type: String

The following values can be specified:

- Specify an absolute path or a relative path to the alternate DD file.

Default value: N/A

`--deploymentorder` *`deployment_order`*

Specifies the deployment order of the application.

Applications with lower numbers are loaded before applications with higher numbers. If two applications have the same deployment order, the first application to be deployed is the first application to be loaded at server startup.

Type: Integer

The following values can be specified:

- `1` to `2147483647`

Default value: `100`

`--enabled={true|false}`

Allows users to access the application.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--generatermistubs={true|false}`

Specifies whether to generate the stubs.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--availabilityenabled={true|false}`

Specifies whether high-availability is enabled for web sessions and for stateful session bean (SFSB) checkpointing and potentially passivation.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--libraries` *`jar_file`*

Specifies a comma-separated list of library JAR files. Specify relative paths relative to *domain-dir*`/lib/applibs`.

Type: String

The following values can be specified:

- *A comma-separated list of library JAR files*

Default value: N/A

`--target` *`target`*

Specifies the target to which you are deploying.

Type: String

The following values can be specified:

- `server`

  Deploys the component to the default server instance "`server`" and is the default value.

- `domain`

  Deploys the component to the domain. If domain is the target for an initial deployment, the application is deployed to the domain, but no server instances or clusters reference the application. If domain is the target for a redeployment (the `--force` option is set to `true`), and dynamic reconfiguration is enabled for the clusters or server instances that reference the application, the referencing clusters or server instances automatically get the new version of the application. If redeploying, and dynamic configuration is disabled, the referencing clusters or server instances do not get the new version of the application until the clustered or standalone server instances are restarted.

- *cluster_name*

  Deploys the component to every server instance in the cluster.

- *instance_name*

  Deploys the component to a particular stand-alone sever instance.

Default value: `server`

`--type` *`pkg-type`*

Specifies the packaging archive type of the component that is being deployed.

Type: String

The following values can be specified:

- `car`

  The component is packaged as a CAR file.

- `ear`

  The component is packaged as an EAR file.

- `ejb`

  The component is an EJB packaged as a JAR file.

- `rar`

  The component is packaged as a RAR file.

- `war`

  The component is packaged as a WAR file.

Default value: N/A

`--properties` *`name=value`* `|` `--property` *`name=value`*

Optional keyword-value pairs that specify additional properties for the deployment.

Type: String

The following values can be specified:

- `keepSessions={false|true}`

  If the `--force` option is set to `true`, this property can be used to specify whether active sessions of the application that is being redeployed are preserved and then restored when the redeployment is complete. Applies to HTTP sessions in a web container. Default is `false`.

  Type: Boolean

  Default value: `false`

  Range Value:

  `false`

  Active sessions of the application are not preserved and restored (default).

  `true`

  Active sessions of the application are preserved and restored.

  If any active session of the application fails to be preserved or restored, none of the sessions will be available when the redeployment is complete. However, the redeployment continues and a warning is logged. To preserve active sessions, Java EE Server serializes the sessions and saves them in memory. To restore the sessions, the class loader of the newly redeployed application deserializes any sessions that were previously saved.

- `preserveAppScopedResources={false|true}`

  If set to `true`, preserves any application-scoped resources and restores them during redeployment. Default is `false`.

  Type: Boolean

  Default value: `false`

  Range Value:

  `true`

  `false`

  Other available properties are determined by the implementation of the component that is being redeployed.

Default value: N/A

`file_archive|filepath`

Specifies the path to the archive that contains the application that is being deployed.

Type: String

The following values can be specified:

- *The path to the archive that contains the application*

Default value: N/A

Examples

The following example deploys the enterprise application packaged in the `Cart.ear` file to the server instance `instance1`. You can use the `--target` option to deploy to a different server instance or to a cluster.

```
asadmin deploy --target instance1 Cart.ear
```

Exit Status

Exit Status	Explanation
0	command executed successfully.

Exit Status	Explanation
1	error in executing the command.

2.8.4 disable

Disables the component.

Synopsis

```
asadmin [asadmin-options] disable [--help]
    [--target target] component_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `disable` subcommand immediately disables the specified deployed component. If the component has not been deployed, an error message is returned.

This command communicates with servers to disable the components remotely.

The timeout can be set using the `AS_ADMIN_READ_TIMEOUT` parameter available in the `asenv` file.

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target on which you are disabling the component.

Type: String

The following values can be specified:

- `server`
  Disables the component on the default server instance `server` and is the default value.
- *domain_name*
  Disables the component on the named domain.
- *cluster_name*
  Disables the component on every server instance in the cluster.
- *instance_name*
  Disables the component on a particular clustered or stand-alone server instance.

Default value: `server`

component_name

Specifies the name of the component to be disabled.

Type: String

The following values can be specified:

- *Name of the component*

Default value: N/A

Examples

The following example disables the deployed component `sampleApp` on the server instance `instance1`.

```
asadmin disable --target instance1 sampleApp
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

Notes

Removing a component can release the instance from the memory. However, if the component is disabled (by running the `disable` command), the instance is not released. Note that when many components are disabled, memory in use increases and OutOfMemoryError may occur.

2.8.5 enable

Enables the component.

Synopsis

```
asadmin [asadmin-options] enable [--help]
    [--target target] component_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `enable` subcommand enables the specified deployed component. If the component is already enabled, then it is re-enabled. If it has not been deployed, then an error message is returned. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target on which you are enabling the component.

Type: String

The following values can be specified:

- `server`
  Enables the default server instance `server` and is the default value.
- *domain_name*
  Enables the named domain.
- *cluster_name*
  Enables every server instance in the cluster.
- *instance_name*
  Enables a particular clustered or stand-alone server instance.

Default value: `server`

component_name

Specifies the name that can include an optional version identifier, which follows the name and is separated from the name by a colon (`:`). The version identifier must begin with a letter or number. It can contain alphanumeric characters plus underscore (_), dash (-), and period (`.`) characters.

Type: String

The following values can be specified:

- *Name of the component to be enabled*

Default value: N/A

Examples

The following enables the disabled component, `sampleApp` on the server instance `instance1`.

```
asadmin enable --target instance1 sampleApp
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.8.6 get-client-stubs

Retrieves the application JAR files needed to launch the application client.

Synopsis

```
asadmin [asadmin-options] get-client-stubs [--help]
    --appname application_name
    local_directory_path
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `get-client-stubs` subcommand copies the required JAR files for an AppClient standalone module or each AppClient module in an application from the server machine to the local directory. Each client's generated JAR file is retrieved, along with any required supporting JAR files. The client JAR file name is of the form *app-name*`Client.jar`. This subcommand is supported in remote mode only.

Precondition

- Domain Administration Server (DAS) is running.
- For the operand below are the pre-conditions:
  Before executing the `get-client-stubs` subcommand, you must deploy the application or module.

Files

The `get-client-stubs` subcommand copies the required JAR files for an AppClient standalone module or each AppClient module in an application from the server machine to the local directory.

The client JAR file name is of the form *app-name*`Client.jar`.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--appname` *application_name*

Specifies the name of the application or stand-alone client module.

The name can include an optional version identifier, which follows the name and is separated from the name by a colon (`:`). The version identifier must begin with a letter or number. It can contain alphanumeric characters plus underscore (_), dash (-), and period (`.`) characters.

Type: String

The following values can be specified:

- *Name of the application*

Default value: N/A

local_directory_path

Specifies the path to the local directory where the client stub JAR file should be stored.

Type: String

The following values can be specified:

- *Path of the directory*

Default value: N/A

Examples

The following example gets the client stubs for the specified deployed application.

```
asadmin get-client-stubs --appname myapplication
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.8.7 list-application-refs

Lists the existing application references.

Synopsis

```
asadmin [asadmin-options] list-application-refs [--help]
    [--long={false|true}] [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-application-refs` subcommand lists all the application references in a cluster or an unclustered server instance. This effectively lists all the modules deployed on the specified target. If multiple versions of a module or application are deployed, this subcommand lists all the versions. This subcommand is supported in remote mode only.

The target instance or instances making up the cluster need not be running or available for this subcommand to succeed.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--long={false|true}`

Displays whether each module or application listed is enabled, if this value is set to `true`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

target

Specifies the target for which you are listing the application references.

Type: String

The following values can be specified:

- `server`

  Specifies the default server instance as the target. Server is the name of the default `server` instance. This is the default value.

- *cluster_name*

  Specifies a certain cluster as the target.

- *instance_name*

  Specifies a certain server instance as the target.

Default value: `server`

Examples

The following example lists the application references for the unclustered server instance `NewServer`.

```
asadmin list-application-refs NewServer
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.8.8 list-applications

Lists the deployed applications.

Synopsis

```
asadmin [asadmin-options] list-applications [--help]
    [--long={false|true}] [--resources] [--subcomponents]
    [--type type] [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-applications` subcommand lists the deployed Java EE applications and the type of each application that is listed. If multiple versions of a module or application are deployed, this subcommand lists all the versions. This subcommand is supported in remote mode only.

If the `--type` option is not specified, all the applications are listed. If the `--type` option is specified, you must specify a type.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--long={false|true}`

Displays whether each module or application listed is enabled, if this value is set to `true`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--resources`

Lists the application-scoped resources for each application.

If the `--subcomponents` option is also used, this subcommand lists the application-scoped resources for each component within the application.

Type: N/A

Default value: N/A

`--subcomponents`

Lists the subcomponents of each application.

The subcomponents listed depend on the application type. For example, for a Java EE application (EAR file), modules are listed. For a web application, servlets and JSP pages are listed. For an EJB module, EJB subcomponents are listed.

Type: N/A

Default value: N/A

`--type` *`type`*

Specifies the type of the applications that are to be listed.

Type: String

The following values can be specified:

- `application`
- `appclient`
- `connector`
- `ejb`
- `web`
- `webservice`

Default value: *If no type is specified, all applications are listed*

`target`

Specifies the name of the target upon which the subcommand operates.

Type: String

The following values can be specified:

- `server`
  Lists the applications for the default server instance `server` and is the default value.
- `domain`
  Lists the applications for the domain.
- *cluster_name*
  Lists the applications for the cluster.
- *instance_name*
  Lists the applications for a particular stand-alone server instance.

Default value: `server`

Examples

The following example lists the Web Applications.

```
asadmin list-applications --type web
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

Notes

As a result of running this subcommand, it displays the GUI and online help that Java EE Server provides. Do not perform operations such as undeploying applications because these are necessary for operating the product. When you undeploy `__internal_admin`, you cannot access the Administration Console. To recover, execute the following subcommand from the Command Prompt.

```
asadmin deploy --name __internal_admin --contextroot
admin "Application Server installation directory\javaee\glassfish\admin\admin"
```

When you undeploy `__internal_manual`, it does not display the online help and message ID in the Administration Console. To recover, execute the following subcommand from the Command Prompt.

```
asadmin deploy --name __internal_manual --contextroot
manual "Application Server installation directory\common\docs"
```

2.8.9 list-containers

Lists the application containers.

Synopsis

```
asadmin [asadmin-options] list-containers [--help]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-containers` subcommand displays a list of application containers. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

Examples

This example lists the current application containers.

```
asadmin list-containers
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.8.10 list-modules

Lists Java EE Server modules.

Synopsis

```
asadmin [asadmin-options] list-modules [--help]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-modules` subcommand of asadmin displays a list of modules that are accessible to Java EE Server module subsystem. The version of each module is displayed. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

Examples

The following example provides a partial listing of modules that are accessible to Java EE Server module subsystem.

```
asadmin list-modules
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.8.11 list-sub-components

Lists EJB or servlet components either in a deployed module or in the module of a deployed application.

Synopsis

```
asadmin [asadmin-options] list-sub-components [--help]
    [--type type]
    [--appname appname] [--resources]
    modulename
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-sub-components` subcommand of asadmin lists the EJB or servlet components in a deployed module or in a module of a deployed application. If a module is not specified, then all the modules are listed. The `--appname` option functions only when the specified module is stand-alone. This subcommand is supported in remote mode only.

To display a specific module in an application, you must specify the module name with the `--appname` option.

Precondition

- Domain Administration Server (DAS) is running.
- For `--appname` option, the `--appname` option functions only when the given module is standalone.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--type` *type*

Specifies the type of component to be listed. The options are `ejbs` and `servlets`. If nothing is specified, then all of the components are listed.

Type: String

The following values can be specified:

- `ejb`
- `servlet`

Default value: *All*

`--appname` *appname*

Indicates the name of the application.

- This option is required when the desired output is the subcomponent of an embedded module of a deployed application.
- The name can include an optional version identifier, which follows the name and is separated from the name by a colon (`:`). The version identifier must begin with a letter or number. It can contain alphanumeric characters with underscore (_), dash (-), and period (`.`) characters.

Type: String

The following values can be specified:

- *Name of the application*

Default value: N/A

`--resources`

Lists the application-scoped resources for each subcomponent.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

modulename

Specifies the name of the module containing the subcomponent.

Type: String

The following values can be specified:

- *Name of the module*

Default value: N/A

Examples

The following example lists the subcomponents of the `MEjbApp` application within the `mejb.jar` module.

```
asadmin list-sub-components --appname MEjbApp mejb.jar
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.

Exit Status	Explanation
1	error in executing the subcommand.

2.8.12 list-timers

Lists all the persistent timers owned by the server instance(s).

Synopsis

```
asadmin [asadmin-options] list-timers [--help] [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-timers` subcommand of asadmin lists the persistent timers owned by a specific server instance or a cluster of server instances. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

target

Specifies the target either as a standalone server instance or a cluster. If the target is a stand-alone instance, then the number of timers owned by the instance is listed. If the target is a cluster, then the number of timers owned by each instance in the cluster is listed.

Type: String

The following values can be specified:

- `server`
- *cluster_name*
- *standalone_instance_name*

Default value: `server`

Examples

The following example lists persistent timers in a particular standalone server instance.

```
asadmin list-timers server
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.8.13 list-web-context-param

Lists the servlet context-initialization parameters of a deployed Web application or module.

Synopsis

```
asadmin [asadmin-options] list-web-context-param [--help]
    [--name=context-param-name] application-name[/module]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-web-context-param` subcommand of asadmin lists the servlet context-initialization parameters of one of the following items:

- A deployed Web application
- A Web module in a deployed Java Platform, Enterprise Edition (Java EE) application

The `list-web-context-param` command lists only those parameters that have previously been set using the `set-web-context-param` command. The command does not list parameters that are set only in the application's DD.

Precondition

- Domain Administration Server (DAS) is running.
- The application must already be deployed. Else, an error occurs.

Files

The path to the module is specified in the `module` element of the application's `application.xml` file.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--name=`*`context-param-name`*

Specifies the name of the servlet `context-initialization` parameter that is to be listed.

Type: String

The following values can be specified:

- *Name of the servlet context-initialization parameter*

Default value: *All parameters of the application that have previously been set are listed.*

application-name/module

Specifies the name of the application. This name can be obtained by using the `list-applications` subcommand.

The *module* is required only if the servlet context-initialization parameter applies to a Web module of a Java EE application.

For the option Module must follow *application-name*, separated by a slash (`/`).

Type: String

The following values can be specified:

- *application-name*
- *module*

Default value: N/A

Examples

The following example lists all servlet context-initialization parameters of the web application `basic-ezcomp` that have been set by using the `set-web-context-param` subcommand.

```
asadmin list-web-context-param basic-ezcomp
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.8.14 list-web-env-entry

Lists the environment entries for a deployed Web application or module.

Synopsis

```
asadmin [asadmin-options] list-web-env-entry [--help]
    [--name=env-entry-name] application-name[/module]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-web-env-entry` subcommand of `asadmin` lists the environment entries for one of the following items:

- A deployed Web application
- A Web module in a deployed Java Platform, Enterprise Edition (Java EE) application

The `list-web-env-entry` command lists only those parameters that have previously been set using the `set-web-env-entry` command. The command does not list parameters that are set only in the application's DD.

Precondition

- Domain Administration Server (DAS) is running.
- For the option `application` must already be deployed. Otherwise, an error occurs.
- For the option *module* must follow *application-name*, separated by a slash (`/`).

Files

The path to the `module` is specified in the module element of the application's `application.xml` file.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--name=`*`env-entry-name`*

Specifies the name of the environment entry that is to be listed.

Type: String

The following values can be specified:

- *Name of the environment entry*

Default value: *All environment entries that have previously been set for the application are listed.*

`application-name/module`

Specifies the name of the application. This name can be obtained by using the `list-applications` subcommand.

- The *module* is required only if the environment entry applies to a Web module of a Java EE application.
- The *module* would be specified as the operand of this command as `myApp/myWebModule.war`.

Type: String

The following values can be specified:

- *application-name*
- *module*

Default value: N/A

Examples

The following example lists all environment entries that have been set for the web application `hello` by using the `set-web-env-entry` subcommand.

```
asadmin list-web-env-entry hello
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.

Exit Status	Explanation
1	error in executing the subcommand.

2.8.15 redeploy

Redeploys the specified component.

Synopsis

```
asadmin [asadmin-options] redeploy [--help]
    --name component_name
    [--upload={true|false}]
    [--retrieve local_dirpath]
    [--deploymentplan deployment_plan]
    [--altdd alternate_deploymentdescriptor]
    [--runtimealtdd runtime_alternate_deploymentdescriptor]
    [--deploymentorder deployment_order]
    [--enabled={true|false}]
    [--generatermistubs={false|true}]
    [--contextroot context_root]
    [--precompilejsp={true|false}]
    [--virtualservers virtual_servers]
    [--libraries jar_file[,jar_file]...]
    [--target target]
    [--type pkg-type]
    [--properties name=value[:name=value]...]
    file_archive|filepath
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `redeploy` subcommand of `asadmin` redeploys an enterprise application, Web application, module based on the Enterprise JavaBeans (EJB) specification (EJB module), connector module, or application client module that is already deployed or already exists. The `redeploy` command preserves the settings and other options with which the application was originally deployed. This subcommand is supported in remote mode only.

If redeploying, and dynamic configuration is disabled, the referencing clusters or server instances do not get the new version of the application until the clustered or standalone server instances are restarted.

If OutOfMemoryError occurs on the Domain Administration Server (DAS) during application deployment, the error might be caused by insufficient Java heap of the domain administration server. When specifying the size of Java heap for the DAS, take into account the size of the application archive to be deployed.

Precondition

Domain Administration Server (DAS) is running.

Files

- *file_archive|filepath* operand takes an input file which will be deployed by the command.
- use of `--retrieve` option generates the client stub JAR files in the file path specifies by `--retrieve` option.

- `--deploymentplan` option takes an input JAR file which contains Java EE RI DD.
- `--altdd` option takes an input JAR file which contains Java EE standard DD.
- `--runtimealtdd` option takes an input JAR file which contains Java EE RI DD.
- Use of `--generatermistubs` option generates static RMI-IIOP stubs and put into the `client.jar`.
- `--libraries` option takes input library JAR files which are made available for the deployed application.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--virtualservers` *`virtual_servers`*

Specifies one or more virtual server IDs. Multiple IDs are separated by commas.

Type: String

The following values can be specified:

- *Specify virtual server IDs separated by comma*

Default value: N/A

`--contextroot` *`context_root`*

Specifies the context root of an application. This is enabled only when the archive is a Web module.

Type: String

The following values can be specified:

- *Name of the context root for the web module*

Default value: File name without extension having archive as Web module.

`--precompilejsp={true|false}`

Prevents the JSP to be precompiled during deployment. Instead, JSPs are compiled during runtime. Default is `false`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--name` *`component_name`*

Specifies the name of the deployable component.

The name can include an optional version identifier, which follows the name and is separated from the name by a colon (`:`). The version identifier must begin with a letter or number. It can contain alphanumeric characters, underscore (_), dash (-), and period (`.`) characters.

Type: String

The following values can be specified:

- *Name of the deployable component*

Default value: N/A

`--upload={true|false}`

Specifies whether the subcommand uploads the file to the DAS.

If a directory *filepath* is specified, this option is ignored.

Type: Boolean

The following values can be specified:

- `false`

  The subcommand does not upload the file and attempts to access the file through the specified file name. If the DAS cannot access the file, the subcommand fails.

  For example, the DAS might be running as a different user than the administration user and does not have read access to the file. In this situation, the subcommand fails if the `--upload` option is `false`.

- `true`

  The subcommand uploads the file to the DAS over the network connection.

Default value:

- If the DAS is on the host where the subcommand is run, the default is `false`.
- If the DAS is on a remote host, the default is `true`.

`--retrieve` *`local_dirpath`*

Retrieves the client stub JAR file from the server machine to the local directory.

Type: String

`--deploymentplan` *`deployment_plan`*

Deploys the deployment plan, which is a JAR file that contains Java EE RI DD.

Specify this option when deploying a pure EAR file. A pure EAR file is an EAR without Java EE Server descriptors.

`--altdd` *`alternate_deploymentdescriptor`*

Deploys the application using a Java EE standard DD that resides outside of the application archive.

The alternate DD overrides the top-level DD packaged in the archive. For example, for an EAR, the `--altdd` option overrides the `application.xml` file. For a standalone module, the `--altdd` option overrides the top-level module descriptor such as `web.xml`.

Type: String

The following values can be specified:

- *Path to the DD file*

Default value: N/A

`--runtimealtdd` *`runtime_alternate_deploymentdescriptor`*

Deploys the application using Java EE RI DD that resides outside of the application archive.

The alternate DD overrides the top-level DD packaged in the archive.

For a standalone module, the `--runtimealtdd` option overrides the top-level module descriptor such as `glassfish-web.xml`. Applies to Java EE Server DD only (*glassfish-*.xml*); the name of the alternate DD file must begin with `glassfish-`. Does not apply to *sun-*.xml* DD, which are deprecated.

Type: String

The following values can be specified:

- Specify an absolute path or a relative path to the alternate DD file. This relative path is *`Application Server installation directory`*`/javaee/glassfish/bin`

Default value: N/A

`--deploymentorder` *`deployment_order`*

Specifies the deployment order of the application. Applications with lower numbers are loaded before applications with higher numbers.

If two applications have the same deployment order, the first application to be deployed is the first application to be loaded at server startup.

Type: Integer

The following values can be specified:

- `1` to `2147483647`

Default value: `100`

`--enabled={true|false}`

Allows users to access the application.

If you deploy to the target domain, this option is ignored, since deploying to the domain doesn't deploy to a specific instance or cluster.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--generatermistubs={false|true}`

Generates static RMI-IIOP stubs (if set to `true`) and adds them to the `client.jar`. If set to `false`, the stubs are not generated. Default is `false`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--libraries` *`jar_file`*`[,`*`jar_file`*`]...`

Specifies a comma-separated list of library JAR files.

Type: String

The following values can be specified:

- *A comma-separated list of library JAR files*

Default value: N/A

`--target` *`target`*

Specifies the target to which you are deploying.

If domain is the target for an initial deployment, the application is deployed to the domain, but no server instances or clusters reference the application.

If redeploying, and dynamic configuration is disabled, the referencing clusters or server instances do not get the new version of the application until the clustered or standalone server instances are restarted.

The following values are valid:

Type: String

The following values can be specified:

- `server`

  Deploys the component to the default server instance `server` and is the default value.

- `domain`

Deploys the component to the domain. If domain is the target for an initial deployment, the application is deployed to the domain, but no server instances or clusters reference the application. If domain is the target for redeployment, and dynamic reconfiguration is enabled for the clusters or server instances that reference the application, the referencing clusters or server instances automatically get the new version of the application. If redeploying, and dynamic configuration is disabled, the referencing clusters or server instances do not get the new version of the application until the clustered or standalone server instances are restarted.

- *cluster_name*

  Deploys the component to every server instance in the cluster.

- *instance_name*

  Deploys the component to a particular stand-alone server instance.

Default value: `server`

`--type` *pkg-type*

Specifies the packaging archive type of the component that is being deployed.

Type: String

The following values can be specified:

- `car`

  The component is packaged as a CAR file.

- `ear`

  The component is packaged as an EAR file.

- `ejb`

  The component is an EJB packaged as a JAR file.

- `rar`

  The component is packaged as a RAR file.

- `war`

  The component is packaged as a WAR file.

Default value: N/A

`{--properties | --property}` *name*`=`*value*`[:`*name*`=`*value*`]...`

Specifies the optional keyword-value pairs that specify additional properties for deployment.

The available properties are determined by the implementation of the component that is being deployed or redeployed. The `--properties` option and the `--property` option are equivalent. You can use either of these options regardless of the number of properties that you specify.

Type: String

The following values can be specified:

- `keepSessions=`*value*

  This property can be used to specify whether active sessions of the application that is being redeployed are preserved and then restored when the redeployment is complete. Applies to HTTP sessions in a web container. Default is `false`.

  `false`

  Active sessions of the application are not preserved and restored (default).

  `true`

  Active sessions of the application are preserved and restored.

  If any active session of the application fails to be preserved or restored, none of the sessions will be available when the redeployment is complete. However, the redeployment continues and a warning is logged.

To preserve active sessions, Java EE Server serializes the sessions and saves them in memory. To restore the sessions, the class loader of the newly redeployed application deserializes any sessions that were previously saved.

Type: Boolean

Default value: `false`

Range Value: `true`/`false`

- `preserveAppScopedResources=`*`value`*

  If set to `true`, preserves any application-scoped resources and restores them during redeployment. Default is `false`.

  Other available properties are determined by the implementation of the component that is being redeployed.

  Type: Boolean

  Default value: `false`

  Range Value: `true`/`false`

Default value: N/A

`file_archive``|`*`filepath`*

Specifies the path to the archive that contains the application that is being redeployed. This path can be a relative path or an absolute path. This relative path is *`Application Server installation directory`*`/javaee/glassfish/bin`.

The operand can specify a directory or an archive file.

Type: String

The following values can be specified:

- *Path to the archive*

Default value: N/A

Examples

The following example redeploys the Web application `hellodir`. The application was originally deployed from a directory on the server instance `instance1`.

```
asadmin redeploy --target instance1 --name hellodir
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.8.16 set-web-context-param

Sets a servlet `context-initialization` parameter of a deployed Web application or module.

Synopsis

```
asadmin [asadmin-options] set-web-context-param
    [--help] --name=context-param-name
```

```
    {--value=value|--ignoredescriptoritem={false|true}}
    [--description description] application-name[/module]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `set-web-context-param` subcommand of `asadmin` sets a servlet `context-initialization` parameter of one of the following items:

- A deployed Web application.
- A Web module in a deployed Java Platform, Enterprise Edition (Java EE) application.

This subcommand enables you to change the configuration of a deployed application without the need to modify the application's DD and repackage and redeploy the application.

This subcommand is supported in remote mode only.

The DAS or server instance must be restarted to set a servlet context-initialization parameter of the deployed application.

Precondition

- Domain Administration Server (DAS) is running.
- The application must already be deployed.

Files

The path to the module is specified in the `module` element of the application's `application.xml` file.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--name=`*`context-param-name`*

Specifies the name of the servlet `context-initialization` parameter that is to be set.

Type: String

The following values can be specified:

- *Name of the parameter*

Default value: N/A

`--value=`*`value`*

Specifies the value to which the servlet `context-initialization` parameter is to be set.

Either the `--value` option or the `--ignoredescriptoritem` option must be set.

Type: String

The following values can be specified:

- *Value of the parameter*

Default value: N/A

`--ignoredescriptoritem={false|true}`

Specifies whether the servlet `context-initialization` parameter is ignored if it is set in the application's DD. When a parameter is ignored, the application behaves as if the parameter had never been set in the application's DD. The behavior of an application in this situation depends on the application.

Type: Boolean

The following values can be specified:

- `false`

  The value is not ignored. This is the default value.

- `true`

  The value is ignored. Either the `--value` option or the `--ignoredescriptoritem` option must be set.

Do not use the `--ignoredescriptoritem` option to unset a servlet `context-initialization` parameter that has previously been set by using the `set-web-context-param` subcommand. Instead, use the `unset-web-context-param` subcommand for this purpose.

Default value: `false`

`--description` *`description`*

Specifies an optional textual description of the context parameter that is being set.

Type: String

The following values can be specified:

- *Description of the environment entry*

Default value: N/A

`application-name`

Specifies the name of the application. This name can be obtained by using the `list-applications` subcommand.

Type: String

The following values can be specified:

- *Name of the application*

Default value: N/A

`module`

Specifies the relative path to the module within the application's enterprise archive (EAR) file. The path to the module is specified in the `module` element of the application's `application.xml` file. `module` is required only if the servlet `context-initialization` parameter applies to a Web module of a Java EE application. The module would be specified as the operand of this command as `myApp/myWebModule.war`.

The module must follow `application-name`, separated by a slash (`/`).

Type: String

The following values can be specified:

- *Name of the module in the application*

Default value: N/A

Examples

The following example sets the servlet `context-initialization` parameter `javax.faces.STATE_SAVING_METHOD` of the web application `basic-ezcomp` to `client`. The description `The location where the application's state is preserved` is provided for this parameter.

```
asadmin set-web-context-param --name=javax.faces.STATE_SAVING_METHOD
--description "The location where the application's state is preserved"
--value=client basic-ezcomp
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.8.17 set-web-env-entry

Sets an environment entry for a deployed Web application or module.

Synopsis

```
asadmin [asadmin-options] set-web-env-entry [--help]
    --name=env-entry-name --type=env-entry-type
    {--value=value|--ignoredescriptoritem={true|false}}
    [--description description] application-name[/module]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `set-web-env-entry` subcommand of `asadmin` sets an environment entry for one of the following items:

- A deployed Web application.
- A Web module in a deployed Java Platform, Enterprise Edition (Java EE) application.

This subcommand enables you to change the configuration of a deployed application without the need to modify the application's DD and repackage and redeploy the application.

This subcommand is supported in remote mode only.

The DAS or server instance must be restarted to set an environment parameter entry for the deployed application.

Precondition

- Domain Administration Server (DAS) is running.
- The application must already be deployed.

Files

The path to the module is specified in the `module` element of the application's `application.xml` file.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--name=`*`env-entry-name`*

Specifies the name of the environment entry that is to be set. The name is a JNDI name relative to the `java:comp/env` context. The name must be unique within a deployment component.

Type: String

The following values can be specified:

- *Name of the environment entry*

Default value: N/A

`--type=`*`env-entry-type`*

Specifies the fully-qualified Java type of the environment entry value that is expected by the application's code.

Type: String

The following values can be specified:

- `java.lang.Boolean`
- `java.lang.Byte`
- `java.lang.Character`
- `java.lang.Double`
- `java.lang.Float`
- `java.lang.Integer`
- `java.lang.Long`
- `java.lang.Short`
- `java.lang.String`

Default value: N/A

`--value=`*`value`*

Specifies the value to which the environment entry is to be set.

- If the `--type` is `java.lang.Character`, the value must be a single character. Else, the value must be a string that is valid for the constructor of the specified type.
- Either the `--value` option or the `--ignoredescriptoritem` option must be set.

Type: String

The following values can be specified:

- *Value of the environment entry*

Default value: N/A

`--ignoredescriptoritem={true|false}`

Specifies whether the environment entry is ignored if it is set in the application's DD.

Type: Boolean

The following values can be specified:

- `false`

  The value is not ignored. This is the default value.

- `true`

  The value is ignored. Either the `--value` option or the `--ignoredescriptoritem` option must be set.

Do not use the `--ignoredescriptoritem` option to unset an environment entry that has previously been set by using the `set-web-env-entry` subcommand. Instead, use the `unset-web-env-entry` subcommand for this purpose.

Default value: `false`

`--description` *`description`*

Specifies an optional textual description of the environment entry that is being set.

Type: String

The following values can be specified:

- *Description of the parameter*

Default value: N/A

`application-name`

Specifies the name of the application. This name can be obtained by using the `list-applications` subcommand.

Type: String

The following values can be specified:

- *Name of the application*

Default value: N/A

`module`

Specifies the relative path to the module within the application's enterprise archive (EAR) file. The path to the module is specified in the `module` element of the application's `application.xml` file. `module` is required only if the environment entry applies to a Web module of a Java EE application. The module would be specified as the operand of this command as `myApp/myWebModule.war`.

The module must follow `application-name`, separated by a slash (`/`).

Type: String

The following values can be specified:

- *Name of the module in the application*

Default value: N/A

Examples

The following example sets the environment entry `Hello User` of the application `hello` to `techscribe`. The Java type of this entry is `java.lang.String`.

```
asadmin set-web-env-entry --name="Hello User"
--type=java.lang.String --value=techscribe
--description "User authentication for Hello application" hello
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.8.18 show-component-status

Displays the status of the deployed component.

Synopsis

```
asadmin [asadmin-options] show-component-status [--help]
    [--target target] component-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `show-component-status` subcommand of `asadmin` gets the status (either enabled or disabled) of the deployed component. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target where you show the component status.

Type: String

The following values can be specified:

- `server`
  Shows the component status for the default server instance `server` and is the default value.
- `domain`
  Shows the component status for the domain.
- *cluster_name*
  Shows the component status for the cluster.
- *instance_name*
  Shows the component status for a clustered or stand-alone server instance.

Default value: `server`

component-name

Specifies the name of the component whose status is to be listed.

The name can include an optional version identifier, which follows the name and is separated from the name by a colon (`:`). The version identifier must begin with a letter or number. It can contain alphanumeric characters plus underscore (_), dash (-), and period (`.`) characters. To list multiple versions, you can use an asterisk (`*`) as a wildcard character.

Type: String

The following values can be specified:

- *Name of the component*

Default value: N/A

Examples

The following example gets the status of the `MEjbApp` component on the server instance `instance1`.

```
asadmin show-component-status --target instance1 MEjbApp
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.8.19 undeploy

Removes a deployed component.

Synopsis

```
asadmin [asadmin-options] undeploy [--help]
    [--target target]
    [--cascade={false|true}] name
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `undeploy` subcommand of `asadmin` uninstalls a deployed application or module and removes it from the repository. This subcommand is supported in remote mode only.

If the removal of the EJB timer application failed because a timeout occurred while accessing the database, then you can no longer deploy the application. In this case, do the following to redeploy the EJB timer application:

1. Restart the server instance.
2. Undeploy the EJB timer application.
3. Deploy the EJB timer application.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--cascade={false|true}`

Specifies whether the related resources are deleted or not.

If set to `true`, deletes all the connection pools and connector resources associated with the resource adapter being undeployed.

If set to `false`, the undeploy fails if any pools and resources are still associated with the resource adapter. Then, either those pools and resources must be deleted explicitly, or the option must be set to `true`.

If the option is set to `false`, and if there are no pools and resources still associated with the resource adapter, the resource adapter is undeployed. This option is applicable to connectors (resource adapters) and applications.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--target` *`target`*

Specifies the target from where you are undeploying.

Type: String

The following values can be specified:

- `server`
  Undeploys the component from the default server instance `server` and is the default value.
- `domain`
  Undeploys the component from the domain.
- *cluster_name*
  Undeploys the component from every server instance in the cluster.
- *instance_name*
  Undeploys the component from a particular stand-alone server instance.

Default value: `server`

`name`

Specifies the name of the deployed component.

The name can include an optional version identifier, which follows the name and is separated from the name by a colon (`:`). The version identifier must begin with a letter or number. It can contain alphanumeric characters plus underscore (_), dash (`-`), and period (`.`) characters. To delete multiple versions, you can use an asterisk (`*`) as a wildcard character.

Type: String

The following values can be specified:

- *Name of the component*

Default value: N/A

Examples

The following example removes an enterprise application named `Cart.ear` on the server instance `instance1`.

```
asadmin undeploy --target instance1 Cart
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.8.20 unset-web-context-param

Unsets a servlet context-initialization parameter of a deployed Web application or module.

Synopsis

```
asadmin [asadmin-options] unset-web-context-param [--help]
    --name=context-param-name application-name[/module]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `unset-web-context-param` subcommand of asadmin unsets a servlet context-initialization parameter of one of the following items:

- A deployed Web application.
- A Web module in a deployed Java Platform, Enterprise Edition (Java EE) application.

When a parameter is unset, its value reverts to the value, if any, that is set in the application's DD. This subcommand enables you to change the configuration of a deployed application without the need to modify the application's DD and repackage and redeploy the application.

This subcommand is supported in remote mode only.

Precondition

- The environment entry must already be set.
- Domain Administration Server (DAS) is running.
- The application must already be deployed.

Files

The path to the module is specified in the module element of the application's `application.xml` file.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--name=`*`context-param-name`*

Specifies the name of the servlet context-initialization parameter that is to be unset.

Type: String

The following values can be specified:

- *Name of the variable*

Default value: N/A

application-name

Specifies the name of the application.

Type: String

The following values can be specified:

- *Name of the application*

Default value: N/A

module

Specifies the relative path to the module within the application's enterprise archive (EAR) file. The path to the module is specified in the `module` element of the application's `application.xml` file. `module` is required only if the servlet context-initialization parameter applies to a Web module of a Java EE application.

If specified, `module` must follow `application-name`, separated by a slash (`/`). For example, the `application.xml` file for the `myApp` application might specify the following Web module:

```
<module>
<web>
<web-uri>myWebModule.war</web-uri>
</web>
</module>
```

The module would be specified as the operand of this command as `myApp/myWebModule.war`.

Type: String

The following values can be specified:

- *Name of the application module*

Default value: N/A

Examples

The following example unsets the servlet context-initialization parameter `javax.faces.STATE_SAVING_METHOD` of the web application `basic-ezcomp`. The parameter reverts to the value, if any, that is defined in the application's DD.

```
asadmin unset-web-context-param --name=javax.faces.STATE_SAVING_METHOD
basic-ezcomp
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.8.21 unset-web-env-entry

Unsets an environment entry for a deployed Web application or module.

Synopsis

```
asadmin [asadmin-options] unset-web-env-entry [--help]
    --name=env-entry-name application-name[/module]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `unset-web-env-entry` subcommand of `asadmin` unsets an environment entry for one of the following items:

- A deployed Web application.
- A Web module in a deployed Java Platform, Enterprise Edition (Java EE) application.

When an entry is unset, its value reverts to the value, if any, that is set in the application's DD. This subcommand enables you to change the configuration of a deployed application without the need to modify the application's DD and repackage and redeploy the application.

This subcommand is supported in remote mode only.

Precondition

- The application must already be deployed.
- The entry must have previously been set.
- Domain Administration Server (DAS) is running.

Environment variable

The path to the module is specified in the `module` element of the application's `application.xml` file.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--name=`*`env-entry-name`*

Specifies the name of the environment entry that is to be unset. The name is a JNDI name relative to the `java:comp/env` context. The name must be unique within a deployment component.

Type: String

The following values can be specified:

- *Name of the environment entry*

Default value: N/A

`application-name`

Specifies the name of the application.

Type: String

The following values can be specified:

- *Name of the application*

Default value: N/A

module

Specifies the relative path to the module within the application's enterprise archive (EAR) file. The path to the module is specified in the `module` element of the application's `application.xml` file. `module` is required only if the environment entry applies to a Web module of a Java EE application.

If specified, *module* must follow `application-name`, separated by a slash (`/`). The *module* would be specified as the operand of this command as `myApp/myWebModule.war`.

Type: String

The following values can be specified:

- *Name of the application module*

Default value: N/A

Examples

The following example unsets the environment entry `Hello User` of the Web application `hello`. The entry reverts to the value, if any, that is defined in the application's DD.

```
asadmin unset-web-env-entry --name="Hello User" hello
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.9 Commands used for security administration

This section describes the syntax and functionality of the commands used for security administration.

2.9.1 change-master-password

Changes the master password.

Synopsis

```
asadmin [asadmin-options] change-master-password [--help]
    [--nodedir node-dir] [--domaindir domain-dir]
    [--savemasterpassword={false|true}] [domain-name|node-name]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `change-master-password` subcommand of asadmin is used to modify the master password. This command is interactive such that the user is prompted for both the old and new master passwords.

This command will not work unless the server is stopped. In a distributed environment, this command must be executed on each machine in the domain.

Precondition

The DAS or server instance must be stopped before executing the `change-master-password` subcommand.

Environment variable

AS_DEF_DOMAINS_PATH

Files

If the `--savemasterpassword` option of `change-master-password` subcommand is set to `true`, then the master password is changed and written to the `master-password` file in the `domains/domain-dir` directory).

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--nodedir` *node-dir*

Specifies the name of the directory containing the node instance, for which the password will be changed. If this option is excluded, the change is applied to the entire domain.

Type: String

The following values can be specified:

- *File path of a directory where the node resides*

Default value: N/A

`--domaindir` *domain-dir*

Specifies the name of the domain directory used for this operation. By default, the `--domaindir` option is `$AS_DEF_DOMAINS_PATH`, which is an environment variable defined in the `asenv.bat` or `asenv.conf` file.

Type: String

The following values can be specified:

- *Domain directory name*

Default value: `$AS_DEF_DOMAINS_PATH (which is an environment variable)`

`--savemasterpassword={false|true}`

Indicates whether the master password should be written to the file system.

This is necessary so that the `start-domain` subcommand can start the server without having to prompt the user.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

domain-name | node-name

Specifies the name of the domain or node for which the password will be changed. If there is a single domain, this is optional.

Type: String

The following values can be specified:

- *Name of a domain or a node*

Default value: N/A

Examples

The following example shows how to change the master password for the `domain44ps` domain.

```
asadmin change-master-password domain44ps
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.9.2 create-auth-realm

Adds the named authentication realm.

Synopsis

```
asadmin [asadmin-options] create-auth-realm [--help]
    --classname realm_class [--property name=value[:name=value]...]
    [--target target_name] auth_realm_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-auth-realm` subcommand adds the named authentication realm.

This command is supported in remote mode only.

Precondition

DAS has to be in a running state.

Files

You can specify the keyfile that stores user names, passwords, and group names for each realm.

The default file is `domain-dir/config/keyfile`.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target_name*

Specifies the target on which you are creating the realm.

Type: String

The following values can be specified:

- `server`
  Creates the realm on the default server instance. This is the default value.
- *configuration_name*
  Creates the realm in the specified configuration.
- *cluster_name*
  Creates the realm on all server instances in the specified cluster.
- *instance_name*
  Creates the realm on a specified server instance.

Default value: `server`

`--classname` *realm_class*

Specifies the Java class which implements this realm.

Type: String

The following values can be specified:

- `com.sun.enterprise.security.auth.realm.file.FileRealm`

- com.sun.enterprise.security.auth.realm.certificate.CertificateRealm
- com.sun.enterprise.security.auth.realm.jdbc.JDBCRealm
- com.sun.enterprise.security.auth.realm.ldap.LDAPRealm
- com.sun.enterprise.security.auth.realm.ldap.PamRealm
- A java class name that implements com.sun.appserv.security.AppservPasswordLoginModule class

Default value: N/A

--property *name*=*value*[:*name*=*value*]...

Specifies the optional attribute *name-value* pairs for configuring the authentication realm. The specification format is name=*value*. Use a colon (:) to separate property names. If you specify the same property name more than once, the property value specified last is valid.

The following properties are common to all of the supported realms including FileRealm, CertificateRealm, JDBCRealm, LDAPRealm, and PamRealm:

- jaas-context=*value*

  Specifies the Java Authentication and Authorization Service (JAAS) context.

  Type: String

  Default value: N/A

  Range Value: N/A

- assign-groups=*value*

  Indicates that its *value* is taken to be a comma-separated list of group names. All the clients who present valid certificates are assigned membership to these groups for the purposes of authorization decisions in the web and EJB containers.

  Type: String

  Default value: N/A

  Range Value: N/A

The following properties are realm-specific:

- FileRealm:
  file=*value*

  Specifies the file that stores user names, passwords, and group names. The default value is domain-dir/config/keyfile.

  Type: String

  Default value: domain-dir/config/keyfile

  Range Value: N/A

- CertificateRealm:
  LoginModule=*value*

  Specifies the name of a JAAS LoginModule to use for performing authentication. To use a JAAS LoginModule, you must first create an implementation of the javax.security.auth.spi.LoginModule interface, and then plug the module into a jaas-context.

  Type: String

  Default value: N/A

  Range Value: N/A

- JDBCRealm:
  datasource-jndi=*value*

Specifies the `jndi-name` of the `jdbc-resource` for the database.

Type: String

Default value: N/A

Range Value: N/A

`user-table=`*`value`*

Specifies the name of the user table in the database.

Type: String

Default value: N/A

Range Value: N/A

`user-name-column=`*`value`*

Specifies the name of the user name column in the database's user table.

Type: String

Default value: N/A

Range Value: N/A

`password-column=`*`value`*

Specifies the name of the password column in the database's user table.

Type: String

Default value: N/A

Range Value: N/A

`group-table=`*`value`*

Specifies the name of the group table in the database.

Specify the group table for an authentication realm of the JDBCRealm class.

Type: String

Default value: N/A

Range Value: N/A

`group-name-column=`*`value`*

Specifies the name of the group name column in the database's group table.

Type: String

Default value: N/A

Range Value: N/A

`db-user=`*`value`*

Allows you to specify the database user name in the realm instead of the `jdbc-connection-pool`. This prevents other applications from looking up the database, getting a connection, and browsing the user table. By default, the `jdbc-connection-pool` configuration is used.

Type: String

Default value: N/A

Range Value: N/A

`db-password=`*`value`*

Allows you to specify the database password in the realm instead of the `jdbc-connection-pool`. This prevents other applications from looking up the database, getting a connection, and browsing the user table. By default, the `jdbc-connection-pool` configuration is used.

Type: String

Default value: N/A

Range Value: N/A

digest-algorithm=*value*

Specifies the digest algorithm. The default algorithm is SHA-256. You can use any algorithm supported in the JDK, or none. If you have applications that depend on the MD5 algorithm, you can override the default SHA-25 algorithm by using the asadmin set subcommand:

```
asadmin set
server.security-service.property.default-digest-algorithm=MD5
```

You can use the asadmin get subcommand to determine what algorithm is currently being used:

```
asadmin get
server.security-service.property.default-digest-algorithm
```

Type: String

Default value: SHA-256

Range Value: N/A

digestrealm-password-enc-algorithm=*value*

Specifies the algorithm for encrypting passwords stored in the database. It is a security risk not to specify a password encryption algorithm.

Type: String

Default value: N/A

Range Value: N/A

encoding=*value*

Specifies the encoding. Allowed values are Hex and Base64. If digest-algorithm is specified, the default value is Hex. If digest-algorithm is not specified, by default no encoding is specified.

Type: String

Default value: If digest-algorithm is specified, the default is Hex, else by default no encoding is specified.

Range Value:

Hex

Base64

charset=*value*

Specifies the charset for the digest algorithm.

Type: String

Default value: N/A

Range Value: N/A

- LDAPRealm:

  directory=*value*

  Specifies the LDAP URL to your server.

  Type: String

  Default value: N/A

  Range Value: N/A

  base-dn=*value*

  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.

  Type: String

  Default value: N/A

  Range Value: N/A

  search-filter=*value*

Specifies the search filter to use to find the user. The default value is `uid=%s` (`%s` expands to the subject name).

Type: String

Default value: `uid=%s`

Range Value: N/A

`group-base-dn=`*`value`*

Specifies the base DN for the location of groups data. By default, it is same as the base-dn, but it can be tuned, if necessary.

Type: String

Default value: N/A

Range Value: N/A

`group-search-filter=`*`value`*

Specifies the search filter to find group memberships for the user. The default value is `uniquemember=%d` (`%d` expands to the user elementDN).

Type: String

Default value: `uniquemember=%d`

Range Value: N/A

`group-target=`*`value`*

Specifies the LDAP attribute name that contains group name entries. The default value is CN.

Type: String

Default value: CN

Range Value: N/A

`search-bind-dn=`*`value`*

Specifies an optional DN used to authenticate to the directory for performing the `search-filter` lookup. Only required for directories that do not allow anonymous search.

Type: String

Default value: N/A

Range Value: N/A

`search-bind-password=`*`value`*

Specifies the LDAP password for the DN given in `search-bind-dn`.

Type: String

Default value: N/A

Range Value: N/A

`auth_realm_name`

Specifies a short name for the realm. This name is used to refer to the realm.

Type: String

The following values can be specified:

- *Specify the realm name*

Default value: N/A

Examples

The following example creates a New Authentication Realm on the server instance `instance1`.

```
asadmin create-auth-realm --target instance1
--classname com.sun.enterprise.security.auth.realm.file.FileRealm
```

```
--property file=${com.sun.aas.instanceRoot}/config/
admin-keyfile:jaas-context=fileRealm file
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.9.3 create-file-user

Creates a new file user.

Synopsis

```
asadmin [asadmin-options] create-file-user [--help]
    [--authrealmname auth_realm_name] [--target target]
    [--groups user_groups[:user_groups]...] user_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-file-user` subcommand creates an entry in the keyfile with the following:

- Specified username
- Password
- Groups

You can create multiple groups can be created by separating them with a colon (`:`). If the `auth_realm_name` is not specified, an entry is created in the keyfile for the default realm. If the `auth_realm_name` is specified, an entry is created in the keyfile using the `auth_realm_name`.

The `--passwordfile` option of the `asadmin` command can be used to specify the password for the user. The password file entry must be of the form `AS_ADMIN_USERPASSWORD=user-password`. If a password is not provided, this command fails (only when secure administration is enabled and the user being created is an administrative user.)

This command is supported in remote mode only.

Precondition

The DAS should be in a running state.

Files

When the `create-file-user` subcommand is executed with `--passwordfile` option , then the password file entry must be of the form `AS_ADMIN_USERPASSWORD=user-password`.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the name of the target on which the command operates.

Type: String

The following values can be specified:

- `server`
- *configuration_name*
- *cluster_name*
- *instance_name*

Default value: `server`

`--groups` *user_groups*`[:`*user_groups*`]...`

Specifies the group associated with this file user.

Type: String

The following values can be specified:

- *Group name*

Default value: N/A

`--authrealmname` *auth_realm_name*

Specifies the name of the realm in which the new user is created. If this option is not specified, the user is created in the `file` realm.

Type: String

The following values can be specified:

- *Realm name*

Default value: `file`

user_name

Specifies the name of the file user to be created.

Type: String

The following values can be specified:

- *Name of user*

Default value: N/A

Examples

The following example creates a file user on the server instance `instance1`.

```
asadmin create-file-user
--target instance1
--groups staff:manager
--authrealmname auth-realm1 sample_user
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.9.4 create-jacc-provider

Enables administrators to create a JACC provider that can be used by third-party authorization modules for applications running in Java EE Server.

Synopsis

```
asadmin [asadmin-options] create-jacc-provider [--help]
    --policyproviderclass pol-provider-class
    --policyconfigfactoryclass pc-factory-class
    [--property name=value[:name=value]...]
    [--target target] jacc-provider-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-jacc-provider` subcommand of asadmin creates a JSR-115—compliant Java Authorization Contract for Containers (JACC) provider that can be used for authorization of applications running in Java EE Server. The default Application Server installation includes two JACC providers, named `default` and `simple`. The `create-jacc-provider` subcommand makes it possible to specify additional third-party JACC providers. This command is supported in remote mode only.

The JACC provider is created as a `jacc-provider` element within the `security-service` element in the domain's `domain.xml` file.

Precondition

DAS has to be in a running state.

Files

- The JACC provider is created as a `jacc-provider` element within the `security-service` element in the domain's `domain.xml` file.
- For Operand `jacc-provider-name`: The name of the provider used to reference the `jacc-provider` element in `domain.xml` file.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--policyproviderclass` *`pol-provider-class`*

Specifies the fully qualified class name for the `javax.security.jacc.policy.provider` that implements the `java.security.Policy`.

Type: String

The following values can be specified:

- A class name which implements the `java.security.Policy` interface.

Default value: N/A

`--policyconfigfactoryclass` *`pc-factory-class`*

Specifies the fully qualified class name for the `javax.security.jacc.PolicyConfigurationFactory.provider` that implements the `provider-specific javax.security.jacc.PolicyConfigurationFactory`.

Type: String

The following values can be specified:

- A class name which implements the `javax.security.jacc.PolicyConfigurationFactory` interface.

Default value: N/A

`--property` *`name`*`=`*`value`*`[:`*`name`*`=`*`value`*`]...`

Specify property name and property value as a pair for the options when configuring a JACC provider. The specification format is *name=value*. To specify multiple formats, separate formats with a colon (`:`). If the same property name is specified more than once, the value specified at the end takes effect.

name

Type: String

The following values can be specified:

- `repository`: Specifies the directory containing the JACC policy file. For the default Java EE Server JACC provider, the default directory is `${com.sun.aas.instanceRoot}/generated/policy`. This property is not defined by default for the simple Java EE Server JACC provider.

Default value: N/A

value

Type: String

The following values can be specified:

- *Path to the directory that stores the JACC policy file*

Default value: `${com.sun.aas.instanceRoot}/generated/policy` (The default directory of the `default` Java EE Server JACC provider)

`--target` *`target`*

Specifies the target for which you are creating the JACC provider.

Type: String

The following values can be specified:

- `server`

  Creates the JACC provider on the default server instance. This is the default value.
- *configuration_name*

  Creates the JACC provider in the specified configuration.
- *cluster_name*

Creates the JACC provider on all server instances in the specified cluster.

- *instance_name*
  Creates the JACC provider on a specified server instance.

Default value: `server`

jacc-provider-name

Specifies the name of the provider, used to reference the `jacc-provider` element in `domain.xml`.

Type: String

The following values can be specified:

- *Specify provider name*

Default value: N/A

Examples

The following example will create jacc provider on the server instance `instance1`:

```
asadmin create-jacc-provider --target instance1
--policyproviderclass com.sun.enterprise.security.provider.PolicyWrapper
--policyconfigfactoryclass
com.sun.enterprise.security.provider.PolicyConfigurationFactoryImpl
testJACC
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.9.5 create-password-alias

Creates a password alias.

Synopsis

```
asadmin [asadmin-options] create-password-alias [--help] aliasname
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-password-alias` subcommand creates an alias for a password. An alias is a token of the form `${ALIAS=aliasname}`. The password that corresponds to the alias name is stored in an encrypted form.

This command can be run interactively or non-interactively.

- When run interactively, the command prompts the user for the alias password and to confirm the alias password.

- When run non-interactively, the command reads the alias password from a file that is passed through the `--passwordfile` option of the `asadmin` utility. The file must contain an entry of the form `AS_ADMIN_ALIASPASSWORD=alias-password`, where `alias-password` is the alias password. The non-interactive form of this command is suitable for use in scripts. This command is supported in remote mode only.

Precondition

DAS has to be in a running state.

Files

The subcommand reads the alias password from a file that is passed through the `--passwordfile` option. The file must contain an entry of the form `AS_ADMIN_ALIASPASSWORD=alias-password`, where `alias-password` is the alias password.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`aliasname`

Indicates your choice of name for the password alias.

Type: String

The following values can be specified:

- *Specify the password*

Default value: N/A

Examples

The following example creates a password alias interactively.

```
asadmin create-password-alias jmspassword-alias
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

Notes

Restart the DAS after creating the alias for the changes to take effect.

2.9.6 delete-auth-realm

Removes the named authentication realm.

Synopsis

```
asadmin [asadmin-options] delete-auth-realm [--help] [--target target]
    auth_realm-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-auth-realm` subcommand removes the named authentication realm. This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target on which you are deleting the authentication realm.

Type: String

The following values can be specified:

- `server`
  Deletes the realm for the default server instance "`server`" and is the default value.
- *configuration_name*
  Deletes the realm for the named configuration.
- *cluster_name*
  Deletes the realm for every server instance in the cluster.
- *instance_name*
  Deletes the realm for a particular server instance.

Default value: `server`

auth_realm_name

Specifies the name of the realm.

Type: String

The following values can be specified:

- *Name of the realm*

Default value: N/A

Examples

The following example deletes the authentication realm `db` in the server instance `instance1`.

```
asadmin delete-auth-realm --target instance1 db
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.9.7 delete-file-user

Removes the named file user.

Synopsis

```
asadmin [asadmin-options] delete-file-user [--help]
    [--authrealmname auth_realm_name] [--target target] username
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-file-user` subcommand deletes the entry in the keyfile for the specified username.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--authrealmname` *auth_realm_name*

Specifies the name of the authentication realm with which the user was created.

Type: String

The following values can be specified:

- *Name of the authentication realm*

Default value: `file`

`--target` *target*

Specifies the name of the target on which the command operates.

Type: String

The following values can be specified:

- `server`
  Deletes the file user on the default server instance. This is the default value.
- *cluster_name*
  Deletes the file user from every server instance in the cluster.
- *instance_name*

Deletes the file user from a particular server instance.

Default value: `server`

username

Specifies the name of the "`file user`".

Type: String

The following values can be specified:

- *Name of the file user*

Default value: N/A

Examples

The following example shows how to delete user named `sample_user` from a `file` realm.

```
asadmin delete-file-user sample_user
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.9.8 delete-jacc-provider

Deletes JACC providers defined for a domain. This option is available for administrators only.

Synopsis

```
asadmin [asadmin-options] delete-jacc-provider [--help]
    [--target target] jacc-provider-name
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `delete-jacc-provider` subcommand enables administrators to delete JACC providers defined for a domain. JACC providers are defined as `jacc-provider` elements in the `security-service` element in the domain's `domain.xml` file. JACC providers can be created using the `create-jacc-provider` subcommand.

The default Java EE Server installation includes two JACC providers, named `default` and `simple`. These default providers should not be deleted.

The JACC provider used by Java EE Server for authorization is identified by the `jacc-provider` element of `security-service` in `domain.xml`. Therefore, if you delete the `jacc-provider` provider, make sure you change `jacc-provider` to the name of some other JACC provider that exists under `security-service`.

If you change the `jacc-provider` element to point to a different JACC provider, you must restart Java EE Server.

This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Files

JACC providers are defined as `jacc-provider` elements in the `security-service` element in the domain's `domain.xml` file.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target from which you are deleting the JACC provider.

Type: String

The following values can be specified:

- `server`
  Deletes the JACC provider on the default server instance. This is the default value.
- *configuration_name*
  Deletes the JACC provider in the specified configuration.
- *cluster_name*
  Deletes the JACC provider on all server instances in the specified cluster.
- *instance_name*
  Deletes the JACC provider on a specified server instance.

Default value: `server`

jacc-provider-name

Specifies the name of the JACC provider.

Type: String

The following values can be specified:

- *Name of the JACC provider*

Default value: N/A

Examples

The following example shows how to delete a JACC provider named `testJACC` from the default domain on the server instance `instance1`.

```
asadmin delete-jacc-provider --target instance1 testJACC
```

Exit Status

Exit Status	Explanation
0	command executed successfully.

Exit Status	Explanation
1	error in executing the command.

2.9.9 delete-password-alias

Deletes a password alias.

Synopsis

```
asadmin [asadmin-options] delete-password-alias [--help]
    aliasname
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-password-alias` subcommand deletes a password alias.

Precondition

Domain Administration Server (DAS) is running.

Files

For *aliasname* operand, this is the name of the substitute password as it appears in `domain.xml`.

Arguments

`--help | -?`

Displays the help text for the subcommand.

aliasname

Specifies the name of the substitute password as it appears in the `domain.xml` file.

Type: String

The following values can be specified:

- *Name of the substitute password*

Default value: N/A

Examples

```
asadmin delete-password-alias jdbcpassword-alias
```

Exit Status

Exit Status	Explanation
0	command executed successfully.

Exit Status	Explanation
1	error in executing the command.

2.9.10 disable-secure-admin

Disables secure admin if it is already enabled.

Synopsis

```
asadmin [asadmin-options] disable-secure-admin [--help]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `disable-secure-admin` subcommand disables secure admin if it is already enabled.

You must restart any running servers in the domain after you enable or disable secure admin. It is simpler to enable or disable secure admin with only the DAS running, then restart the DAS, and then start any other instances.

Precondition

The DAS must be running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

Examples

The following example shows how to disable secure admin for a domain.

```
asadmin disable-secure-admin
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.9.11 enable-secure-admin

Enables secure admin.

Synopsis

```
asadmin [asadmin-options] enable-secure-admin [--help]
    [--adminalias=alias]
    [--instancealias=alias]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `enable-secure-admin` subcommand enables secure admin (if it is not already enabled), optionally changing the alias used for DAS-to-instance admin messages or the alias used for instance-to-DAS admin messages. It causes the DAS and the instances in the domain to use SSL certificates for encrypting the messages they send to each other. This subcommand also allows the DAS to accept administration messages from remote admin clients such as the `asadmin` utility and IDEs. By default, when secure admin is enabled, the DAS and the instances use these SSL certificates to authenticate each other as security "principals" and to authorize admin access. The `--adminalias` value indicates to the DAS which SSL certificate it must use to identify itself to the instances. The `--instancealias` value determines for instances, which SSL certificate they must use to identify themselves to the DAS. The `enable-secure-admin` subcommand fails if any administrative user in the domain has a blank password.

The `enable-secure-admin` subcommand fails if any administrative user has a blank password.

Precondition

Domain Administration Server (DAS) is running, and not any instances.

HTTPS ports of DAS must be open.

Enable the required settings to open the HTTPS port of instance.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--adminalias=`*alias*

Specifies the alias that refers to the SSL/TLS certificate on the DAS. This alias is used by the DAS to identify itself to instances.

Type: String

The following values can be specified:

- *Alias name of the SSL/TLS certificate*

Default value: `s1as`

`--instancealias=`*alias*

Specifies the alias that refers to the SSL/TLS certificate on the instances. This alias is used by the instances to identify themselves to the DAS.

Type: String

The following values can be specified:

- *Alias name of the SSL/TLS certificate*

Default value: `glassfish-instances`

Examples

The following example informs how to enable secure admin for a domain using an `admin` alias `adtest` and an instance alias `intest`.

```
asadmin enable-secure-admin --adminalias adtest --instancealias intest
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.9.12 list-auth-realms

Lists the authentication realms.

Synopsis

```
asadmin [asadmin-options] list-auth-realms [--help]
    [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-auth-realms` subcommand lists the authentication realms. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

target

Specifies the name of the target for which you want to list the authentication realms.

Type: String

The following values can be specified:

- `server`
  Lists the realms for the default server instance `server` and is the default value.
- *configuration_name*
  Lists the realms for the named configuration.

- *cluster_name*
  Lists the realms for every server instance in the cluster.
- *instance_name*
  Lists the realms for a particular server instance.

Default value: `server`

Examples

The following example lists the authentication realms.

```
asadmin list-auth-realms
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.9.13 list-file-groups

Lists the file groups.

Synopsis

```
asadmin [asadmin-options] list-file-groups [--help]
    [--name username] [--authrealmname auth_realm_name]
    target
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-file-groups` subcommand lists the file users and groups supported by the file realm authentication. This subcommand lists available groups in the file user. This subcommand is supported in remote mode only.

If the `--name` option is not specified, all groups are listed.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`
Displays the help text for the subcommand.

`--name` *username*

Identifies the name of the file user for whom the groups will be listed.

Type: String

The following values can be specified:

- *Name of the file user*

Default value: N/A

`--authrealmname` *auth_realm_name*

Specifies the name of the authentication realm for which to list available groups.

Type: String

The following values can be specified:

- *Name of the auth realm*

Default value: N/A

target

Specifies which configurations you can list.

Type: String

The following values can be specified:

- `server`
  Lists the file groups in the current server. This is the default value.
- *cluster_name*
  Lists the file groups in a cluster.
- *instance_name*
  Lists the file groups for a particular instance.

Default value: `server`

Examples

This example list all file realm groups defined for the `server`.

```
asadmin list-file-groups
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.9.14 list-file-users

Lists the file users.

Synopsis

```
asadmin [asadmin-options] list-file-users [--help]
    [--authrealmname auth_realm_name] [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-file-users` subcommand displays a list of file users supported by file realm authentication.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--authrealmname` *auth_realm_name*

Lists only the users in the specified authentication realm.

Type: String

The following values can be specified:

- *Name of the auth realm*

Default value: N/A

target

Specifies the target for which you want to list file users.

Type: String

The following values can be specified:

- `server`
  Lists the file users on the default server instance. This is the default value.
- *configuration_name*
  Lists the file users in the specified configuration.
- *cluster_name*
  Lists the file users on all server instances in the specified cluster.
- *instance_name*
  Lists the file users on a specified server instance.

Default value: `server`

Examples

This example lists file users on the default file realm file.

```
asadmin list-file-users
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.9.15 list-jacc-providers

Lists JACC providers defined for a domain. This option is applicable only for administrators.

Synopsis

```
asadmin [asadmin-options] list-jacc-providers [--help]
    [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-jacc-providers` subcommand enables administrators to list the JACC providers defined for a domain. JACC providers are defined as `jacc-provider` elements in the `security-service` element in the `domain.xml` file of the corresponding domain. JACC providers can be created using the `create-jacc-provider` subcommand. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Files

JACC providers are defined as `jacc-provider` elements in the `security-service` element in the domain's `domain.xml` file.

Arguments

`--help | -?`

Displays the help text for the subcommand.

target

Specifies the target for which you want to list JACC providers.

Type: String

The following values can be specified:

- `server`
  Lists the JACC providers on the default server instance. This is the default value.
- *configuration_name*
  Lists the JACC providers in the specified configuration.

- *cluster_name*
  Lists the JACC providers on all server instances in the specified cluster.
- *instance_name*
  Lists the JACC providers on a specified server instance.

Default value: `server`

Examples

The following example informs how to list JACC providers for the default domain.

```
asadmin list-jacc-providers
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.9.16 list-password-aliases

Lists all the password aliases.

Synopsis

```
asadmin [asadmin-options] list-password-aliases [--help]
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `list-password-aliases` subcommand of asadmin lists all the existing password aliases.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`
Displays the help text for the subcommand.

Examples

The following example lists all the password aliases.

```
asadmin list-password-aliases
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.9.17 list-supported-cipher-suites

Lists the cipher suites that are supported and made available to a specific Java EE Server target. This option is used only by administrators.

Synopsis

```
asadmin [asadmin-options] list-supported-cipher-suites [--help]
    [--target target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-supported-cipher-suites` subcommand of asadmin enables administrators to list the cipher suites that are supported and made available to a specified Java EE Server target. This subcommand is supported in remote mode only.

The cipher suites that may be available in addition to the default SSL/TLS providers that are bundled with Java EE Server packages will vary depending on the third-party provider.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target where you want to list the cipher suites.

Type: String

The following values can be specified:

- `server`
  Lists the cipher suites for the default `server` instance. This is the default value.
- *configuration_name*
  Lists the cipher suites for the specified configuration.
- *cluster_name*
  Lists the cipher suites for all the server instances in the specified cluster.

- *instance_name*
  Lists the cipher suites for a specified server instance.

Default value: `server`

Examples

The following example shows how to list cipher suites for the default domain.

```
asadmin list-supported-cipher-suites
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.9.18 update-file-user

Updates a current `file user` as specified.

Synopsis

```
asadmin [asadmin-options] update-file-user [--help]
    [--groups user_groups[:user_groups]...] [--target target]
    [--authrealmname authrealm_name] username
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `update-file-user` subcommand of `asadmin` updates an existing entry in the keyfile using the specified user name, password and groups. Multiple groups can be entered by separating them, with a colon (`:`).

If a new password is not provided, this subcommand fails if secure administration is enabled and the user being updated is an administrative user.

Precondition

Domain Administration Server (DAS) is running.

Files

`keyfile` file at *domain-dir*/*domain_name*/`config` directory is updated.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--groups` *`user_groups`*`[:`*`user_groups`*`]...`

Specifies the name of the group to which the `file user` belongs.

Type: String

The following values can be specified:

- *Group names*

Default value: N/A

`--target` *`target`*

Specifies the target on which you are updating a `file user`.

Type: String

The following values can be specified:

- `server`
  Updates the `file user` in the default server instance. This is the default value.
- *cluster_name*
  Updates the `file user` on every server instance in the cluster.
- *instance_name*
  Updates the `file user` on a specified sever instance.

Default value: `server`

`--authrealmname` *`authrealm_name`*

Specifies the name of the authentication realm where the user to be updated can be found.

Type: String

The following values can be specified:

- *Name of the authentication realm*

Default value: `file`

`username`

Specifies the name of the `file user` to be updated.

Type: String

The following values can be specified:

- *File user name*

Default value: N/A

Examples

The following example updates information for a file realm user named `sample_user` on the server instance `instance1`.

```
asadmin update-file-user --target instance1
--groups staff:manager:engineer sample_user
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.9.19 update-password-alias

Updates a password alias.

Synopsis

```
asadmin [asadmin-options] update-password-alias [--help]
    aliasname
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The update-password-alias subcommand of asadmin updates the password alias IDs in the named target. An alias is a token of the form ${ALIAS=password-alias-password}. The password corresponding to the alias name is stored in an encrypted form. The update-password-alias subcommand takes both a secure interactive form (in which the user is prompted for all information) and a more script-friendly form, in which the password is propagated on the command line. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Files

This is the name of the password as it appears in domain.xml

Arguments

--help | -?

Displays the help text for the subcommand.

aliasname

Specifies the name of the password as it appears in the domain.xml file.

Type: String

The following values can be specified:

- *Name of the password alias*

Default value: N/A

Examples

The following example updates the password for the `jmspassword-alias` alias.

```
asadmin update-password-alias jmspassword-alias
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

Notes

Restart the DAS after updating the alias for the changes to take effect.

2.10 Commands used for log administration

This section describes the syntax and functionality of the commands used for log administration.

2.10.1 collect-local-snapshot

The `collect-local-snapshot` subcommand collects system information when the domain administration server stops.

Synopsis

```
asadmin [asadmin_options] collect-local-snapshot
    [--infotype info_type1[,info_type2]...]
    [--domaindir domain_root_dir]
    [--domain domain_name] [--nodedir node_root_dir]
    --targettype={node|server}
    [--targetname target_name1[,target_name2]...]
    [--archivefile archive_file] [--timeout timeout]
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `collect-local-snapshot` subcommand collects the system information of the products that configure Application Server. The collected information is output as a system information archive file.

If there is a forced termination during the execution of the subcommand, the following files and directories may remain in the output destination directory of the system information archive files.

- Files generated temporarily by the subcommand with a `.zip` extension.
- Files generated temporarily by the subcommand with a `.tmp` extension.
- Directories generated temporarily by the subcommand (the directory name differs for each execution).

As you can set the maximum value for the number of system information archive files to be generated in the output destination directory, the ".zip" file will not remain. Delete the other files and directories as needed. Before deleting, check the message log to confirm that the `collect-snapshot` and `collect-local-snapshot` subcommands are not running.

Important note

- The thread dump file is generated each time the `collect-local-snapshot` subcommand is run. If the thread dump file is not required, delete the file after running the subcommand. By default, the thread dump file is saved in the current directory. The current directory of the domain administration server is *Application Server installation directory*`/javaee/glassfish/domains/`*domain_name*`/config`, and the current directory of the Java EE server is *Application Server installation directory*`/javaee/glassfish/nodes/`*node_name*`/`*server_instance_name*`/config`.

- If the output destination of the thread dump file is changed from the current directory, the thread dump file is not included in the system information archive file.
- If the system information collection process is interrupted by a timeout, the execution of the `collect-local-snapshot` subcommand, which is internally called, is also interrupted. However, if the internal process (`collect-local-snapshot` subcommand) has called another process, the `collect-local-snapshot` subcommand process might not be interruptible.

Execution permission

General user

Precondition

- Do not run this subcommand multiple times on the same server simultaneously.

Arguments

`--infotype` *info_type1*`[,`*info_type2*`]...`

Specifies the type of system information to be collected. You can specify multiple types separated by a comma.

Type: String

You can specify the following values:

- `initinfo`

  Collects the system information for initial verification, which is the minimum requirement to eliminate a problem.

- `sysinfo`

  Collects all the system information required for verifying the cause. It also includes the system information required for the initial verification.

Default value: `initinfo,sysinfo`

`--domaindir` *domain_root_dir*

Specifies the absolute path of the domain root directory. If the domain root directory was not changed from the default value of Application Server or if the subcommand is run on a host in which the domain administration server does not exist, you can ignore this option.

Type: String

You can specify the following values:

- Absolute path of the domain root directory

Default value: *Application Server installation directory*`/javaee/glassfish/domains`

`--domain` *domain_name*

If a node or server of the same name exists in multiple servers, specify the name of the domain that includes the node or server from which system information should be collected. If not, you can omit this.

If multiple domains have the name specified in this option, and if all the domains have a server of the same name specified in the `--targetname` option, system information of all the servers whose domain names and server names are identical is collected.

Type: String

You can specify the following values:

- Domain name

Default value: (None)

`--nodedir` *node_root_dir*

Specify the absolute path of the node root directory on which a server (the Java EE server, the web server, or PRF) is created. If the node root directory in which the server is created was not changed from the default value of Application Server or if the subcommand is run on a host in which the domain administration server does not exist, you can ignore this option.

This option can be used to specify only one node root directory. To retrieve system information of nodes or servers under different node root directories, run the `collect-local-snapshot` subcommand in each of the node root directories.

Type: String

You can specify the following values:

- Absolute path of the node root directory

Default value: *Application Server installation directory*`/javaee/glassfish/nodes`

`--targettype={node|server}`

Specifies the target for collecting the system information.

Type: String

You can specify the following values:

- `node`

  Collects the system information of all the servers configured in the node specified in the `--targetname` option.

- `server`

  Collects the system information of the servers (Java EE, Web, or PRF servers) specified in the `--targetname` option.

Default value: (None)

`--targetname` *target_name1*`[,`*target_name2*`]...`

Specifies the name of the server or node that collects the system information. You can specify multiple names separated by a comma.

Type: String

You can specify the following values:

- Node name or server name

Default value: `server`

`--archivefile` *archive_file*

Specifies the character string at the end of the system information archive file name. The format of the file name is as follows:

`snapshot-`*info_type*`-`*archive_file*

info_type: The value specified in the `--infotype` option.

archive_file: The value specified in the `--archivefile` option.

When multiple values are specified in the `--infotype` option, the subcommand creates the system information archive file corresponding to each specified value.

If a file with the same name exists in the destination directory of system information archive file, then the existing file is overwritten.

Type: String

You can specify the following values:

- String attached at the end of the system information archive file
  You can use the following single byte characters.
  Lower-case letters
  Upper-case letters
  Numbers
  Hyphen
  Period
  Underscore

Default value: *domain-name-date-process-id-thread-id*.zip

date is in the yyyyMMddHHmmss format.

process-id is process id (Hexadecimal) of collect-local-snapshot subcommand.

thread-id is thread id (Hexadecimal) of collect-local-snapshot subcommand.

--timeout *timeout*

Specifies the time-out period that the collect-local-snapshot subcommand suspends the process of collecting the system information and terminates the command execution.

If the collection of the system information stops, an incomplete system information archive file might remain in the output destination directory. In this case, delete the file as needed.

Type: Integer

You can specify the following values:

- 1 to 2147483647

Default value: (None. The collect-local-snapshot subcommand does not time out.)

Examples

The following example collects the system information of the server with the name server:

```
asadmin collect-local-snapshot --targettype=server
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

Output Format

This subcommand collects the system information specified in the --infotype option and creates the system information archive file corresponding to each target.

For details on the name of the system information archive file, refer to the explanation about the --archivefile option.

The system information archive file is output to the following directory in the host where the subcommand is executed:

Application Server installation directory/javaee/snapshot

The following properties of the `set` and `get` subcommands of the `asadmin` utility can be used to set and verify the location of the output destination directory and the number of system information archive files to be generated in the output destination directory.

- `hitachi-domain.hitachi-snapshot.output-dir`#
- `hitachi-domain.hitachi-snapshot.max-num`
- `nodes.node.`*`node_name`*`.hitachi-node.hitachi-node-snapshot.output-dir`#
- `nodes.node.`*`node_name`*`.hitachi-node.hitachi-node-snapshot.max-num`

#

If you do not have access to the specified output destination directory, an error message (KDKD10211-E) is output and the collection of system information stops.

For details on properties, see List of parameters used with the set and get subcommands.

2.10.2 collect-snapshot

The `collect-snapshot` subcommand collects system information when the domain administration server starts.

Synopsis

```
asadmin [asadmin_options] collect-snapshot
    [--infotype info_type1[,info_type2]...]
    --targettype={domain|node|server}
    [--targetname target_name1[,target_name2]...] [--timeout timeout]
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `collect-snapshot` subcommand collects the system information of the products that configure Application Server. The collection is performed on multiple hosts where the nodes and the servers that are used to configure the domain exist. The collected information is output to the hosts as system information archive files.

If there is a forced termination during the execution of the subcommand, the following files and directories may remain in the output destination directory of the system information archive files.

- Files generated temporarily by the subcommand with a `.zip` extension.
- Files generated temporarily by the subcommand with a `.tmp` extension.
- Directories generated temporarily by the subcommand (directory name changes for each execution).

As you can set the maximum value for the number of system information archive files to be generated in the output destination directory, the ".zip" file will not remain. Delete the other files and directories as needed. Before deleting, check the message log to confirm that the `collect-snapshot` and `collect-local-snapshot` subcommands are not running.

Important note

- The thread dump file is generated each time the `collect-snapshot` subcommand is run. If the thread dump file is not required, delete the file after running the subcommand. By default, the thread dump file is saved in the current directory. The current directory of the domain administration server is *Application Server installation directory*`/javaee/glassfish/domains/`*domain_name*`/config`, and the current directory of the Java EE server is *Application Server installation directory*`/javaee/glassfish/nodes/`*node_name*`/`*server_instance_name*`/config`.
- If the output destination of the thread dump file is changed from the current directory, the thread dump file is not included in the system information archive file.
- If the system information collection process is interrupted by a timeout, the execution of the `collect-snapshot` subcommand, which is internally called, is also interrupted. However, if the internal process (`collect-snapshot` subcommand) has called another process, the `collect-snapshot` subcommand process might not be interruptible.
- If the host where the domain administration server exists is unable to communicate with a host, make sure that the node and the server that exist in the host are excluded from the collection targets, and then run the `collect-snapshot` subcommand.

Execution permission

General user

Precondition

- The domain administration server is running.
- Communication must be possible with the host that collects the system information archive file.
- Do not run this subcommand multiple times on the same server simultaneously.

Arguments

`--infotype` *info_type1*`[,`*info_type2*`]...`

Specifies the type of system information to be collected. You can specify multiple types separated by a comma.

Type: String

You can specify the following values:

- `initinfo`

  Collects the system information for initial verification, which is the minimum requirement to eliminate a problem.

- `sysinfo`

  Collects all the system information required for verifying the cause. It also includes the system information required for the initial verification.

Default value: `initinfo,sysinfo`

`--targettype={domain|node|server}`

Specifies the target for collecting the system information.

Type: String

You can specify the following values:

- `domain`

Collects the system information of all the servers that configure the domain. The value specified in the `--targetname` option is not used.

- `node`

  Collects the system information of all the servers that configure the node specified in the `--targetname` option.

- `server`

  Collects the system information of the servers (Java EE, Web, or PRF servers) specified in the `--targetname` option.

Default value: (None)

`--targetname` *`target_name1`*`[,`*`target_name2`*`]...`

Specifies the name of the server or node that collects the system information. You can specify multiple names separated by a comma.

If `domain` is specified in `--targettype` option, the specified values for this option are not used.

Type: String

You can specify the following values:

- Node name or server name.

Default value: `server`

`--timeout` *`timeout`*

Specifies the time period required for the `collect-local-snapshot` subcommand called by this subcommand to stop the collection of system information and finish the command execution.

If the collection of the system information stops, an incomplete system information archive file might remain in the output destination directory. In this case, delete the file as needed.

Type: Integer

You can specify the following values:

- `1` to `2147483647`

Default value: (None. The `collect-local-snapshot` subcommand does not time out.)

Examples

The following example collects the system information of all the servers configured in the domain:

```
asadmin collect-snapshot --targettype=domain
```

Exit Status

Exit Status	Explanation
`0`	Normal termination.
`1`	Abnormal termination.

Output Format

This subcommand collects the system information specified in the `--infotype` option and creates the system information archive file corresponding to each target. The naming convention of the system information archive file is as follows:

```
snapshot-info_type#1-collection_target_name#2-date_and_time#3-process-id#4-thread-
id#5.zip
```

#1

The value specified in the `--infotype` option. If multiple values are specified, multiple system information archive files that correspond to the specified values are generated.

#2

This is *domain_name* if the file is output to a host where the domain administration server exists. This is *domain_name_host_name* if the file is output to a node where the domain administration server does not exist.

#3

The format is *yyyyMMddHHmmss*. If a node or server that exists on multiple hosts is specified in the `--targetname` option, the date and time of the file name to be saved in each host changes to the date and time when the `collect-snapshot` subcommand was executed.

#4

process-id is process id (Hexadecimal) of the domain administration server.

#5

thread-id is thread id (Hexadecimal) of `collect-snapshot` subcommand.

The system information archive file is saved in the following directory in the host where the server specified by the `--targetname` option exists:

Application Server installation directory`/javaee/snapshot`

The following properties of the `set` and `get` subcommands of the `asadmin` utility can be used to set and verify the location of the output destination directory and the number of system information archive files to be generated in the output destination directory.

- `hitachi-domain.hitachi-snapshot.output-dir`#
- `hitachi-domain.hitachi-snapshot.max-num`
- `nodes.node.`*node_name*`.hitachi-node.hitachi-node-snapshot.output-dir`#
- `nodes.node.`*node_name*`.hitachi-node.hitachi-node-snapshot.max-num`

#:

If you do not have access to the specified output destination directory, an error message (KDKD10211-E) is output and the collection of system information stops.

For details on properties, see List of parameters used with the set and get subcommands.

2.10.3 list-log-attributes

Lists all the logging attributes defined for a specific target in a domain.

Synopsis

```
asadmin [asadmin-options] list-log-attributes [--help]
    [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The list-log-attributes subcommand of asadmin lists all the logging attributes currently defined for the specified Java EE Server domain or a target within a domain. The values listed correspond to the values in the logging.properties file for the domain. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Files

The logging attributes are currently defined for a specified Java EE Server domain or target within a domain. These attributes are listed against the corresponding values in the logging.properties file for the domain.

Arguments

--help | -?

Displays the help text for the subcommand.

target

Type: String

The following values can be specified:

- server
  The default server instance. This is the default value.
- *configuration_name*
  The name of a specific configuration.
- *cluster_name*
  The name of a target cluster.
- *instance_name*
  The name of a target server instance.

Default value: server

Examples

The following example lists all the loggers attributes for the default domain.

```
asadmin list-log-attributes
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.10.4 set-log-attributes

Sets the logging attributes for one or more loggers.

Synopsis

```
asadmin [asadmin-options] set-log-attributes [--help]
    [--target=target]
    attribute-name=attribute-value[:attribute-name=attribute-value]...
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `set-log-attributes` subcommand of `asadmin` sets the logging attributes for one or more loggers. The set attributes correspond to the attributes that are available in the `logging.properties` file for the domain. Depending on the attributes set, a server restart may be necessary.

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Files

The attributes you can set correspond to the attributes that are available in the `logging.properties` file for the domain.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target=`*target*

Specifies the server domain, instance, or cluster for which the logger attributes will be set. If this option is omitted, attributes are set for the default server.

Type: String

The following values can be specified:

- `server`

  Default target is `server`. If no target is specified then log attributes are set for the `server`.

- *cluster_name*

  The name of a target cluster.

- *instance_name*

  The name of a target server instance.

Default value: `server`

attribute-name

Specifies the fully scoped name of the logging attribute. For details about the specifiable attribute names, refer to List of parameters used with the set and get subcommands.

Type: String

Default value: N/A

attribute-value

Specifies the value to be applied to the specified attribute. For details about the specifiable attribute values, refer to List of parameters used with the set and get subcommands.

Type: String

Examples

In the following example, the switch of the message log of the server instance `InstanceA` size is set to `16777216`.

```
asadmin set-log-attributes --target instanceA \
ServerInstance.message_log.rotation-size=16777216
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.10.5 List of parameters used for the set-log-attributes command

This section describes parameters that can be specified for *attribute-name* of the `set-log-attributes` subcommand.

Parameters used for the set-log-attributes command

No.	Parameter name	Description	Range Value[#1]	Default Value	Value specified to the --target option	Example
1	`ServerInstance.message_log.enabled`	Enables or disables message log output of the server instance.	`true`|`false`	`true`	*server-instance-name* or *cluster-name*	`asadmin set-log-attributes --target=instanceA ServerInstance.message_log.enabled=true`
2	`ServerInstance.message_log.rotation-size`	Specifies the size at which the message log of the server instance is to be rotated.	`4096` to `2147483647`	`16777216`	*server-instance-name* or *cluster-name*	`asadmin set-log-attributes --target=instanceA ServerInstance.message_log.rotation-size=4096`

No.	Parameter name	Description	Range Value[#1]	Default Value	Value specified to the --target option	Example
3	ServerInstance.message_log.rotation-time	Specifies the time when the message log of the server instance is to be rotated.	Specify the lap time in the HHMMSS format.	000000	*server-instance-name* or *cluster-name*	asadmin set-log-attributes --target=instanceA ServerInstance.message_log.rotation-time=000000
4	ServerInstance.message_log.file-number	Specifies the number of message log files to be kept for the server instance.	1 to 16	8	*server-instance-name* or *cluster-name*	asadmin set-log-attributes --target=instanceA ServerInstance.message_log.file-number=8
5	ServerInstance.message_log.log-level	Specifies the log level for the message log of the server instance.[#2]	1 to 3	2	*server-instance-name* or *cluster-name*	asadmin set-log-attributes --target=instanceA ServerInstance.message_log.log-level=2
6	ServerInstance.stacktrace_log.enabled	Enables or disables stack trace log output of the server instance.	true|false	true	*server-instance-name* or *cluster-name*	asadmin set-log-attributes --target=instanceA ServerInstance.stacktrace_log.enabled=true
7	ServerInstance.stacktrace_log.rotation-size	Specifies the size at which the stack trace log of the server instance is to be rotated.	4096 to 2147483647	16777216	*server-instance-name* or *cluster-name*	asadmin set-log-attributes --target=instanceA ServerInstance.stacktrace_log.rotation-size=4096
8	ServerInstance.stacktrace_log.rotation-time	Specifies the time when the stack trace log of the server instance is to be rotated.	Specify the lap time in the HHMMSS format.	000000	*server-instance-name* or *cluster-name*	asadmin set-log-attributes --target=instanceA ServerInstance.stacktrace_log.rotation-time=000000
9	ServerInstance.stacktrace_log.file-number	Specifies the number of stack trace log files to be kept for the server instance.	1 to 16	8	*server-instance-name* or	asadmin set-log-attributes --

No.	Parameter name	Description	Range Value[#1]	Default Value	Value specified to the --target option	Example
					cluster-name	target=instanceA ServerInstance.stacktrace_log.file-number=8
10	ServerInstance.maintenance_log.enabled	Enables or disables maintenance log output of the server instance.	true|false	true	*server-instance-name* or *cluster-name*	asadmin set-log-attributes --target=instanceA ServerInstance.maintenance_log.enabled=true
11	ServerInstance.maintenance_log.rotation-size	Specifies the size at which the maintenance log of the server instance is to be rotated.	4096 to 2147483647	16777216	*server-instance-name* or *cluster-name*	asadmin set-log-attributes --target=instanceA ServerInstance.maintenance_log.rotation-size=4096
12	ServerInstance.maintenance_log.rotation-time	Specifies the time when the maintenance log of the server instance is to be rotated.	Specify the lap time in the HHMMSS format.	000000	*server-instance-name* or *cluster-name*	asadmin set-log-attributes --target=instanceA ServerInstance.maintenance_log.rotation-time=000000
13	ServerInstance.maintenance_log.file-number	Specifies the number of maintenance log files to be kept for the server instance.	1 to 16	8	*server-instance-name* or *cluster-name*	asadmin set-log-attributes --target=instanceA ServerInstance.maintenance_log.file-number=8
14	ServerInstance.maintenance_log.maintenance-log-level	Specifies the log level for the maintenance log of the server instance.	1 to 2	1	*server-instance-name* or *cluster-name*	asadmin set-log-attributes --target=instanceA ServerInstance.maintenance_log.maintenance-log-level=1
15	ServerInstance.glassfish_debug_log.enabled	Enables or disables the debugging log output of the server instance.	true|false	false	*server-instance-name* or *cluster-name*	asadmin set-log-attributes --target=instanceA ServerInstance.

No.	Parameter name	Description	Range Value[#1]	Default Value	Value specified to the --target option	Example
						glassfish_debug_log.enabled=false
16	ServerInstance.glassfish_debug_log.rotation-size	Specifies the size at which the debugging log of the server instance is to be rotated.	4096 to 2147483647	2147483647	*server-instance-name* or *cluster-name*	asadmin set-log-attributes --target=instanceA ServerInstance.glassfish_debug_log.rotation-size=4096
17	ServerInstance.glassfish_debug_log.rotation-time	Specifies the time when the debugging log of the server instance is to be rotated.	Specify the lap time in the HHMMSS format.	000000	*server-instance-name* or *cluster-name*	asadmin set-log-attributes --target=instanceA ServerInstance.glassfish_debug_log.rotation-time=000000
18	ServerInstance.glassfish_debug_log.file-number	Specifies the number of debugging log files to be kept for the server instance.	1 to 16	8	*server-instance-name* or *cluster-name*	asadmin set-log-attributes --target=instanceA ServerInstance.glassfish_debug_log.file-number=8
19	DAS.message_log.enabled	Enables or disables message log output of the domain administration server.	true|false	true	server	asadmin set-log-attributes --target=server DAS.message_log.enabled=true
20	DAS.message_log.rotation-size	Specifies the size at which the message log of the domain administration server is to be rotated.	4096 to 2147483647	16777216	server	asadmin set-log-attributes --target=server DAS.message_log.rotation-size=4096
21	DAS.message_log.rotation-time	Specifies the time when the message log of the server administration server is to be rotated.	Specify the lap time in the HHMMSS format.	000000	server	asadmin set-log-attributes --target=server DAS.message_log.rotation-time=000000
22	DAS.message_log.file-number	Specifies the number of message log files to be kept for the domain administration server.	1 to 16	8	server	asadmin set-log-attributes --target=server DAS.message_log.file-number=8

No.	Parameter name	Description	Range Value[#1]	Default Value	Value specified to the --target option	Example
23	DAS.message_log.log-level	Specifies the log level for the message log of the domain administration server.[#2]	1 to 3	2	server	asadmin set-log-attributes --target=server DAS.message_log.log-level=2
24	DAS.stacktrace_log.enabled	Enables or disables stack trace log output of the domain administration server.	true|false	true	server	asadmin set-log-attributes --target=server DAS.stacktrace_log.enabled=true
25	DAS.stacktrace_log.rotation-size	Specifies the size at which the stack trace log of the server administration server is to be rotated.	4096 to 2147483647	16777216	server	asadmin set-log-attributes --target=server DAS.stacktrace_log.rotation-size=4096
26	DAS.stacktrace_log.rotation-time	Specifies the time when the stack trace log of the server administration server is to be rotated.	Specify the lap time in the HHMMSS format.	000000	server	asadmin set-log-attributes --target=server DAS.stacktrace_log.rotation-time=000000
27	DAS.stacktrace_log.file-number	Specifies the number of stack trace log files to be kept for the domain administration server.	1 to 16	8	server	asadmin set-log-attributes --target=server DAS.stacktrace_log.file-number=8
28	DAS.maintenance_log.enabled	Enables or disables maintenance log output of the domain administration server.	true|false	true	server	asadmin set-log-attributes --target=server DAS.maintenance_log.enabled=true
29	DAS.maintenance_log.rotation-size	Specifies the size at which the maintenance log of the server administration server is to be rotated.	4096 to 2147483647	16777216	server	asadmin set-log-attributes --target=server DAS.maintenance_log.rotation-size=4096
30	DAS.maintenance_log.rotation-time	Specifies the time when the maintenance log of the server administration server is to be rotated.	Specify the lap time in the HHMMSS format.	000000	server	asadmin set-log-attributes --target=server DAS.maintenance_log.rotation-time=000000
31	DAS.maintenance_log.file-number	Specifies the number of maintenance log files to be kept for the domain administration server.	1 to 16	8	server	asadmin set-log-attributes --target=server DAS.maintenance

No.	Parameter name	Description	Range Value[#1]	Default Value	Value specified to the --target option	Example
						_log.file-number=8
32	DAS.maintenance_log.maintenance-log-level	Specifies the log level for the maintenance log of the domain administration server.	1 to 2	1	server	asadmin set-log-attributes --target=server DAS.maintenance_log.maintenance-log-level=1
33	DAS.glassfish_debug_log.enabled	Enables or disables debugging log output of the domain administration server.	true|false	false	server	asadmin set-log-attributes --target=server DAS.glassfish_debug_log.enabled=false
34	DAS.glassfish_debug_log.rotation-size	Specifies the size at which the debugging log of the server administration server is to be rotated.	4096 to 2147483647	2147483647	server	asadmin set-log-attributes --target=server DAS.glassfish_debug_log.rotation-size=4096
35	DAS.glassfish_debug_log.rotation-time	Specifies the time when the debugging log of the server administration server is to be rotated.	Specify the lap time in the HHMMSS format.	000000	server	asadmin set-log-attributes --target=server DAS.glassfish_debug_log.rotation-time=000000
36	DAS.glassfish_debug_log.file-number	Specifies the number of debugging log files to be kept for the domain administration server.	1 to 16	8	server	asadmin set-log-attributes --target=server DAS.glassfish_debug_log.file-number=8

#1:

If a value outside of the range is specified, the command execution fails.

#2:

To output messages that are output as the standard output to the message log, set the log level to 3. The default setting (2) outputs the messages that are output as the standard error output to the message log.

2.11 Commands used for setting parameters

This section describes the syntax and functionality of the commands used for setting the various parameters.

2.11.1 create-jvm-options

Creates options for Java VM.

Synopsis

```
asadmin [asadmin-options] create-jvm-options [--help] [--target target]
    [--profiler={true|false}]
    option-name[=option-value][:option-name[=option-value]]...
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-jvm-options` subcommand creates command-line options that are passed to Java VM when Java EE Server is started. The options that this command creates are in addition to the options that are preset with Java EE Server. Java VM options are stored in the Java configuration `java--config` element or the `profiler` element of the `domain.xml` file. The options are sent to the command line in the order they appear in the `java--config` element or the `profiler` element in the `domain.xml` file.

This command can be used to create the following types of options:

- Java system properties: These options are set through the -D option of Java VM.
- Startup parameters for Java VM. These options are preceded by the dash character (-).

If an inappropriate value is specified for an option for Java VM of the Domain Administration Server (DAS), you might need to reconfigure the domain without starting the DAS. To avoid this problem, we recommend that before changing options for Java VM on the DAS, use the `backup-domain` command to back up the domain.

This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Files

Java VM options are stored in the Java configuration `java--config` element or the `profiler` `profiler` element of the `domain.xml` file.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target on which you are creating Java VM options.

Type: String

The following values can be specified:

- `server`
  Specifies the DAS (default).
- *configuration-name*
  Specifies a named configuration.
- *cluster-name*
  Specifies a cluster.
- *instance-name*
  Specifies a server instance.

Default value: `server`

`--profiler={true|false}`

Indicates whether the Java VM options are for the profiler.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

option-name`[=`*option-value*`][:`*option-name*`[=`*option-value*`]]...`

Indicates one or more options delimited by a colon (`:`). The format of an option depends on the following:

- If the option has a name and a value, the format is *option-name=value*.
- If the option has only a name, the format is *option-name*.

Type: String

The following values can be specified:

- *Specify the jvm-option name or name with value*

Default value: N/A

Examples

The following example creates Java VM options on the server instance `instance1`.

```
asadmin create-jvm-options --target instance1 -Dunixlocation=/root/example:
-Dvariable=\$HOME:-Dwindowslocation=d\:\\sun\\appserver:-Doption1=-value1
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

Notes

Do not specify the following system properties for the options of this command. If specified, the log configuration (`config.properties`) of the Message Queue is affected.

- `java.util.logging.FileHandler.count`
- `java.util.logging.FileHandler.limit`
- `java.util.logging.FileHandler.level`
- `java.util.logging.FileHandler.pattern`

2.11.2 create-system-properties

Adds one or more system property elements that can be referenced elsewhere in the configuration.

Synopsis

```
asadmin [asadmin-options] create-system-properties [--help]
    [--target target] [name=value[:name=value]...]
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `create-system-properties` subcommand adds or updates system properties that can be referenced elsewhere on the server.

Java EE Server provides hooks where tokens (system properties) can be specified. As Java EE Server does not have multiple server elements, you can specify a particular token at any level. When a domain supports multiple servers, the override potential can be exploited. When a domain is started or restarted, all `system-property` elements are resolved and available to Java VM by using the `System.setProperty()` call on each of them (with its name and value derived from the corresponding attributes of the element). This is analogous to sending the elements as `-D` parameters on the Java command line.

This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target on which you are creating the system properties.

Type: String

The following values can be specified:

- `server`
  Creates the properties on the default server instance. This is the default value.
- `domain`
  Creates the properties for all server instances in the default domain.
- *configuration-name*
  Creates the properties in the specified configuration.
- *cluster-name*
  Creates the properties on all the server instances in the specified cluster.
- *instance_name*
  Creates the properties on a specified server instance.

Default value: `server`

name=*value*[:*name*=*value*]...]
Specifies the name value pairs of the system properties to be added to the specified target. For `name`=*value* operand, multiple system properties must be separated by a colon (`:`). If a colon (`:`) appears in the name or value of a system property, it must be escaped with a backslash (\).

Type: String

The following values can be specified:

- *Name-value pairs of the system properties*

Default value: N/A

Examples

The following example creates a system property associated with an HTTP listener on a server instance named `myserver`.

```
asadmin create-system-properties --target myserver http-listener-port=1088
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

Notes

Do not specify the following system properties for the options of this command. When these options are specified, the log configuration (`config.properties`) of Message Queue is affected.

- `java.util.logging.FileHandler.count`
- `java.util.logging.FileHandler.limit`
- `java.util.logging.FileHandler.level`
- `java.util.logging.FileHandler.pattern`

2.11.3 delete-jvm-options

Removes one or more options for Java VM.

Synopsis

```
asadmin [asadmin-options] delete-jvm-options [--help]
    [--target target] [--profiler={true|false}]
    option-name[=option-value] [:option-name[=option-name]]...
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-jvm-options` subcommand removes one or more command-line options for Java VM. These options are removed from the Java configuration `java-config` element or the `profiler` profiler element of the `domain.xml` file. The deletion of some options requires a server restart for changes to become effective.

Other options are set immediately in the environment of the domain administration server (DAS) and do not require a restart.

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Files

The `delete-jvm-options` subcommand removes one or more command-line options for Java VM. These options are removed from the Java configuration `java—config` element or the `profiler` profiler element of the `domain.xml` file.

Arguments

`--help|-?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target from which you are removing Java VM options.

Type: String

The following values can be specified:

- `server`
  Specifies the DAS (default).
- *configuration-name*
  Specifies a named configuration.
- *cluster-name*
  Specifies a cluster.
- *instance-name*

Specifies a server instance.

Default value: `server`

`--profiler={true|false}`

Indicates whether Java VM options are for the profiler.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`option-name=option-value`

Specifies the One or more options delimited by a colon (`:`).

Notes: The format of the *option-name* operand depends on the following:

- If the option has a name and a value, the format is `option-name=value`.
- If the option has only a name, the format is `option-name`. For example, `-Xmx2048m`.

If an option name or option value contains a colon, the backslash (`\`) must be used to escape the colon in the name or value. Other characters might also require an escape character.

Type: String

The following values can be specified:

- *One or more Java VM options delimited by a colon (:)*

Default value: N/A

Examples

The following example removes multiple Java VM options on the server instance `instance1`.

```
asadmin delete-jvm-options --target instance1 -Doption1=value1
"-Doption1=value1:-Doption2=value2"
```

Exit Status

Exit Status	Explanation
`0`	command executed successfully.
`1`	error in executing the command.

2.11.4 delete-system-property

Removes a system property of the domain, configuration, cluster, or server instance, one at a time.

Synopsis

```
asadmin [asadmin-options] delete-system-property [--help]
    [--target target] property_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-system-property` subcommand deletes a system property of a domain, configuration, cluster, or server instance. Ensure that the system property is not referenced elsewhere in the configuration before deleting it.

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target on which you are deleting the system properties.

Type: String

The following values can be specified:

- `server`
- `domain`
- *configuration_name*
- *cluster_name*
- *instance_name*

Default value: `server`

property_name

Specifies the name of the system property.

Type: String

The following values can be specified:

- *Name of the system property*

Default value: N/A

Examples

The following example deletes the system property named `http-listener-port` on the server instance `instance1`.

```
asadmin delete-system-property --target instance1 http-listener-port
```

Exit Status

Exit Status	Explanation
0	command executed successfully.

Exit Status	Explanation
1	error in executing the command.

2.11.5 get

Retrieves the values of configurable attributes.

Synopsis

```
asadmin [asadmin-options] get [--help]
    attribute-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `get` subcommand uses identifiers targeted by the setting to acquire setup values of Application Server.

For the parameters that can be specified as identifiers targeted by the setting, refer to List of parameters used with the set and get subcommands.

For the PRF-related parameters, if values are set for "`Parameters starting with hitachi-prfs.hitachi-prf.`*`PRF-name`*`.`" and "`Parameters starting with hitachi-prf-configs.hitachi-prf-config.`*`PRF-configuration-name`*", then the value for "`Parameters starting with hitachi-prfs.hitachi-prf.`*`PRF-name`*`.`" is valid.

For the Web server-related parameters, if values are set for "`Parameters starting with hitachi-webservers.hitachi-webserver.`*`web-server-name`*" and "`Parameters starting with hitachi-webserver-configs.hitachi-webserver-config.`*`web-server-configuration-name`*", then the value for "`Parameters starting with hitachi-webservers.hitachi-webserver.`*`web-server-name`*" is valid.

For the server instance-related parameters, if values are set for "`Parameters starting with servers.server.`*`Java-EE-server-name`*" and "`Parameters starting with configs.config.`*`configuration-name-of-Java-EE-server`*", then the value for "`Parameters starting with servers.server.`*`Java-EE-server-name`*" is valid.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`attribute-name`

Specify the identifiers targeted by the setting. Specify at least one identifier targeted by the setting. Identifiers targeted by the setting are used to reference settings of Application Server.

You can use wild cards (*) in identifiers targeted by the setting. When using a wild card (*), enclose the identifier targeted by the setting with straight quotation marks (" ").

Type: String

The following values can be specified:

- *The identifier targeted by the setting*

Default value: N/A

Examples

The following example gets the attributes of `listener.http-listener-1`.

```
asadmin get server.http-service.http-listener.http-listener-1.*
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.11.6 list

Lists the configurable elements.

Synopsis

```
asadmin [asadmin-options] list [--help]
    dotted-parent-attribute-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list` subcommand lists the configurable attributes of Java EE Server.

Arguments

`--help | -?`

Displays the help text for the subcommand.

dotted-parent-element-name

Specifies the configurable element name.

Type: String

The following values can be specified:

- *Dotted Name pattern*

Default value: N/A

Examples

The following example lists the elements that can be configured.

```
asadmin list *
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.11.7 list-configs

Lists the named configurations in the configuration of the Domain Administration Server (DAS).

Synopsis

```
asadmin [asadmin-options] list-configs [--help]
    [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-configs` subcommand lists named configurations in the configuration of the Domain Administration Server (DAS). The list can be filtered by cluster, instance, or named configuration.

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

target

Filters the list of configurations.

Type: String

The following values can be specified:

- `domain`

  Lists all named configurations in the current domain.

- *configuration-name*

  Lists the specified named configuration. Use this option to determine whether a named configuration exits.

- *cluster-name*
  Lists the named configuration that defines the configuration of instances in the specified cluster.
- *instance-name*
  Lists the named configuration that defines the configuration of the specified instance.

Default value: `domain`

Examples

The following example lists all named configurations in the current domain.

```
asadmin list-configs
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.11.8 list-jvm-options

Lists options for Java VM.

Synopsis

```
asadmin [asadmin-options] list-jvm-options [--help]
    [--target target]
    [--profiler={false|true}]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-jvm-options` subcommand of asadmin displays a list of command-line options that are passed to Java VM when Java EE Server is started.

The options are managed using the `create-jvm-options` and `delete-jvm-options` commands. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`
  Displays the help text for the subcommand.

`--target` *target*

Specifies the target for which Java VM options are to be listed.

Type: String

The following values can be specified:

- `server`
  Specifies the DAS. This is the default value.
- *configuration-name*
  Specifies a named configuration.
- *cluster-name*
  Specifies a cluster.
- *instance-name*
  Specifies a server instance.

Default value: `server`

`--profiler={false|true}`

Specifies whether Java VM options to list, are for the profiler.

Set this option to `true` only if a profiler has been configured. If this option is set to `true` and no profiler is configured, an error occurs.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

Examples

The following example lists the options that are used by Java VM on the server instance `instance1`.

```
asadmin list-jvm-options --target instance1
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.11.9 list-system-properties

Lists the system properties of the domain, configuration, cluster, or a server instance.

Synopsis

```
asadmin [asadmin-options] list-system-properties [--help]
    [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-system-properties` subcommand of asadmin lists the system properties of a domain, configuration, cluster, or server instance. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

target

Limits the listing to system properties for a specific target.

Type: String

The following values can be specified:

- `domain`

  Lists the system properties defined for the domain.

- *configuration_name*

  Lists the system properties for the named configuration as well as for those which, the cluster inherits from the domain.

- *cluster_name*

  Lists the system properties defined for the named cluster as well as for those which, the cluster inherits from its configuration and the domain.

- *instance_name*

  Lists the system properties defined for the named server instance as well as for those which, the server inherits from its cluster (if the instance is clustered), its configuration, and the domain.

- `server`

  Lists the system properties defined for the DAS. Server is the name of the DAS.

Default value: `server`

Examples

The following example lists the system properties on localhost.

```
asadmin list-system-properties
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.11.10 set

Sets the values of configurable attributes.

Synopsis

```
asadmin [asadmin-options] set [--help]
    attribute-name=value
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `set` subcommand uses identifiers targeted by the setting to modify the values of the settings for Application Server. For the parameters that can be specified as identifiers targeted by the setting, refer to List of parameters used with the set and get subcommands. Characters that have special meaning to the shell or command interpreter, such as an asterisk (`*`), must be quoted or escaped as appropriate to the shell, for example, by enclosing the argument in quotes. In multimode, quotes are needed only for arguments that include spaces, quotes, or backslash. By modifying attributes, we can enable and disable services, and customize how an existing element functions. Any changes made by using the `asadmin` utility commands are automatically applied to the associated Java EE Server configuration file.

To apply the set values, after executing this subcommand, restart the servers to which the settings are to be applied (domain administration server, web server, Java EE server, and performance tracer). If the set values failed to be made persistent due to a file write failure or other failures, the command can still be successfully completed, but the set values might not be applied after the restart. When the set values failed to be made persistent, an error is output to the `server.log` of the domain administration server.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

attribute-name=value

Specify the identifier targeted by the setting and its value.

Type: String

The following values can be specified:

- *The identifiers targeted by the setting and their values*

Default value: N/A

Examples

The following example changes the steady pool size of the `DerbyPool` connection pool to `9`.

```
asadmin set resources.jdbc-connection-pool.DerbyPool.steady-pool-size=9
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.11.11 List of parameters used with the set and get subcommands

This topic explains parameters that can be specified as the identifiers to be set for the `set` and `get` subcommands.

Resource relation

Parameters starting with `applications.application.`*`application-name`*`.module.`*`module-name`*`.resources`

No.	Parameter name	Description	Range Value	Default Value	Initial Value	Example
1	`connector-connection-pool.`*`connector_connection_pool_name`*`.hitachi-connection-sharing-component-enabled`	Enables or disables connection sharing within components. This setting is made in the connector connection pool in the module scope.	`true`|`false`	No Default Value	`false`	`asadmin set applications.application.myapplication.module.mymodule.resources.connector-connection-pool.myconnectorconnectionpool.hitachi-connection-sharing-component-enabled=false`
2	`jdbc-connection-pool.`*`JDBC_connection_pool_name`*`.hitachi-connection-sharing-component-enabled`	Enables or disables connection sharing within components. This setting is made in the JDBC connection pool in the module scope.	`true`|`false`	No Default Value	`false`	`asadmin set applications.application.myapplication.module.mymodule.resources.jdbc-connection-pool.myjdbcconnectionpool.hitachi-connection-sharing-component-enabled=false`
3	`jdbc-connection-pool.`*`JDBC_connection_pool_name`*`.hitachi-jdbc-transaction-rollback-enabled`	Enables or disables rollback of JDBC transactions when a connection closes. This setting is made in the JDBC connection pool in the module scope.	`true`|`false`	No Default Value	`true`	`asadmin set applications.application.myapplication.module.mymodule.resources.jdbc-connection-pool.myjdbcconnectionpool.hitachi-jdbc-transaction-`

No.	Parameter name	Description	Range Value	Default Value	Initial Value	Example
						rollback-enabled=true
4	jdbc-connection-pool.*JDBC_connection_pool_name*.hitachi-validation-timeout	Specifies the timeout time in seconds for connection error detection. This setting is made in the JDBC connection pool in the module scope.	1 to 2147483647	No Default Value	5	asadmin set applications.application.myapplication.module.mymodule.resources.jdbc-connection-pool.myjdbcconnectionpool.hitachi-validation-timeout=5
5	jdbc-connection-pool.*JDBC_connection_pool_name*.hitachi-validation-timeout-enabled	Enables or disables timeout for connection error detection. This setting is made in the JDBC connection pool in the module scope.	true|false	No Default Value	true	asadmin set applications.application.myapplication.module.mymodule.resources.jdbc-connection-pool.myjdbcconnectionpool.hitachi-validation-timeout-enabled=true

Parameters starting with applications.application.*application-name*.resources

No.	Parameter name	Description	Range Value	Default Value	Initial Value	Example
1	connector-connection-pool.*connector_connection_pool_name*.hitachi-connection-sharing-component-enabled	Enables or disables connection sharing within components. This setting is made in the connector connection pool in the application scope.	true|false	No Default Value	false	asadmin set applications.application.myapplication.resources.connector-connection-pool.myconnectorconnectionpool.hitachi-connection-sharing-component-enabled=false
2	jdbc-connection-pool.*JDBC_connection_pool_name*.hitachi-connection-sharing-component-enabled	Enables or disables connection sharing within components. This setting is made in the JDBC connection pool in the application scope.	true|false	No Default Value	false	asadmin set applications.application.myapplication.resources.jdbc-connection-pool.myjdbcconnectionpool.hitachi-connection-sharing-component-enabled=false
3	jdbc-connection-	Enables or disables rollback of JDBC	true|false	No Default Value	true	asadmin set applications.ap

No.	Parameter name	Description	Range Value	Default Value	Initial Value	Example
	pool.*JDBC_connection_pool_name*.hitachi-jdbc-transaction-rollback-enabled	transactions when a connection closes. This setting is made in the JDBC connection pool in the application scope.				plication.myapplication.resources.jdbc-connection-pool.myjdbcconnectionpool.hitachi-jdbc-transaction-rollback-enabled=true
4	jdbc-connection-pool.*JDBC_connection_pool_name*.hitachi-validation-timeout	Specifies the timeout time in seconds for connection error detection. This setting is made in the JDBC connection pool in the application scope.	1 to 2147483647	No Default Value	5	asadmin set applications.application.myapplication.resources.jdbc-connection-pool.myjdbcconnectionpool.hitachi-validation-timeout=5
5	jdbc-connection-pool.*JDBC_connection_pool_name*.hitachi-validation-timeout-enabled	Enables or disables timeout for connection error detection. This setting is made in the JDBC connection pool in the application scope.	true|false	No Default Value	true	asadmin set applications.application.myapplication.resources.jdbc-connection-pool.myjdbcconnectionpool.hitachi-validation-timeout-enabled=true

Parameters starting with resources

No.	Parameter name	Description	Range Value	Default Value	Initial Value	Example
1	connector-connection-pool.*connector_connection_pool_name*.hitachi-connection-sharing-component-enabled	Enables or disables connection sharing within components. This setting is made in the connector connection pool created in the domain.	true|false	No Default Value	false	asadmin set resources.connector-connection-pool.myconnectorconnectionpool.hitachi-connection-sharing-component-enabled=false
2	connector-connection-pool.*connector_connection_pool_name*.idle-timeout-in-seconds	Set the maximum time that a connection can remain idle in the pool.	1 to 2147483647	300	No Value Specified	asadmin set resources.connector-connection-pool.ConnectionPool1.idle-timeout-in-seconds=200

No.	Parameter name	Description	Range Value	Default Value	Initial Value	Example
3	connector-connection-pool.*connector_connection_pool_name*.max-pool-size	Set the maximum number of connections that can be created to satisfy client requests.	1 to 2147483647	24	No Value Specified	asadmin set resources.connector-connection-pool.ConnectionPool1.max-pool-size=60
4	connector-connection-pool.*connector_connection_pool_name*.max-wait-time-in-millis	Set the amount of time, in milliseconds, that a caller must wait before a connection is created, if a connection is not available.	0 to 2147483647	60000	No Value Specified	asadmin set resources.connector-connection-pool.ConnectionPool1.max-wait-time-in-millis=700000
5	connector-connection-pool.*connector_connection_pool_name*.pool-resize-quantity	Set the quantity by which the pool will scale up or scale down the number of connections.	1 to 2147483647	2	No Value Specified	asadmin set resources.connector-connection-pool.ConnectionPool1.pool-resize-quantity=4
6	connector-connection-pool.*connector_connection_pool_name*.property.*property_name*	Set the optional attribute name/value pairs for configuring the pool.	Type: String	No Default Value	No Value Specified	asadmin set resources.connector-connection-pool.ConnectionPool1.property.User=User1
7	connector-connection-pool.*connector_connection_pool_name*.steady-pool-size	The minimum and initial number of connections maintained in the pool.	0 to 2147483647	24	No Value Specified	asadmin set resources.connector-connection-pool.ConnectionPool1.steady-pool-size=30
8	jdbc-connection-pool.*JDBC_connection_pool_name*.hitachi-connection-sharing-component-enabled	Enables or disables connection sharing within components. This setting is made in the JDBC connection pool created in the domain.	true|false	No Default Value	false	asadmin set resources.jdbc-connection-pool.myjdbcconnectionpool.hitachi-connection-sharing-component-enabled=false
9	jdbc-connection-pool.*JDBC_connection_pool_name*.hitachi-jdbc-transaction-rollback-enabled	Enables or disables rollback of JDBC transactions when a connection closes. This setting is made in the JDBC connection pool created in the domain.	true|false	No Default Value	true	asadmin set resources.jdbc-connection-pool.myjdbcconnectionpool.hitachi-jdbc-transaction-rollback-enabled=true

No.	Parameter name	Description	Range Value	Default Value	Initial Value	Example
10	jdbc-connection-pool.*JDBC_connection_pool_name*.hitachi-validation-timeout	Specifies the timeout time in seconds for connection error detection. This setting is made in the JDBC connection pool created in the domain.	1 to 2147483647	No Default Value	5	asadmin set resources.jdbc-connection-pool.myjdbcconnectionpool.hitachi-validation-timeout=5
11	jdbc-connection-pool.*JDBC_connection_pool_name*.hitachi-validation-timeout-enabled	Enables or disables timeout for connection error detection. This setting is made in the JDBC connection pool created in the domain.	true|false	No Default Value	true	asadmin set resources.jdbc-connection-pool.myjdbcconnectionpool.hitachi-validation-timeout-enabled=true
12	jdbc-connection-pool.*JDBC_connection_pool_name*.idle-timeout-in-seconds	Set the maximum time, in seconds, that a connection can remain idle in the pool. This timeout value must be kept shorter than the database server side timeout value to prevent the accumulation of unusable connections in the application.	0 to 2147483647	300	No Value Specified	asadmin set resources.jdbc-connection-pool.ConnectionPool1.idle-timeout-in-seconds=200
13	jdbc-connection-pool.*JDBC_connection_pool_name*.max-pool-size	Set the maximum number of connections that can be created.	1 to 2147483647	24	No Value Specified	asadmin set resources.jdbc-connection-pool.ConnectionPool1.max-pool-size=60
14	jdbc-connection-pool.*JDBC_connection_pool_name*.max-wait-time-in-millis	Set the amount of time, in milliseconds, that a caller will wait before a JDBC connection timeout is sent. A value of 0 forces the caller to wait indefinitely.	0 to 2147483647	60000	No Value Specified	asadmin set resources.jdbc-connection-pool.ConnectionPool1.max-wait-time-in-millis=70000
15	jdbc-connection-pool.*JDBC_connection_pool_name*.pool-resize-quantity	Set the number of connections to be removed when idle-timeout-in-seconds timer expires.	1 to 2147483647	2	No Value Specified	asadmin set resources.jdbc-connection-pool.ConnectionPool1.pool-resize-quantity=4
16	jdbc-connection-pool.*JDBC_connection_pool_name*.property.*property_name*	Set the optional attribute name/value pairs for configuring the pool.	Type: String	No Default Value	No Value Specified	asadmin set resources.jdbc-connection-pool.ConnectionPool1.property.User=User1

No.	Parameter name	Description	Range Value	Default Value	Initial Value	Example
17	jdbc-connection-pool.*JDBC_connection_pool_name*.steady-pool-size	Set the minimum and initial number of connections maintained in the pool.	0 to 2147483647	24	No Value Specified	asadmin set resources.jdbc-connection-pool.ConnectionPool1.steady-pool-size=30
18	mail-resource.*JavaMail_resource_name*.property.*property_name*	Set the optional property name/value pairs for configuring the JavaMail resource.	Type: String	No Default Value	No Value Specified	asadmin set resources.mail-resource.mymailresource.property.mail-from=xyz@abc.com

PRF relation

Parameters starting with hitachi-prfs.hitachi-prf.*PRF-name*.

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
1	property.*ex_property*	Specifies the name and the value of an extended property to be added to PRF. The property name must start with ex_. The added extended property can be used as a replacement character string to be written in PRF server template files.	Type: String	Property deletion	No Value Specified	asadmin set hitachi-prfs.hitachi-prf.PRF01.property.ex_timeout=600
2	property.env_*environment_variables_name*	Specifies the environment variable to be set when PRF starts. If you want to specify more than one environment variable, you must specify this standard property more than once. If the specified environment variable value includes ${apserver.home}, it will be replaced by *Application Server installation directory*. The environment variable value will not be inherited even if this property value contains a setting such as %PATH%;C:\temp or ${PATH}:/temp, which indicates that the	Type: String Only single-byte alphanumeric characters and the following symbols can be specified: percent signs (%), at-signs (@), plus symbols (+), hyphens (-), underscores (_), commas (,), and periods (.).	Property deletion	No Value Specified	asadmin set hitachi-prfs.hitachi-prf.PRF01.property.env_TZ=JST-9

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		environment variable value should be inherited. If you wish to specify such a value, make sure that you specify it in one of the following: • System environment variable of the operating system on the node • asenv definition on the node • Startup batch or shell script for the server template If you need to set characters that cannot be specified for this property to the server environment variable, you must set the environment variable as explained below. • To set the environment variable, create an extended property on the server and then make sure that the value to be set for the environment variable is assigned to this extended property. An example of such an extended property is shown below. Extended property name: `ex_myenv` Extended property value: *value-to-be-set-for-environment-variable* • Add lines of code to the script that exists in the server template and starts the server. For the environment variable, these lines of code must set the value of the extended property.				

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example

		The following is an example of lines of code to be added:
		`MY_ENV=${property.ex_myenv.value}`
		`export MY_ENV`				

3	`property.log-file-count`	Specifies the number of log files to be output by PRF. This property corresponds to the `-PrfLogFileCount` option of the PRF `cprfstart` command.	`1` to `32`	Property deletion	No Value Specified	`asadmin set hitachi-prfs.hitachi-prf.PRF01.property.log-file-count=16`
4	`property.log-file-size`	Specifies in megabytes the largest possible size of log files to be output by PRF. This property corresponds to the `-PrfLogFileSize` option of the PRF `cprfstart` command.	`1` to `100`	Property deletion	No Value Specified	`asadmin set hitachi-prfs.hitachi-prf.PRF01.property.log-file-size=20`
5	`property.log-shift-time`	Specifies the shift time for log files to be output by PRF. This property corresponds to the `-PrfLogShiftTime` option of the PRF `cprfstart` command.	*HHMMSS*(`000000` to `235959`)	Property deletion	No Value Specified	`asadmin set hitachi-prfs.hitachi-prf.PRF01.property.log-shift-time=121530`
6	`property.prf-spool`	Specifies the output directory for the PRF trace.	Type: String	Property deletion	No Value Specified	`asadmin set hitachi-prfs.hitachi-prf.PRF01.property.prf-spool=C:\PRFSPOOL\PRF01`
7	`property.trace-buffer-size`	Specifies in kilobytes the buffer size for PRF trace files, to be allocated to shared memory. This property corresponds to the `-PrfTraceBufferSize` option of the PRF `cprfstart` command.	`512` to `102400`	Property deletion	No Value Specified	`asadmin set hitachi-prfs.hitachi-prf.PRF01.property.trace-buffer-size=16384`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
8	property.trace-count	Specifies the number of PRF trace files. This property corresponds to the -PrfTraceCount option of the PRF cprfstart command.	3 to 256	Property deletion	No Value Specified	asadmin set hitachi-prfs.hitachi-prf.PRF01.property.trace-count=8
9	property.trace-file-size	Specifies in kilobytes the maximum possible size of one PRF trace file. This property corresponds to the -PrfTraceFileSize option of the PRF cprfstart command.	1024 to 1048576	Property deletion	No Value Specified	asadmin set hitachi-prfs.hitachi-prf.PRF01.property.trace-file-size=16384
10	property.trace-level	Specifies the level of obtaining the PRF trace by using a four-byte hexadecimal number (eight digits). You can add 0x to the beginning, but 0x will be ignored. You can specify multiple values by using commas (,) at the left to separate the level values of index numbers. This property corresponds to the -PrfTraceLevel option of the PRF cprfstart command.	Four-byte hexadecimal number (eight digits)[,four-byte hexadecimal number (eight digits)...]	Property deletion	No Value Specified	asadmin set hitachi-prfs.hitachi-prf.PRF01.property.trace-level=0x44445555
11	hitachi-prf-config-ref	Specifies a PRF configuration element. This attribute value cannot be changed by the set subcommand.	PRF configuration name that exists in the domain	Cannot be changed	No Value Specified	Cannot be changed

Legend:

#: If "Property deletion" is shown in the "Default Value" column, specifying an empty string for the parameter deletes the property itself.

Parameters starting with hitachi-prf-configs.hitachi-prf-config.*PRF-configuration-name*

If a parameter whose name begins with property. is also defined by Parameters starting with hitachi-prfs.hitachi-prf.*PRF-name*, the value defined by Parameters starting with hitachi-prfs.hitachi-prf.*PRF-name* has priority.

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
1	hitachi-manage-info.construct-command	Specifies the relative path (without the extension) from the template directory of each server type to the build command to be executed by PRF.	Type: String	There is no command to be executed.	<empty string>	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.hitachi-manage-info.construct-command=create-prf
2	hitachi-manage-info.destruct-command	Specifies the relative path (without the extension) from the template directory of each server type to the deletion command to be executed by PRF.	Type: String	There is no command to be executed.	<empty string>	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.hitachi-manage-info.destruct-command=delete-prf
3	hitachi-manage-info.get-pid-command	Specifies the relative path (without the extension) from the template directory of each server type to the process ID acquisition command to be executed by PRF.	Type: String	Cannot be omitted	bin/getpid-prf	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.hitachi-manage-info.get-pid-command=getpid-prf
4	hitachi-manage-info.running-watch-interval-in-seconds	Specifies in seconds the interval at which PRF operations are verified.	1 to 86400	Cannot be omitted	10	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.hitachi-manage-info.running-watch-interval-in-seconds=20
5	hitachi-manage-info.start-command	Specifies the relative path (without the extension) from the template directory of each server type to the startup command to be executed by PRF.	Type: String	Cannot be omitted	bin/start-prf	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.hitachi-manage-info.start-command=start-prf
6	hitachi-manage-info.start-timeout-in-seconds	Specifies in seconds the time during which to monitor whether PRF is started. If 0 is specified, the monitoring will not be performed.	0 to 3600	Cannot be omitted	60	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.hitachi-manage-

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
						info.start-timeout-in-seconds=180
7	hitachi-manage-info.start-type	Specifies how to start PRF. • direct: Direct startup	direct	Cannot be omitted	direct	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.hitachi-manage-info.start-type=direct
8	hitachi-manage-info.starting-watch-interval-in-seconds	Specifies in seconds the interval for monitoring whether PRF is started.	1 to 86400	Cannot be omitted	1	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.hitachi-manage-info.starting-watch-interval-in-seconds=3
9	hitachi-manage-info.starting-watch-start-time-in-seconds	Specifies the time (in seconds) from executing the startup command by PRF until starting to check the operation.	0 to 86400	Cannot be omitted	0	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.hitachi-manage-info.starting-watch-start-time-in-seconds=10
10	hitachi-manage-info.stop-command	Specifies the relative path (without the extension) from the template directory of each server type to the stop command to be executed by PRF.	Type: String	Cannot be omitted	bin/stop-prf	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.hitachi-manage-info.stop-command=stop-prf
11	hitachi-manage-info.stop-timeout-in-seconds	Specifies in seconds the time during which to monitor whether PRF is stopped. If 0 is specified, the monitoring will not be performed.	0 to 1800	Cannot be omitted	60	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.hitachi-manage-info.stop-timeout-in-seconds=180
12	hitachi-manage-info.stopping-watch-interval-in-seconds	Specifies in seconds the interval for monitoring whether PRF is stopped.	1 to 86400	Cannot be omitted	1	asadmin set hitachi-prf-configs.hitachi-prf-

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
						config.PRF01-config.hitachi-manage-info.stopping-watch-interval-in-seconds=3
13	hitachi-manage-info.template-path	Specifies the absolute path of the server template for PRF.	An existing directory path	Cannot be omitted	*domain root directory*/*domain name*/server_templates/*server type*	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.hitachi-manage-info.template-path=C:\temp\prf
14	hitachi-manage-info.working-dir	Specifies the absolute path to the current directory of each command to be executed by PRF.	Type: String	*node-directory*/*server-name*	<empty string>	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.hitachi-manage-info.working-dir=C:\temp\prf\command
15	property.*ex_property*	Specifies the name and the value of an extended property to be added to PRF. The property name must start with ex_. The added extended property can be used as a replacement character string to be written in PRF server template files.	Type: String	Property deletion	No Value Specified	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.property.ex_timeout=600
16	property.env_*environment_variables_name*	Specifies the environment variable to be set when PRF starts. If you want to specify more than one environment variable, you must specify this standard property more than once. If the specified environment variable value includes ${apserver.home}, it will be replaced by *Application Server installation directory*. The environment variable value will not be inherited even if this property value	Type: String Single-byte alphanumeric characters and the following symbols can only be specified: percent signs (%), at-signs (@), plus symbols (+), hyphens (-), underscores (_), commas (,), and periods (.).	Property deletion	No Value Specified	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.property.env_TZ=JST-9

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		contains a setting such as *%PATH%;C:\temp* or *${PATH}:/temp*, which indicates that the environment variable value should be inherited. If you wish to specify such a value, make sure that you specify it in one of the following: • System environment variable of the operating system on the node • asenv definition on the node • Startup batch or shell script for the server template If you need to set characters that cannot be specified for this property to the server environment variable, you must set the environment variable as explained below. • To set the environment variable, create an extended property on the server and then make sure that the value to be set to the environment variable is assigned to this extended property. An example of such an extended property is shown below. Extended property name: `ex_myenv` Extended property value: *value-to-be-set-to-environment-variable* • Add lines of code to the script that exists in the server template and starts the server. These lines of code must set the value of the extended property				

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example

		to the environment variable.
		The following is an example of lines of code to be added:
		`MY_ENV=${property.ex_myenv.value}`
		`export MY_ENV`				

17	`property.log-file-count`	Specifies the number of log files to be output by PRF. This property corresponds to the `-PrfLogFileCount` option of the PRF `cprfstart` command.	`1` to `32`	Property deletion	`8`	`asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.property.log-file-count=16`
18	`property.log-file-size`	Specifies in megabytes the largest possible size of log files to be output by PRF. This property corresponds to the `-PrfLogFileSize` option of the PRF `cprfstart` command.	`1` to `100`	Property deletion	`10`	`asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.property.log-file-size=20`
19	`property.log-shift-time`	Specifies the time to shift the log files output by PRF. This property corresponds to the `-PrfLogShiftTime` option of the PRF `cprfstart` command.	*HHMMSS*(`000000` to `235959`)	Property deletion	`000000`	`asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.property.log-shift-time=121530`
20	`property.prf-spool`	Specifies the output directory for the PRF trace.	Type: String	Property deletion	No Value Specified	`asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.property.prf-spool=C:\PRFSPOOL\PRF01`
21	`property.trace-buffer-size`	Specifies in kilobytes the buffer size for PRF trace files, to be allocated to shared memory. This property corresponds to the `-PrfTraceBufferSize` option of the PRF	`512` to `102400`	Property deletion	`8192`	`asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.property.trace-buffer-size=16384`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		cprfstart command.				
22	property.trace-count	Specifies the number of PRF trace files. This property corresponds to the -PrfTraceCount option of the PRF cprfstart command.	3 to 256	Property deletion	4	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.property.trace-count=8
23	property.trace-file-size	Specifies in kilobytes the maximum possible size of one PRF trace file. This property corresponds to the -PrfTraceFileSize option of the PRF cprfstart command.	1024 to 1048576	Property deletion	262144	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.property.trace-file-size=16384
24	property.trace-level	Specifies the level of obtaining the PRF trace by using a four-byte hexadecimal number (eight digits). You can add 0x to the beginning, but 0x will be ignored. You can specify multiple values by using commas (,) at the left to separate the level values of index numbers. This property corresponds to the -PrfTraceLevel option of the PRF cprfstart command.	Four-byte hexadecimal number (eight digits)[,four-byte hexadecimal number (eight digits)...]	Property deletion	0	asadmin set hitachi-prf-configs.hitachi-prf-config.PRF01-config.property.trace-level=0x44445555

Legend:

#: If "Property deletion" is shown in the "Default Value" column, specifying an empty string for the parameter deletes the property itself.

Web server relation

Parameters starting with hitachi-webservers.hitachi-webserver.*web-server-name*

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
1	hitachi-relations.*hitachi-relation-type-ref*.hitachi-	Specifies the sequence number of an inter-server relation.	0 to 65535	Cannot be omitted	No Value Specified	asadmin set hitachi-webservers.hitachi-webserver.WEB01

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
	relation.*relation_name*.order					.hitachi-relations.redirect.hitachi-relation.redirect1.order=10
2	hitachi-relations.prf-relation.hitachi-relation.*relation_name*.property.*ex_property*	Specifies the name and value of an extended property to be added to the web-server PRF relation. The property name must start with ex_. The added extended property can be used as a replacement character string to be written in server template files for the web server that is the source of the inter-server relation.	Type: String	Property deletion	No Value Specified	asadmin set hitachi-webservers.hitachi-webserver.WEB01.hitachi-relations.prf-relation.hitachi-relation.prfref2.property.ex_timeout=600
3	hitachi-relations.redirect.hitachi-relation.*relation_name*.property.*ex_property*	Specifies the name and value of an extended property to be added to a redirection relation. The property name must start with ex_. The added extended property can be used as a replacement character string to be written in server template files for the web server that is the source of the inter-server relation.	Type: String	Property deletion	No Value Specified	asadmin set hitachi-webservers.hitachi-webserver.WEB01.hitachi-relations.redirect.hitachi-relation.redirector1.property.ex_timeout=600
4	hitachi-relations.redirect.hitachi-relation.*relation_name*.property.negative	Specifies, in the redirector behavior definition, whether to exclude requests to the mapping definition. This property corresponds to the ! option of the ProxyPass directive of Web Server.	true|false	Property deletion	false	asadmin set hitachi-webservers.hitachi-webserver.WEB01.hitachi-relations.redirect.hitachi-relation.redirector1.property.negative=true
5	hitachi-relations.redirect.hitachi-relation.*relation_name*.property.network-listener	Specifies the network listener name that is to be the redirection destination. This property corresponds to the URL of the ProxyPass directive of Web Server. The host name and the port number identified from	A character string representing the network listener name	Property deletion	http-listener-1	asadmin set hitachi-webservers.hitachi-webserver.WEB01.hitachi-relations.redirect.hitachi-relation.redirector1.property.network-

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		the specified network listener name will be set to the URL.				`listener=http-listener-1`
6	`hitachi-relations.redirect.hitachi-relation.`*`relation_name`*`.property.path`	Specifies, in the redirector behavior definition, the name of the local virtual path of the mapping definition. This property corresponds to the pathname specification of the `ProxyPass` directive of Web Server.	Format of `path` (the name of the local virtual path) defined in `ProxyPass` of the `httpsd.conf` file	Property deletion	`/`	`asadmin set hitachi-webservers.hitachi-webserver.WEB01.hitachi-relations.redirect.hitachi-relation.redirector1.property.path=/news/`
7	`hitachi-relations.redirect.hitachi-relation.`*`relation_name`*`.property.proxy-pass-option`	Specifies, in the redirector behavior definition, the adjustment parameter for the connection pooling in the mapping definition. This property corresponds to the "*key=value*" specification of the `ProxyPass` directive of Web Server.	Format of the adjustment parameter for the connection pooling defined in `ProxyPass` of the `httpsd.conf` file ([*key=valuekey=value* ...])	Property deletion	<empty string>	`asadmin set hitachi-webservers.hitachi-webserver.WEB01.hitachi-relations.redirect.hitachi-relation.redirector1.property.proxy-pass-option=timeout=300`
8	`hitachi-relations.redirect.hitachi-relation.`*`relation_name`*`.property.set-proxy-pass-reverse-cookie`	Specifies, in the redirector behavior definition, whether to re-assign the Set-Cookie header. This property corresponds to the `HWSProxyPassReverseCookie` directive of Web Server.	`true`|`false`	Property deletion	`true`	`asadmin set hitachi-webservers.hitachi-webserver.WEB01.hitachi-relations.redirect.hitachi-relation.redirector1.property.set-proxy-pass-reverse-cookie=true`
9	`hitachi-relations.redirect.hitachi-relation.`*`relation_name`*`.property.stickysession`	Specifies, in the redirector behavior definition, whether to enable Sticky. This property corresponds to the `stickysession` key of the `ProxyPass` directive of Web Server. If this property is `true`, the required `Header` directive is added.	`true`|`false`	Property deletion	`true`	`asadmin set hitachi-webservers.hitachi-webserver.WEB01.hitachi-relations.redirect.hitachi-relation.redirector1.property.stickysession=true`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
10	`hitachi-webserver-config-ref`	Specifies the the web server config name. This attribute value cannot be changed by the `set` subcommand.	The config name of the web server that exists in the domain	Cannot be changed	No Value Specified	Cannot be edited
11	`property.`*`ex_property`*	Specifies the name and the value of an extended property to be added to the web server. The property name must start with `ex_`. The added extended property can be used as a replacement character string to be written in server template files for the web server.	Type: String	Property deletion	No Value Specified	`asadmin set hitachi-webservers.hitachi-webserver.WEB01.property.ex_timeout=600`
12	`property.user`	Specifies the user name for Web Server's server process operations. This property corresponds to the `User` directive of Web Server.	Type: String	Property deletion	No Value Specified	`asadmin set hitachi-webservers.hitachi-webserver.Web01.property.user=user01`
13	`property.group`	Specifies the group name for Web Server's server process operations. This property corresponds to the `Group` directive of Web Server.	Type: String	Property deletion	No Value Specified	`asadmin set hitachi-webservers.hitachi-webserver.Web01.property.group=group01`
14	`property.env_`*`environment_variables_name`*	Specifies the environment variable to be set when the web server starts. If you want to specify more than one environment variable, you must specify this standard property more than once. If the specified environment variable value includes `${apserver.home}`, it will be replaced by *Application Server installation directory*. The environment variable value will not be inherited even if this property value contains a setting such as `%PATH%;C:\temp` or `$`	Type: String Single-byte alphanumeric characters and the following symbols can only be specified: percent signs (`%`), at-signs (`@`), plus symbols (`+`), hyphens (`-`), underscores (`_`), commas (`,`), and periods (`.`).	Property deletion	No Value Specified	`asadmin set hitachi-webservers.hitachi-webserver.WEB01.property.env_TZ=JST-9`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		`{PATH}:/temp`, which indicates that the environment variable value should be inherited. If you wish to specify such a value, make sure that you specify it in one of the following: • System environment variable of the operating system on the node • asenv definition on the node • Startup shell script for the server template If you need to set characters that cannot be specified for this property to the server environment variable, you must set the environment variable as explained below. • To set the environment variable, create an extended property on the server and then make sure that the value to be set to the environment variable is assigned to this extended property. An example of such an extended property is shown below. Extended property name: `ex_myenv` Extended property value: *value-to-be-set-to-environment-variable* • Add lines of code to the script that exists in the server template and starts the server. These lines of code must set the value of the extended property to the environment variable.				

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example

		The following is an example of lines of code to be added:
		`MY_ENV=${property.ex_myenv.value}`
		`export MY_ENV`				

15	`property.listen-add-count`	Specify the number of Listen directives to be added if more than one Listen directive needs to be specified. This property specifies the number of directives to be added except for the Listen directive setting specified in the `listen-port` and `listen-host` properties. Therefore, the total number of Listen directives is calculated by adding one to the value specified in this property. The default is `0`. This property indicates how many times the following properties must be added: • `listen-add_port`*n*# • `listen-add_host`*n*# #: *n*: The minimum value is 1. The maximum value is the value specified in this property.	`0` to `2147483647`	Property deletion	No Value Specified	`asadmin set hitachi-webservers.hitachi-webserver.WEB01.property.listen-add-count=2`
16	`property.listen-add-host`*n*	Specifies the IP address of a Listen directive to be added. *n* in the property name must be a value within the range from 1 to `listen-add-count`. A Listen directive is specified by a pair of this and `listen-add-port`*n*, in which *n* is the same number.	*host-name*|*IPv4-dot-notation*|*IPv6-dot-notation (The IPv6 address must be enclosed by angle brackets [].)*	Property deletion	No Value Specified	`asadmin set hitachi-webservers.hitachi-webserver.WEB01.property.listen-add-host1=192.168.1.102`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		This property can be omitted. If *n* is outside the range from 1 to `listen-add-count`, the port number is not applicable to the web server. This property corresponds to the `Listen` directive of Web Server.				
17	`property.listen-add-port`*n*	Specifies the port number of a Listen directive to be added. *n* in the property name must be a value within the range from 1 to `listen-add-count`. A Listen directive is specified by a pair of this and `listen-add-host`*n*, in which *n* is the same number. If *n* is outside the range from 1 to `listen-add-count`, the port number is not applicable to the web server. This property corresponds to the `Listen` directive of Web Server.	`1` to `65535`	Property deletion	No Value Specified	`asadmin set hitachi-webservers.hitachi-webserver.WEB01.property.listen-add-port1=82`
18	`property.listen-host`	Specifies the IP address that will receive requests. This property can be omitted. This property corresponds to the `Listen` directive of Web Server.	*host-name*|*IPv4-dot-notation*|*IPv6-dot-notation (The IPv6 address must be enclosed by angle brackets [].)*	Property deletion	No Value Specified	`asadmin set hitachi-webservers.hitachi-webserver.WEB01.property.listen-host=192.168.1.101`
19	`property.listen-port`	Specifies the port number that will receive requests. This property corresponds to the `Listen` directive of Web Server.	`1` to `65535`	Property deletion	No Value Specified	`asadmin set hitachi-webservers.hitachi-webserver.WEB01.property.listen-port=81`
20	`property.max-request-workers`	Specifies the maximum possible number of clients that can be connected concurrently.	`1` to `1024`	Property deletion	No Value Specified	`asadmin set hitachi-webservers.hitachi-webserver.WEB01`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		This property corresponds to the `MaxRequestWorkers` directive of Web Server.				`.property.max-request-workers=300`
21	`property.server-name`	Specifies the server name and the port number of Web Server. This property corresponds to the `ServerName` directive of Web Server.	Type: String	Property deletion	No Value Specified	`asadmin set hitachi-webservers.hitachi-webserver.WEB01.property.server-name=www.host1.co.jp`
22	`property.start-servers`	Specifies the number of server processes at web server startup. This property corresponds to the `StartServers` directive of Web Server.	`1` to `1024`	Property deletion	No Value Specified	`asadmin set hitachi-webservers.hitachi-webserver.WEB01.property.start-servers=10`
23	`property.threads-per-child`	Specifies the number of threads to be started as servers. This property corresponds to the `ThreadsPerChild` directive of Web Server.	`1` to `1024`	Property deletion	No Value Specified	`asadmin set hitachi-webservers.hitachi-webserver.WEB01.property.threads-per-child=100`

Legend:

#: If "Property deletion" is shown in the "Default Value" column, specifying an empty string for the parameter deletes the property itself.

Parameters starting with `hitachi-webserver-configs.hitachi-webserver-config.`*`web-server-configuration-name`*

If a parameter whose name begins with `property.` is also defined by `Parameters starting with hitachi-webservers.hitachi-webserver.`*`web-server-name`*, the value defined by `Parameters starting with hitachi-webservers.hitachi-webserver.`*`web-server-name`* has priority.

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
1	`hitachi-manage-info.construct-command`	Specifies the relative path (without the extension) from the template directory of each server type to the build command to be executed by the web server.	Type: String	There is no command to be executed.	`bin/create-webserver`	`asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.hitachi-manage-info.construct-command=create-webserver`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
2	hitachi-manage-info.destruct-command	Specifies the relative path (without the extension) from the template directory of each server type to the deletion command to be executed by the web server.	Type: String	There is no command to be executed.	bin/delete-webserver	asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.hitachi-manage-info.destruct-command=delete-webserver
3	hitachi-manage-info.get-pid-command	Specifies the relative path (without the extension) from the template directory of each server type to the process ID acquisition command to be executed by the web server.	Type: String	Cannot be omitted	bin/getpid-webserver	asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.hitachi-manage-info.get-pid-command=getpid-webserver
4	hitachi-manage-info.running-watch-interval-in-seconds	Specifies in seconds the interval at which web server operations are verified.	1 to 86400	Cannot be omitted	10	asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.hitachi-manage-info.running-watch-interval-in-seconds=20
5	hitachi-manage-info.start-command	Specifies the relative path (without the extension) from the template directory of each server type to the startup command to be executed by the web server.	Type: String	Cannot be omitted	bin/start-webserver	asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.hitachi-manage-info.start-command=start-webserver
6	hitachi-manage-info.start-timeout-in-seconds	Specifies in seconds the time during which to monitor whether the web server is started. If 0 is specified, the monitoring will not be performed.	0 to 3600	Cannot be omitted	60	asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.hitachi-manage-info.start-timeout-in-seconds=180
7	hitachi-manage-info.start-type	Specifies how to start the web server.	direct	Cannot be omitted	direct	asadmin set hitachi-

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		• `direct`: Direct startup				`webserver-configs.hitachi-webserver-config.WEB01-config.hitachi-manage-info.start-type=direct`
8	`hitachi-manage-info.starting-watch-interval-in-seconds`	Specifies in seconds the interval for monitoring whether the web server is started.	`1` to `86400`	Cannot be omitted	`1`	`asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.hitachi-manage-info.starting-watch-interval-in-seconds=3`
9	`hitachi-manage-info.starting-watch-start-time-in-seconds`	Specifies the time (in seconds) from executing the startup command by the web server until starting to check the operation.	`0` to `86400`	Cannot be omitted	`0`	`asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.hitachi-manage-info.starting-watch-start-time-in-seconds=20`
10	`hitachi-manage-info.stop-command`	Specifies the relative path (without the extension) from the template directory of each server type to the stop command to be executed by the web server.	Type: String	Cannot be omitted	`bin/stop-webserver`	`asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.hitachi-manage-info.stop-command=stop-webserver`
11	`hitachi-manage-info.stop-timeout-in-seconds`	Specifies in seconds the time during which to monitor whether the web server is stopped. If `0` is specified, the monitoring will not be performed.	`0` to `1800`	Cannot be omitted	`60`	`asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.hitachi-manage-info.stop-timeout-in-seconds=180`
12	`hitachi-manage-info.stopping-watch-interval-in-seconds`	Specifies in seconds the interval for monitoring whether	`1` to `86400`	Cannot be omitted	`1`	`asadmin set hitachi-webserver-configs.hitachi-webserver-`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		the web server is stopped.				`config.WEB01-config.hitachi-manage-info.stopping-watch-interval-in-seconds=3`
13	`hitachi-manage-info.template-path`	Specifies the absolute path of the server template for the web server.	An existing directory path	Cannot be omitted	*domain root directory*`/`*domain name*`/server_templates/`*server type*	`asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.hitachi-manage-info.template-path=C:\temp\webserver`
14	`hitachi-manage-info.working-dir`	Specifies the absolute path to the current directory of each command to be executed by the web server.	Type: String	*node-directory*`/`*server-name*	<empty string>	`asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.hitachi-manage-info.working-dir=C:\temp\webserver\command`
15	`property.`*ex_property*	Specifies the name and the value of an extended property to be added to the web server. The property name must start with `ex_`. The added extended property can be used as a replacement character string to be written in server template files for the web server.	Type: String	Property deletion	No Value Specified	`asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.property.ex_timeout=600`
16	`property.user`	Specifies the user name for Web Server's server process operations. This property corresponds to the `User` directive of Web Server.	Type: String	Property deletion	No Value Specified	`asadmin set hitachi-webserver-configs.hitachi-webserver-config.Web01-config.property.user=user01`
17	`property.group`	Specifies the group name for Web Server's server process operations. This property corresponds	Type: String	Property deletion	No Value Specified	`asadmin set hitachi-webserver-configs.hitachi-webserver-config.Web01-`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		to the `Group` directive of Web Server.				`config.property.group=group01`
18	`property.env_`*`environment_variables_name`*	Specifies the environment variable to be set when the web server starts. If you want to specify more than one environment variable, you must specify this standard property more than once. If the specified environment variable value includes `${apserver.home}`, it will be replaced by *Application Server installation directory*. The environment variable value will not be inherited even if this property value contains a setting such as `%PATH%;C:\temp` or `${PATH}:/temp`, which indicates that the environment variable value should be inherited. If you wish to specify such a value, make sure that you specify it in one of the following: • System environment variable of the operating system on the node • asenv definition on the node • Startup shell script for the server template If you need to set characters that cannot be specified for this property to the server environment variable, you must set the environment variable as explained below. • To set the environment variable, create an extended property on the server and then make sure that	Type: String Single-byte alphanumeric characters and the following symbols can only be specified: percent signs (`%`), at-signs (`@`), plus symbols (+), hyphens (−), underscores (_), commas (`,`), and periods (`.`).	Property deletion	No Value Specified	`asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.property.env_TZ=JST-9`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example

		the value to be set to the environment variable is assigned to this extended property.
		An example of such an extended property is shown below.
		Extended property name: `ex_myenv`
		Extended property value: *value-to-be-set-to-environment-variable*
		• Add lines of code to the script that exists in the server template and starts the server. These lines of code must set the value of the extended property to the environment variable.
		The following is an example of lines of code to be added:
		`MY_ENV=${property.ex_myenv.value}`
		`export MY_ENV`				

19	`property.listen-add-count`	Specify the number of Listen directives to be added if more than one Listen directive needs to be specified. This property specifies the number of directives to be added except for the Listen directive setting specified in the `listen-port` and `listen-host` properties. Therefore, the total number of Listen directives is calculated by adding one to the value specified in this property. The default is `0`. This property indicates how many times the	`0` to `2147483647`	Property deletion	No Value Specified	`asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.property.listen-add-count=2`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		following properties must be added: • `listen-add_port`*n*# • `listen-add_host`*n*# #: *n*: The minimum value is 1. The maximum value is the value specified in this property.				
20	`property.listen-add-host`*n*	Specifies the IP address of a Listen directive to be added. *n* in the property name must be a value within the range from 1 to `listen-add-count`. A Listen directive is specified by a pair of this and `listen-add-port`*n*, in which *n* is the same number. This property can be omitted. If *n* is outside the range from 1 to `listen-add-count`, the port number is not applicable to the web server. This property corresponds to the Listen directive of Web Server.	*host-name*|*IPv4-dot-notation*|*IPv6-dot-notation (The IPv6 address must be enclosed by angle brackets [].)*	Property deletion	No Value Specified	`asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.property.listen-add-host1=192.168.1.102`
21	`property.listen-add-port`*n*	Specifies the port number of a Listen directive to be added. *n* in the property name must be a value within the range from 1 to `listen-add-count`. A Listen directive is specified by a pair of this and `listen-add-host`*n*, in which *n* is the same number. If *n* is outside the range from 1 to `listen-add-count`, the port number is not applicable to the web server.	`1` to `65535`	Property deletion	No Value Specified	`asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.property.listen-add-port1=82`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		This property corresponds to the Listen directive of Web Server.				
22	property.listen-host	Specifies the IP address that will receive requests. This property can be omitted. This property corresponds to the Listen directive of Web Server.	*host-name*|*IPv4-dot-notation*|*IPv6-dot-notation (The IPv6 address must be enclosed by angle brackets [].)*	Property deletion	No Value Specified	asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.property.listen-host=192.168.1.101
23	property.listen-port	Specifies the port number that will receive requests. This property corresponds to the Listen directive of Web Server.	1 to 65535	Property deletion	80	asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.property.listen-port=81
24	property.max-request-workers	Specifies the maximum possible number of clients that can be connected concurrently. This property corresponds to the MaxRequestWorkers directive of Web Server.	1 to 1024	Property deletion	150	asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.property.max-request-workers=300
25	property.server-name	Specifies the server name and the port number of Web Server. This property corresponds to the ServerName directive of Web Server.	Type: String	Property deletion	www.example.com	asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.property.server-name=www.host1.co.jp
26	property.start-servers	Specifies the number of server processes at web server startup. This property corresponds to the StartServers directive of Web Server.	1 to 1024	Property deletion	20	asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-config.property.start-servers=10
27	property.threads-per-child	Specifies the number of threads to be started as servers. This property corresponds to the	1 to 1024	Property deletion	50	asadmin set hitachi-webserver-configs.hitachi-webserver-config.WEB01-

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		ThreadsPerChild directive of Web Server.				`config.property.threads-per-child=100`

Legend:

#: If "Property deletion" is shown in the "Default Value" column, specifying an empty string for the parameter deletes the property itself.

Server instance relation

Parameters starting with `servers.server.`*`Java-EE-server-name`*

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
1	`hitachi-relations.`*`hitachi-relation-type-ref`*`.hitachi-relation.`*`relation_name`*`.order`	Specifies the sequence number of an inter-server relation.	`0` to `65535`	Cannot be omitted	No Value Specified	`asadmin set servers.server.JavaEE01.hitachi-relations.redirect.hitachi-relation.redirect1.order=10`
2	`hitachi-relations.prf-relation.hitachi-relation.`*`relation_name`*`.property.`*`ex_property`	Specifies the name and the value of an extended property to be added to the Java-EE-server PRF relation. The property name must start with `ex_`. The added extended property can be used as a replacement character string to be written in server template files for the Java EE server that is the source of the inter-server relation.	Type: String	Property deletion	No Value Specified	`asadmin set servers.server.JavaEE01.hitachi-relations.prf-relation.hitachi-relation.prfref1.property.ex_timeout=600`
3	`hitachi-session-config.server-id`	Specifies the server ID that identifies the Java EE server on which load balancing should be performed with Web Server. The server ID specified in this property will be set to the 33rd and subsequent characters in the session ID.	Type : String	No Default Value	No Value Specified	`asadmin set servers.server.JavaEE01.hitachi-session-config.server-id=server1`
4	`monitoring-service.module-monitoring-levels.hitachi-jvm-memory-extensions`	Specifies the monitoring level for Java VM memory extension operation information.	`LOW`|`HIGH`|`OFF`	`OFF`	`HIGH`	`asadmin set configs.config.JavaEE01-config.monitoring-service.module-monitoring-`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
						`levels.hitachi-jvm-memory-extensions=OFF`
5	`property.`*`ex_property`*	Specifies the name and the value of an extended property to be added to the Java EE server. The property name must start with `ex_`. The added extended property can be used as a replacement character string to be written in server template files for the Java EE server.	Type: String	Property deletion	No Value Specified	`asadmin set servers.server.JavaEE01.property.ex_timeout=600`
6	`property.balancer-member-root-id`	Specifies the cookie ID to be used to perform redirection from the web server to this server via the cluster.	Type: String	Property deletion	No Value Specified	`asadmin set servers.server.JavaEE01.property.balancer-member-root-id=route01`
7	`property.env_`*`environment_variables_name`*	Specifies the environment variable to be set when the Java EE server starts. If you want to specify more than one environment variable, you must specify this standard property more than once. If the specified environment variable value includes `${apserver.home}`, it will be replaced by *Application Server installation directory*. The environment variable value will not be inherited even if this property value contains a setting such as `%PATH%;C:\temp` or `${PATH}:/temp`, which indicates that the environment variable value should be inherited. If you wish to specify such a value, make sure that you specify it in one of the following: • System environment	Type: String	Property deletion	No Value Specified	`asadmin set servers.server.JavaEE01.property.env_TZ=JST-9`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		variable of the operating system on the node • asenv definition on the node				

Legend:

#: If "Property deletion" is shown in the "Default Value" column, specifying an empty string for the parameter deletes the property itself.

Parameters starting with `configs.config.`*`configuration-name-of-Java-EE-server`*

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
1	`-config.java-config.native-library-path-prefix`	Specifies a prefix for the native library path. Path separator is ":". After the change of this parameter is the need to restart the server.	Type: String	<empty string>	No Value Specified	`asadmin set configs.config.instance1-config.java-config.native-library-path-prefix=D:\usr\nativelib`
2	`admin-service.jmx-connector.system.port`	The network port for JMX. On UNIX, creating sockets that listen on ports 1 - 1024 requires superuser privileges.	`1` to `65535`	No Default Value	8686<for DAS> ${JMX_SYSTEM_CONNECTOR_PORT}<for server instace>	`asadmin set configs.config.server-config.admin-service.jmx-connector.system.port=8686`
3	`availability-service.availability-enabled`	Used to enable availability for a specific cluster, or for a specific Web, EJB, or JMS container using `asadmin set` subcommands.	`true|false`	`true`	No Value Specified	`asadmin set configs.config.JavaEE01-config.availability-service.availability-enabled=true`
4	`connector-service.shutdown-timeout-in-seconds`	According to the Connector specification, while Application Server shuts down, all resource adapters should be stopped. A resource adapter might hang during shutdown, since shutdown is typically a resource intensive operation. To avoid such a situation, you can set a timeout that aborts resource adapter shutdown if exceeded. The default timeout is 30 seconds per	`1` to `2147483647`	`30`	No Value Specified	`asadmin set configs.config.JavaEE01-config.connector-service.shutdown-timeout-in-seconds=60`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		resource adapter module. To configure this timeout: Use the following asadmin set command: asadmin set server.connector-service.shutdown-timeout-in-seconds="num-secs"				
5	ejb-container.max-cache-size	Specifies the maximum number of instances that can be cached.	0 to 2147483647	512	2000	asadmin set configs.config.JavaEE01-config.ejb-container.max-cache-size=1000
6	ejb-container.max-pool-size	Specifies a maximum size, a pool can grow to. A value of 0 implies an unbounded pool. Unbounded pools eventually shrink to the steady-pool-size, in steps defined by pool-resize-quantity.	0 to 2147483647	32	24	asadmin set configs.config.JavaEE01-config.ejb-container.max-pool-size=30
7	ejb-container.steady-pool-size	Specifies the number of bean instances normally maintained in pool.	0 to 2147483647	0	24	asadmin set configs.config.JavaEE01-config.ejb-container.steady-pool-size=20
8	group-management-service.group-discovery-timeout-in-millis	Indicates the amount of time (in milliseconds) an instance's GMS module will wait during instance startup for discovering other members of the group. The group-discovery-timeout-in-millis timeout value should be set to the default or higher. The default is 5000.	1000 to 120000	5000	No Value Specified	asadmin set configs.config.JavaEE01-config.group-management-service.group-discovery-timeout-in-millis=8000
9	hitachi-eheap.httpsession-enabled	Specifies whether to apply explicit memory management to HTTP sessions.	true|false	true	true	asadmin set configs.config.JavaEE01-config.hitachi-eheap.httpsession-enabled=false
10	hitachi-jaxrs-config.connect-timeout	Specifies, in milliseconds, the initial value of	Type : Integer	No Default Value	No Value Specified	asadmin set configs.config.JavaEE01-

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		jersey.config.client.connectTimeout of client configuration properties provided by Jersey.				config.hitachi-jaxrs-config.connect-timeout=0
11	hitachi-jaxrs-config.read-timeout	Specifies, in milliseconds, the initial value of jersey.config.client.readTimeout of client configuration properties provided by Jersey.	Type : Integer	No Default Value	No Value Specified	asadmin set configs.config.JavaEE01-config.hitachi-jaxrs-config.read-timeout=0
12	hitachi-jaxws-config.connect-timeout	Specifies, in milliseconds, the initial value of the JAX-WS RI property com.sun.xml.ws.connect.timeout of JAX-WS BindingProvider properties.	Type : Integer	No Default Value	No Value Specified	asadmin set configs.config.JavaEE01-config.hitachi-jaxws-config.connect-timeout=60000
13	hitachi-jaxws-config.request-timeout	Specifies, in milliseconds, the initial value of the JAX-WS RI property com.sun.xml.ws.request.timeout of JAX-WS BindingProvider properties.	Type : Integer	No Default Value	No Value Specified	asadmin set configs.config.JavaEE01-config.hitachi-jaxws-config.request-timeout=300000
14	hitachi-jaxws-config.security-auth-password	Specifies the initial value of javax.xml.ws.security.auth.password of JAX-WS BindingProvider properties.	Type: String	No Default Value	No Value Specified	asadmin set configs.config.JavaEE01-config.hitachi-jaxws-config.security-auth-password=pass
15	hitachi-jaxws-config.security-auth-username	Specifies the initial value of javax.xml.ws.security.auth.username of JAX-WS BindingProvider properties.	Type: String	No Default Value	No Value Specified	asadmin set configs.config.JavaEE01-config.hitachi-jaxws-config.security-auth-username=user
16	hitachi-jaxws-config.session-maintain	Specifies, in milliseconds, the initial value of javax.xml.ws.session.maintain of JAX-WS BindingProvider properties.	true|false	No Default Value	No Value Specified	asadmin set configs.config.JavaEE01-config.hitachi-jaxws-config.session-maintain=true

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
17	hitachi-jca.connection-auto-close-enabled	Specifies whether to enable or disable automatic closing of connections. If a connection is obtained through a thread generated with a servlet or JSP, the connection will not close automatically. If connections obtained through callback methods of servlets and JSP (for example, the init method) or EJB (for example, the ejbCreate or PostConstruct method) are not explicitly closed, the connections might be automatically closed.	true|false	No Default Value	true	asadmin set configs.config.JavaEE01-config.hitachi-jca.connection-auto-close-enabled=true
18	hitachi-manage-info.get-pid-command	Specifies the command to obtain the ID of the process to be executed on the Java EE server. To specify a command, specify the relative path from the template directory for a server type. Do not specify the file extension.	Type: String	Cannot be omitted	bin/getpid-javaee	asadmin set configs.config.JavaEE01-config.hitachi-manage-info.get-pid-command=getpid-javaee
19	hitachi-manage-info.running-watch-interval-in-seconds	Specifies in seconds the interval at which Java EE server operations are verified.	1 to 86400	Cannot be omitted	10	asadmin set configs.config.JavaEE01-config.hitachi-manage-info.running-watch-interval-in-seconds=20
20	hitachi-manage-info.start-timeout-in-seconds	Specifies in seconds the time during which to monitor whether the Java EE server is started. If 0 is specified, the monitoring will not be performed.	0 to 3600	Cannot be omitted	60	asadmin set configs.config.JavaEE01-config.hitachi-manage-info.start-timeout-in-seconds=180
21	hitachi-manage-info.starting-watch-interval-in-seconds	Specifies in seconds the interval for monitoring whether the Java EE server is started.	1 to 86400	Cannot be omitted	1	asadmin set configs.config.JavaEE01-config.hitachi-manage-info.starting-watch-interval-in-seconds=3

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
22	hitachi-manage-info.starting-watch-start-time-in-seconds	Specifies, in seconds, the time from when a command is executed to start the Java EE server until when verification of operations is started.	0 to 86400	Cannot be omitted	0	asadmin set configs.config.JavaEE01-config.hitachi-manage-info.starting-watch-start-time-in-seconds=60
23	hitachi-manage-info.stop-timeout-in-seconds	Specifies in seconds the time during which to monitor whether the Java EE server is stopped. If 0 is specified, the monitoring will not be performed.	0 to 1800	Cannot be omitted	60	asadmin set configs.config.JavaEE01-config.hitachi-manage-info.stop-timeout-in-seconds=180
24	hitachi-manage-info.stopping-watch-interval-in-seconds	Specifies in seconds the interval for monitoring whether the Java EE server is stopped.	1 to 86400	Cannot be omitted	1	asadmin set configs.config.JavaEE01-config.hitachi-manage-info.stopping-watch-interval-in-seconds=3
25	hitachi-manage-info.template-path	Specifies the absolute path of the server template for the Java EE server.	An existing directory path	Cannot be omitted	*domain-root-directory*/*domain-name*/server_templates/*server-type*	asadmin set configs.config.JavaEE01-config.hitachi-manage-info.template-path=C:\temp\javaee
26	hitachi-monitoring.hitachi-statistics-files.base-time	Specifies the base time to switch operation information files. To specify the base time, specify how many minutes have elapsed since 00:00:00. To set the base time to h:m, specify the minutes calculated by the following formula: h x 60 + m.	An integer from 0 to 1439 (from 0 to 23 hours and 59 minutes)	0	0	asadmin set configs.config.JavaEE01-config.hitachi-monitoring.hitachi-statistics-files.base-time=360
27	hitachi-monitoring.hitachi-statistics-files.hitachi-output-enabled.jvm-memory-extensions	Specifies whether to accumulate in files the Java VM memory extension operation information collected by the operation information collection function of Java EE Server.	true|false	true	true	asadmin set configs.config.JavaEE01-config.hitachi-monitoring.hitachi-statistics-files.hitachi-output-enabled.jvm-memory-extensions=true

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
28	hitachi-monitoring.hitachi-statistics-files.interval	Specifies in seconds the accumulation interval for operation information files.	An integer from 1 to 86400 (from one second to one day)	60	60	asadmin set configs.config.JavaEE01-config.hitachi-monitoring.hitachi-statistics-files.interval=300
29	hitachi-monitoring.hitachi-statistics-files.output-dir	Specify the absolute path of the output destination directory if you want to change the output destination of operation information files. If the same output destination directory is specified for more than one server instance, the multiple server instances might output operation information to the same file. Therefore, specify different output directories for different server instance.	Single-byte alphanumeric characters, hyphens (-), underscores (_), colons (:), backslash (\), and forward slashes (/)	<empty string>	<empty string>	asadmin set configs.config.JavaEE01-config.hitachi-monitoring.hitachi-statistics-files.output-dir=D:/monitoring_dir
30	hitachi-monitoring.hitachi-statistics-files.num	Specifies the number of operation information files.	An integer from 2 to 168	8	8	asadmin set configs.config.JavaEE01-config.hitachi-monitoring.hitachi-statistics-files.num=14
31	hitachi-monitoring.hitachi-statistics-files.period	Specifies in hours the time for switching operation information files.	An integer from 1 to 744 (from 1 hour to 31 days)	24(1 day)	24(1 day)	asadmin set configs.config.JavaEE01-config.hitachi-monitoring.hitachi-statistics-files.period=1
32	http-service.virtual-server.*id*.property.errorReportValve.value	Specifies a fully qualified class name of a custom valve that produces default error pages for applications on this virtual server. Specify org.apache.catalina.valves.ErrorReportValve to enable the default error page mechanism for this virtual server. Specify an empty string to disable the default error page	• org.apache.catalina.valves.ErrorReportValve • <empty string>	org.apache.catalina.valves.ErrorReportValve	<empty string>	• To enable the default error page mechanism for the virtual server: asadmin set configs.config.JavaEE01-config.http-service.virtual-server.server.property.errorReportValve.value=org.apache.catalina.valves

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		mechanism for this virtual server.				.ErrorReportValve • To disable the default error page mechanism for the virtual server: asadmin set configs.config.JavaEE01-config.http-service.virtual-server.server.property.errorReportValve.value=""
33	iiop-service.iiop-listener.*id*.port	This property indicates the IIOP port numbers. Port number for the listener. Legal values are 1 - 65535. On UNIX, creating sockets that listen on ports 1 - 1024 requires super user privileges.	1 to 65535	1072	3700 <for {*name*="server-config", *id*="orb-listener-1"}> ${IIOP_LISTENER_PORT}<for server instace>	asadmin set configs.config.JavaEE01-config.iiop-service.iiop-listener.orb-listener-1.port=9999
34	jms-service.jms-host.*JMS host name*.port	The network port for JMS Provider. On UNIX, creating sockets that listen on ports 1 - 1024 requires superuser privileges.	1 to 65535	No Default Value	${JMS_PROVIDER_PORT}	asadmin set configs.config.server-config.jms-service.jms-host.default_JMS_host.port=7677
35	jms-service.reconnect-interval-in-seconds	You can change the default values for this timeout. The number of seconds between reconnect attempts. This interval applies for attempts on each JMS host in the AddressList and for successive addresses in the list. If it is too short, this time interval does not give a JMS host time to recover. If it is too long, the reconnect might represent an unacceptable delay.	1 to 2147483647	5	No Value Specified	asadmin set configs.config.JavaEE01-config.jms-service.reconnect-interval-in-seconds=5
36	mdb-container.max-pool-size	Specifies the maximum number of connections that can be	0 to 2147483647	32	No Value Specified	asadmin set configs.config.JavaEE01-config.mdb-

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		created to satisfy client requests.				container.max-pool-size=20
37	monitoring-service.module-monitoring-levels.http-service	Enables and Disables monitoring for the HTTP service by setting the monitoring level. Enables monitoring for the HTTP service by setting the monitoring level to HIGH/LOW. Disables monitoring for the HTTP service by setting the monitoring level to OFF. • OFF: No monitoring, no impact on performance. • LOW: Simple statistics, such as create count, byte count, and so on. • HIGH: Simple statistics plus method statistics, such as method count, duration, and so on.	OFF|LOW|HIGH	OFF	HIGH	asadmin set configs.config.JavaEE1-config.monitoring-service.module-monitoring-levels.http-service=HIGH
38	monitoring-service.module-monitoring-levels.jdbc-connection-pool	Enables and Disables monitoring for the JDBC connection pool by setting the monitoring level. Enables monitoring for the JDBC connection pool by setting the monitoring level to HIGH/LOW. Disables monitoring for the JDBC connection pool by setting the monitoring level to OFF. • OFF: No monitoring, no impact on performance. • LOW: Simple statistics, such as create count, byte count, and so on. • HIGH: Simple statistics plus method statistics, such as method	OFF|LOW|HIGH	OFF	HIGH	asadmin set configs.config.JavaEE1-config.monitoring-service.module-monitoring-levels.jdbc-connection-pool=HIGH

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		count, duration, and so on.				
39	monitoring-service.module-monitoring-levels.web-container	Enables and Disables monitoring for the web container by setting the monitoring level. Enables monitoring for the web container by setting the monitoring level to HIGH/LOW. Disables monitoring for the web container by setting the monitoring level to OFF. • OFF: No monitoring, no impact on performance. • LOW: Simple statistics, such as create count, byte count, and so on. • HIGH: Simple statistics plus method statistics, such as method count, duration, and so on.	OFF|LOW|HIGH	OFF	HIGH	asadmin set configs.config.JavaEE1-config.monitoring-service.module-monitoring-levels.web-container=HIGH
40	network-config.network-listeners.network-listener.*name*.enabled	Determines whether the listener is active.	true|false	true	No Value Specified	asadmin set configs.config.server-config.network-config.network-listeners.network-listener.http-listener-1.enabled=false
41	network-config.network-listeners.network-listener.*name*.port	The network port for network listener. On UNIX, creating sockets that listen on ports 1 - 1024 requires superuser privileges.	1 to 65535	No Default Value	8080<for http-listener-1 of DAS> 8181<for http-listener-2 of DAS> ${HTTP_LISTENER_PORT}<for http-listener-1 of server instance> ${HTTP_SSL_LISTENER	asadmin set configs.config.server-config.network-config.network-listeners.network-listener.http-listener-1.port=8080

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
					_PORT}<for http-listener-2 of server instance>	
42	network-config.protocols.protocol.admin-listener.http.request-timeout-seconds	Specifies time in seconds after which the request times out. Request timeout occurs when request processing time (from the start of the request processing to completion) exceeds this parameter. If 0 is specified, a timeout does not occur. We do not recommend specifying 0 (timeout disabled) for the timeout value. If you specify 0 and the processing time for an admin-listener request becomes long, the command processing might occupy the threads for processing the asadmin subcommand and user requests.	0 to 2147483647	900	900	asadmin set configs.config.server-config.network-config.protocols.protocol.admin-listener.http.request-timeout-seconds=600
43	network-config.protocols.protocol.http-listener-1.http.max-connections	Specifies the maximum number of HTTP requests that can be pipelined until the connection is closed by the server. Set this property to 1 to disable HTTP/1.0 keep-alive, as well as HTTP/1.1 keep-alive and pipelining.	1 to 2147483647	256	No Value Specified	asadmin set configs.config.server-config.network-config.protocols.protocol.http-listener-1.http.max-connections=120
44	network-config.protocols.protocol.http-listener-1.http.request-timeout-seconds	Specifies time in seconds after which the request times out. Request timeout occurs when request processing time (from the start of the request processing to completion) exceeds this parameter. If 0 is specified, a timeout does not occur.	0 to 2147483647	900	190	asadmin set configs.config.JavaEE01-config.network-config.protocols.protocol.http-listener-1.http.request-timeout-seconds=600
45	network-config.protocols.protocol.http	Specifies max time a connection can be deemed as idle and	-1 to 2147483647	30	-1	asadmin set configs.config.JavaEE01-

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
	-listener-1.http.timeout-seconds	kept in the keep-alive state. A value of less than 0 means keep alive connections are kept open indefinitely.				config.network-config.protocols.protocol.http-listener-1.http.timeout-seconds=20
46	network-config.protocols.protocol.sec-admin-listener.http.request-timeout-seconds	Specifies time in seconds after which the request times out. Request timeout occurs when request processing time (from the start of the request processing to completion) exceeds this parameter. If 0 is specified, a timeout does not occur. We do not recommend specifying 0 (timeout disabled) for the timeout value. If you specify 0 and the processing time for an admin-listener request becomes long, the command processing might occupy the threads for processing the asadmin subcommand and user requests.	0 to 2147483647	900	900	asadmin set configs.config.instance1-config.network-config.protocols.protocol.sec-admin-listener.http.request-timeout-seconds=600
47	property.*ex_property*	Specifies the name and the value of an extended property to be added to the Java EE server. The property name must start with ex_. The added extended property can be used as a replacement character string to be written in server template files for the Java EE server.	Type: String	Property deletion	No Value Specified	asadmin set configs.config.JavaEE01-config.property.ex_timeout=600
48	property.balancer-member-root-id	Specifies the cookie ID to be used to perform redirection from the web server to this server via the cluster.	Type: String	Property deletion	No Value Specified	asadmin set configs.config.JavaEE01-config.property.balancer-member-root-id=route01
49	property.env_*environment_variables_name*	Specifies the environment variable to be set when the Java	Type: String	Property deletion	No Value Specified	asadmin set configs.config.JavaEE01-

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		EE server starts. If you want to specify more than one environment variable, you must specify this standard property more than once. If the specified environment variable value includes `${apserver.home}`, it will be replaced by *Application Server installation directory*. The environment variable value will not be inherited even if this property value contains a setting such as `%PATH%;C:\temp` or `${PATH}:/temp`, which indicates that the environment variable value should be inherited. If you wish to specify such a value, make sure that you specify it in one of the following: • System environment variable of the operating system on the node • asenv definition on the node				`config.property.env_TZ=JST-9`
50	`security-service.activate-default-principal-to-role-mapping`	Applies a default principal for role mapping to any application that does not have an application-specific mapping defined. Every role is mapped to an instance of a java.security.Principal implementation class defined by `mapped-principal-class`. This class has the same name as the role.	`true|false`	`false`	No Value Specified	`asadmin set configs.config.JavaEE01-config.security-service.activate-default-principal-to-role-mapping=true`
51	`security-service.default-principal`	Used as the identity of the default security context when necessary and when no principal is provided. This attribute need not	Type: String	No Default Value	No Value Specified	`asadmin set configs.config.JavaEE01-config.security-service.default`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		be set for normal server operation.				`-principal=dsmith`
52	`security-service.default-principal-password`	The password of the default principal. This attribute need not be set for normal server operation.	Type: String	No Default Value	No Value Specified	`asadmin set configs.config.JavaEE01-config.security-service.default-principal-password=secret`
53	`security-service.mapped-principal-class`	Customizes the java.security.Principal implementation class used when `activate-default-principal-to-role-mapping` is set to `true`.	Type: String	No Default Value	No Value Specified	`asadmin set configs.config.JavaEE01-config.security-service.mapped-principal-class=CustomPrincipalImplClass`
54	`thread-pools.thread-pool.`*name*`.idle-thread-timeout-seconds`	Specifies the amount of time in seconds after which idle threads are removed from the pool.	1 to 2147483647	900	No Value Specified	`asadmin set configs.config.instance1-config.thread-pools.thread-pool.thread-pool-1.idle-thread-timeout-seconds=1000`
55	`thread-pools.thread-pool.`*name*`.max-queue-size`	The maxim number of tasks, which could be queued on the thread pool. `-1` disables any maximum checks.	• `-1` • 1 to 2147483647	4096	80 <for `http-thread-pool`>	`asadmin set configs.config.JavaEE01-config.thread-pools.thread-pool.thread-pool-1.max-queue-size=1000`
56	`thread-pools.thread-pool.`*name*`.max-thread-pool-size`	Specifies the maximum number of threads the pool can contain. To optimize this thread pool for use with an `iiop-listener`, set this value to `200`.	1 to 2147483647	5	200 <for `thread-pool-1`> 24 <for `http-thread-pool`>	`asadmin set configs.config.JavaEE01-config.thread-pools.thread-pool.http-thread-pool.max-thread-pool-size=8`
57	`thread-pools.thread-pool.`*name*`.min-thread-pool-size`	Minimum number of threads in the thread pool servicing requests in this queue. These are created up front when this thread pool is instantiated	1 to 2147483647	2	24 <for `http-thread-pool`>	`asadmin set configs.config.JavaEE01-config.thread-pools.thread-pool.thread-pool-1.min-thread-pool-size=20`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
58	transaction-service.automatic-recovery	Pending transactions are completed upon server startup if automatic-recovery is set to true.	true|false	false	true	asadmin set configs.config.JavaEE01-config.transaction-service.automatic-recovery=false
59	transaction-service.heuristic-decision	If the outcome of a distributed transaction cannot be determined because other participants are unreachable, this property determines the outcome.	commit|rollback	rollback	No Value Specified	asadmin set configs.config.JavaEE01-config.transaction-service.heuristic-decision=rollback
60	transaction-service.keypoint-interval	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 results in a larger transaction log file, but fewer keypoint operations and potentially better performance. A smaller value results in smaller log files, but slightly reduced performance due to the greater frequency of keypoint operations.	0 to 2147483647	65536	No Value Specified	asadmin set configs.config.JavaEE01-config.transaction-service.keypoint-interval=2048
61	transaction-service.property.db-logging-resource	Used to configuring the transaction service. The db-logging-resource property does not have a default value. It is unset by default.	Type: String	No Default Value	No Value Specified	asadmin set configs.config.JavaEE01-config.transaction-service.property.db-logging-resource="jdbc/TxnDS"
62	transaction-service.property.oracle-xa-recovery-workaround	The XAResource.commit method also has some issues. To disable Java EE Server workaround, set the oracle-xa-recovery-workaround property value to false.	true|false	true	No Value Specified	asadmin set configs.config.JavaEE01-config.transaction-service.property.oracle-xa-recovery-workaround=true

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
63	transaction-service.property.pending-txn-cleanup-interval	Used to configuring the transaction service. The pending-txn-cleanup-interval default of -1 means the periodic recovery thread doesn't run. The units are seconds. Periodic automatic recovery is performed by a background thread if the pending-txn-cleanup-interval property is set to a positive value.	• -1 • 1 to 2147483647	No Default Value <60, if this property is present but value is empty>	No Value Specified	asadmin set configs.config.JavaEE01-config.transaction-service.property.pending-txn-cleanup-interval=-1
64	transaction-service.property.use-last-agent-optimization	Used to configuring the transaction service. The use-last-agent-optimization property is set to true by default. 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.	true|false	true	No Value Specified	asadmin set configs.config.JavaEE01-config.transaction-service.property.use-last-agent-optimization=true
65	transaction-service.property.wait-time-before-recovery-insec	Used to configuring the transaction service.	1 to 2147483647	60	No Value Specified	asadmin set configs.config.JavaEE01-config.transaction-service.property.wait-time-before-recovery-insec=60
66	transaction-service.property.xaresource-txn-timeout	Used to configuring the transaction service. The xaresource-txn-timeout default of 0 means there is no timeout. The units are seconds. Changes the XAResource timeout. In some cases, the XAResource default timeout can cause transactions to be aborted, so it is desirable to change it.	0 to 2147483647	120 <depends on XAResource >	180	asadmin set configs.config.JavaEE01-config.transaction-service.property.xaresource-txn-timeout=0

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
67	transaction-service.retry-timeout-in-seconds	Determines the retry time in the following scenarios: • 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. -1 specifies infinite retries. A value of 0 (zero) specifies no retries. A positive value indicates the time after which a retry is attempted.	-1 to 2147483647	600	No Value Specified	asadmin set configs.config.JavaEE01-config.transaction-service.retry-timeout-in-seconds=600
68	transaction-service.timeout-in-seconds	Specifies the amount of time after which the transaction is aborted. If set to 0, the transaction never times out.	0 to 2147483647	0	180	asadmin set configs.config.JavaEE01-config.transaction-service.timeout-in-seconds=0
69	web-container.session-config.session-manager.manager-properties.max-sessions	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 IllegalStateException to be thrown.	• -1 • 1 to 2147483647	-1	No Value Specified	asadmin set configs.config.instance1-config.web-container.session-config.session-manager.manager-properties.max-sessions=10
70	web-container.session-config.session-properties.timeout-in-seconds	Specifies the value in seconds after which session should be timed out.	0 to 2147483647	1800	7200	asadmin set configs.config.JavaEE01-config.web-container.session-config.session-properties.timeout-in-seconds=5000

Legend:

#: If "Property deletion" is shown in the "Default Value" column, specifying an empty string for the parameter deletes the property itself.

Administration relation

Administration relation

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
1	clusters.cluster.*cluster_name*.config-ref	References the name of the "config" used by the server instance. If cluster is created with name "mycluster" then name of the config-ref is mycluster-config.	Type: String	No Default Value	No Value Specified	asadmin set clusters.cluster.mycluster.config-ref=mycluster-config
2	clusters.cluster.*cluster_name*.gms-bind-interface-address	The Internet Protocol (IP) address of the network interface to which GMS binds. If cluster is created with name "mycluster" then name of the config-ref is mycluster-config.	Type: String	No Default Value	No Value Specified	asadmin set clusters.cluster.mycluster.gms-bind-interface-address=%GMS-BIND-INTERFACE-ADDRESS-mycluster%
3	hitachi-domain.hitachi-snapshot.max-num	Specifies the number of files that can be generated in the output destination directory of the system information archive file. To specify the same directory as the archive file output destination for the domain or multiple nodes, specify the same numeric value to this parameter. If the number of the files exceeds the specified value, files are deleted starting from the oldest. The value specified to this parameter is applied to the entire domain. To apply the specified value after you have specified it, restart all web servers, Java EE servers, and performance tracers in the domain.	5 to 2147483647	10	10	asadmin set hitachi-domain.hitachi-snapshot.max-num=20
4	hitachi-domain.hitachi-snapshot.output-dir	Specifies the absolute path to the output destination directory of the system information archive file. For the output destination directory, only a directory already	Single-byte alphanumeric characters, hyphens (-), underscores (_), colons (:), backslash (\), and	*Application Server installation directory*/javaee/snapshot	*Application Server installation directory*/javaee/snapshot	asadmin set hitachi-domain.hitachi-snapshot.output-dir=/logs/snapshot

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		created on the local drive can be specified. The value specified to this parameter is applied to the entire domain. To apply the specified value after you have specified it, restart all web servers, Java EE servers, and performance tracers in the domain.	forward slashes (/)			
5	`hitachi-manage-configs.hitachi-hook-policy.hitachi-event-hooks.hitachi-event-hook.`*`event-name`*`.enabled`	Specifies whether to execute this event hook command.	`true`|`false`	Cannot be omitted	`true`	`asadmin set hitachi-manage-configs.hitachi-hook-policy.hitachi-event-hooks.hitachi-event-hook.Event1.enabled=false`
6	`hitachi-manage-configs.hitachi-hook-policy.hitachi-event-hooks.hitachi-event-hook.`*`event-name`*`.monitoring-max-count`	Specifies the maximum number of event hook command executions permitted for the same type of events within the time specified by `hitachi-hook-policy.hitachi-event-hooks.hitachi-event-hook.`*`event-name`*`.monitoring-time-span-in-seconds`. If `0` is specified, execution of the event hook command is always permitted. If 1 or higher is specified, and if the number of events that occurred exceeds the specified number within the time specified by `hitachi-hook-policy.hitachi-event-hooks.hitachi-event-hook.`*`event-name`*`.monitoring-time-span-in-seconds`, the event hook command is not	`0` to `256`	Cannot be omitted	`0`	`asadmin set hitachi-manage-configs.hitachi-hook-policy.hitachi-event-hooks.hitachi-event-hook.Event1.monitoring-max-count=5`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		executed for the excessive events.				
7	hitachi-manage-configs.hitachi-hook-policy.hitachi-event-hooks.hitachi-event-hook.*event-name*.monitoring-time-span-in-seconds	Specifies the time interval (in seconds) for setting the maximum number of permitted executions#. If 0 is specified, the monitoring will not be performed. If 1 or higher is specified, monitoring is performed during the specified time. #: To specify the maximum number of permitted executions, use hitachi-hook-policy.hitachi-event-hooks.hitachi-event-hook.*event-name*.monitoring-max-count.	0 to 2147483647	Cannot be omitted	0	asadmin set hitachi-manage-configs.hitachi-hook-policy.hitachi-event-hooks.hitachi-event-hook.Event1.monitoring-time-span-in-seconds=60
8	hitachi-manage-configs.hitachi-hook-policy.hitachi-event-hooks.hitachi-event-hook.*event-name*.path	Specifies the absolute path to the event hook command. For separating files, use a slash (/). If you used a shell script to create the event hook command, add /bin/sh before the command name. If /bin/sh is not specified, the command might not function properly. Specification example /bin/sh /home/user1/javaeeabnormalend.sh	Type: String	Cannot be omitted	*Application Server installation directory*/glassfish/config/manager/snapshot_event-hook To run this script properly, you need to specify the following variables in the script file: • ADMIN_HOST • UID • PWDFILE	asadmin set hitachi-manage-configs.hitachi-hook-policy.hitachi-event-hooks.hitachi-event-hook.Event1.path=/temp/event-hook.sh
9	hitachi-relation-types.hitachi-relation-type.prf-	Specifies the name and the value of an extended property to be added to the PRF relation. The property	Type: String	Property deletion	No Value Specified	asadmin set hitachi-relation-types.hitachi-relation-

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
	relation.property.*ex_property*	name must start with ex_. The added extended property can be used as a replacement character string to be written in server template files for the server that is the source of the inter-server relation.				type.prf-relation.property.ex_timeout=600
10	hitachi-relation-types.hitachi-relation-type.redirect.proroperty.*ex_property*	Specifies the name and the value of an extended property to be added to a redirection relation. The property name must start with ex_. The added extended property can be used as a replacement character string to be written in server template files for the web server that is the source of the inter-server relation.	Type: String	Property deletion	No Value Specified	asadmin set hitachi-relation-types.hitachi-relation-type.redirect.proroperty.ex_timeout=600
11	hitachi-relation-types.hitachi-relation-type.redirect.proroperty.negative	Specifies, in the redirector behavior definition, whether to exclude requests to the mapping definition. This property corresponds to the ! option of the ProxyPass directive of Web Server.	true|false	Property deletion	false	asadmin set hitachi-relation-types.hitachi-relation-type.redirect.proroperty.negative=true
12	hitachi-relation-types.hitachi-relation-type.redirect.proroperty.network-listener	Specifies the name of the network listener that is to be the redirection destination. This property corresponds to the URL of the ProxyPass directive of Web Server. The host name and the port number identified from the specified network listener name will be set to the URL.	A character string representing the network listener name	Property deletion	http-listener-1	asadmin set hitachi-relation-types.hitachi-relation-type.redirect.proroperty.network-listener=http-listener-1
13	hitachi-relation-types.hitachi-relation-type.redirect.proroperty.path	Specifies, in the redirector behavior definition, the name of the local virtual path of the mapping definition.	Format of path (the name of the local virtual path) defined in	Property deletion	/	asadmin set hitachi-relation-types.hitachi-relation-type.redirect.p

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		This property corresponds to the pathname specification of the ProxyPass directive of Web Server.	ProxyPass of the httpsd.conf file			roperty.path=/news/
14	hitachi-relation-types.hitachi-relation-type.redirect.property.proxy-pass-option	Specifies, in the redirector behavior definition, the adjustment parameter for the connection pooling in the mapping definition. This property corresponds to the "*key=value*" specification of the ProxyPass directive of Web Server.	Format of the adjustment parameter for the connection pooling defined in ProxyPass of the httpsd.conf file ([*key=value*key=value* ...])	Property deletion	<empty string>	asadmin set hitachi-relation-types.hitachi-relation-type.redirect.property.proxy-pass-option=timeout=300
15	hitachi-relation-types.hitachi-relation-type.redirect.property.set-proxy-pass-reverse-cookie	Specifies, in the redirector behavior definition, whether to re-assign the Set-Cookie header. This property corresponds to the HWSProxyPassReverseCookie directive of Web Server.	true|false	Property deletion	true	asadmin set hitachi-relation-types.hitachi-relation-type.redirect.property.set-proxy-pass-reverse-cookie=true
16	hitachi-relation-types.hitachi-relation-type.redirect.property.stickysession	Specifies, in the redirector behavior definition, whether to enable Sticky. This property corresponds to the stickysession key of the ProxyPass directive of Web Server. If this property is true, the required Header directive is added.	true|false	Property deletion	true	asadmin set hitachi-relation-types.hitachi-relation-type.redirect.property.stickysession=true
17	nodes.node.*node_name*.hitachi-node.hitachi-node-snapshot.max-num	Specifies the number of files that can be generated in the output destination directory of the system information archive file for each node. You can change the setting for outputting the system information of the domain administration server by specifying	5 to 2147483647	Value specified for the hitachi-domain.hitachi-snapshot.max-num parameter	<empty string>	asadmin set nodes.node.node1.hitachi-node.hitachi-node-snapshot.max-num=20

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		the parameter of the default node (`localhost-`*`domain-name`*). To specify the same directory as the archive file output destination for the domain or multiple nodes, specify the same numeric value to this parameter. If the number of the files exceeds the specified value, files are deleted from the oldest. If the specified value is a null character, the value specified to the `hitachi-domain.hitachi-snapshot.max-num` parameter is applied. If you specify a value other than the null character, that value has priority over the value of the `hitachi-domain.hitachi-snapshot.max-num` parameter. This parameter is applied only to the node to which you specified the value.				
18	`nodes.node.`*`node_name`*`.hitachi-node.hitachi-node-snapshot.output-dir`	Specifies the absolute path to the output destination directory of the system information archive file for each node. You can change the setting for outputting the system information of the domain administration server by specifying the parameter of the default node (`localhost-`*`domain-name`*). For the output destination directory, only a directory already created on the local drive can be specified.	Single-byte alphanumeric characters, hyphens (-), underscores (_), colons (:), backslash (\), and forward slashes (/)	Value specified for the `hitachi-domain.hitachi-snapshot.output-dir` parameter	<empty string>	`asadmin set nodes.node.node1.hitachi-node.hitachi-node-snapshot.output-dir=/logs/node1/snapshot`

No.	Parameter name	Description	Range Value	Default Value#	Initial Value	Example
		If the specified value is a null character, the value specified to the `hitachi-domain.hitachi-snapshot.output-dir` parameter is applied. If you specify a value other than the null character, that value has priority over the value of the `hitachi-domain.hitachi-snapshot.output-dir` parameter. This parameter is applied only to the node to which you specified the value.				
19	`nodes.node.`*`node_name`*`.hitachi-node.property.ex_property`	Specifies the name and the value of an extended property to be added to the node. The property name must start with `ex_`. The added extended property can be used as a replacement character string to be written in server template files for the server that exists in the node.	Type: String	Property deletion	No Value Specified	`asadmin set nodes.node.node1.hitachi-node.property.ex_timeout=600`
20	`nodes.node.`*`node_name`*`.node-host`	Specifies the name of the host indicated by the node.	Type: String	No Default Value	`localhost`	`asadmin set nodes.nodenode1.node-host=HostA`

Legend:

#: If "Property deletion" is shown in the "Default Value" column, specifying an empty string for the parameter deletes the property itself.

2.12 Commands used for system administration

This section describes the syntax and functionality of the commands used for system administration.

2.12.1 add-library

Adds one or more library JAR files to Java EE Server.

Synopsis

```
asadmin [asadmin-options] add-library [--help]
    [--type={common|ext|app}] [--upload={false|true}]
    library-file-path [library-file-path ... ]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `add-library` subcommand adds one or more library archive files to Java EE Server.

This subcommand is supported in remote mode only.

The DAS or server instance has to be restarted so that the libraries are picked up by the server runtime.

Precondition

DAS has to be in a running state.

Files

The library .jar files that need to be added to the DAS or server instance are entered as inputs for the `add-library` subcommand execution.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--type={common|ext|app}`

Specifies the library type and Application Server directory to which the library is added.

Type: String

The following values can be specified:

- `common`
  Adds the library files to the Common class loader directory: `domain-dir/lib`. This is the default value.
- `ext`
  Adds the library files to the Java optional package directory: `domain-dir/lib/ext`.
- `app`

Adds the library files to the application-specific class loader directory: `domain-dir/lib/applibs`.

Default value: `common`

`--upload={false|true}`

Specifies whether the subcommand uploads the file to the DAS. In most situations, this option can be omitted. The valid values are as follows: `true` and `false`.

The default value depends on whether the DAS is on the host where the subcommand is run or is on a remote host.

If the DAS is on the host where the subcommand is run, then the default value is `false`.

If the DAS is on a remote host, then the default value is `true`.

In this situation, the subcommand fails if the `--upload` option is `false`.

If a directory file path is specified, this option is ignored.

Type: Boolean

The following values can be specified:

- `true`

  The subcommand uploads the file to the DAS over a network connection.

- `false`

  The subcommand does not upload the file and attempts to access the file through a specified file name. If the DAS cannot access the file, the subcommand fails.

  For example, the DAS may be running as a different user than the administration user and does not have read access to the file.

Default value:

If the DAS is on the host where the subcommand is run, then the default value is `false`.

If the DAS is on a remote host, then the default value is `true`.

`library-file-path`

Specifies the path to archive files that contain the libraries that have to be added. The relative path is the *`Application Server installation directory`*`/javaee/glassfish/bin` directory.

Type: String

The following values can be specified:

- *Multiple paths separated by a space*
- *Absolute or relative path*

Default value: N/A

Examples

The following example adds the library available in the archive file `mylib.jar`, to the application-specific class loader directory on the default server instance.

```
asadmin add-library --type app /tmp/mylib.jar
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.12.2 create-profiler

Creates the profiler element.

Synopsis

```
asadmin [asadmin-options] create-profiler [--help]
    [--target target] [--classpath classpath]
    [--nativelibrarypath native_library_path] [--enabled={false|true}]
    profiler_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-profiler` subcommand creates the profiler element. A server instance is tied to the profiler by the profiler element in the Java configuration. Only one profiler exists at a time. If you attempt to create a profiler while one already exists, an error message is displayed. For changes to take effect, the server must restarted. This command is supported in remote mode only.

Only one profiler exists at a time. If you attempt to create a profiler while one already exists, an error message is displayed.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target_name*

Specifies the target on which you are creating a profiler.

Type: String

The following values can be specified:

- `server`
  Creates the profiler for the default server instance. This is the default value.
- *configuration_name*
  Creates the profiler for the named configuration.
- *cluster_name*
  Creates the profiler for every server instance in the cluster.
- *instance_name*
  Creates the profiler for a particular server instance.

Default value: `server`

`--classpath` *classpath*

Specifies the `Java classpath` string classes needed by the profiler.

Type: String

The following values can be specified:

- *Specify the classpath in string needed by profiler*

Default value: N/A

`--nativelibrarypath` *native_library_path*

This path is automatically constructed to be a concatenation of Java EE Server installation relative path for its native shared libraries, standard JRE native library path, the shell environment setting (LD_LIBRARY_PATH on UNIX) and any path that may be specified in the profile element.

Type: String

The following values can be specified:

- *Specify the nativelib path*

Default value: N/A

`--enabled={false|true}`

Enables the profiler by default.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

profiler_name

Specifies the name of the profiler.

Type: String

The following values can be specified:

- *Specify the profiler name*

Default value: N/A

Examples

The following example creates a profiler on the server instance `instance1`.

```
asadmin create-profiler --target instance1 --classpath /home/appserver/
--nativelibrarypath /u/home/lib --enabled=false sample_profiler
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.12.3 delete-profiler

Removes the profiler element.

Synopsis

```
asadmin [asadmin-options] delete-profiler [--help]
    [--target target] profiler_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-profiler` subcommand deletes the profiler element in the Java configuration. Only one profiler can exist at a time. If no profiler exists and when you attempt to delete a profiler, an error message appears.

This command is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target profiler element which you are deleting.

Type: String

The following values can be specified:

- `server`
  Deletes the profiler element for the default server instance `server` and is the default value.
- *configuration_name*
  Deletes the profiler element for the named configuration.
- *cluster_name*
  Deletes the profiler element for every server instance in the cluster.
- *instance_name*
  Deletes the profiler element for a particular server instance.

Default value: `server`

profiler_name

Specifies the name of the profiler.

Type: String

The following values can be specified:

- *Name of the profiler*

Default value: N/A

Examples

The following example deletes the profiler which exists on server instance `instance1`.

```
asadmin delete-profiler --target instance1
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.12.4 list-commands

Lists the available commands.

Synopsis

```
asadmin [asadmin-options] list-commands [--help]
    [--localonly={false|true}] [--remoteonly={false|true}]
    [pattern-list]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-commands` subcommand lists the `asadmin` subcommands. By default, the `list-commands` subcommand displays a list of local subcommands followed by a list of remote subcommands. You can specify the following to be listed:

- only remote subcommands
- only local subcommands
- only subcommands whose names contain a specified text string

This subcommand is supported in local mode and remote mode.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--localonly={false|true}`

Lists only the local commands if this option is set to `true`.

If this option is set to `true`, the `--remoteonly` option must be set to `false`. Otherwise, an error occurs.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--remoteonly={false|true}`

Lists only the remote commands if this option is set to `true`.

If this option is set to `true`, the `--localonly` option must be set to `false`. Otherwise, an error occurs.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`pattern-list`

Specifies a space-separated list of text strings on which to filter the list of subcommands. Only the subcommands that contain any one of the specified text strings is listed.

Type: String

The following values can be specified:

- *Name of the subcommand*

Default value: N/A

Examples

The following example lists only the local subcommands.

```
asadmin list-commands --localonly=true
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.12.5 list-libraries

Lists the library archive files present on Java EE Server.

Synopsis

```
asadmin [asadmin-options] list-libraries [--help]
    [--type={common|ext|app}]
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `list-libraries` subcommand of asadmin lists library archive files present on Java EE Server. This subcommand is supported in remote mode only.

The `--type` option specifies the library type and Java EE Server directory for which the libraries are to be listed.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--type={common|ext|app}`

Specifies the library type and Java EE Server directory for which libraries are listed.

Type: String

The following values can be specified:

- `common`
  Lists the library files for the Common class loader directory, `domain-dir/lib`. This is the default value.
- `ext`
  Lists the library files for the Java optional package directory, `domain-dir/lib/ext`.
- `app`
  Lists the library files for the application-specific class loader directory, `domain-dir/lib/applibs`.

Default value: `common`

Examples

The following example lists the libraries in the application-specific class loader directory on the default server instance.

```
asadmin list-libraries --type app
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.12.6 login

Logs you into a domain.

Synopsis

```
asadmin [asadmin-options] login [--help]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `login` subcommand of `asadmin` enables you to log into a particular domain, thus simplifying domain administration. If Java EE Server domains are created on various machines (locally), you can run the `asadmin` utility from any of these machines and manage domains located elsewhere (remotely). This is especially useful when a particular machine is chosen as an administration client that manages multiple domains and servers.

Files

`.asadminpass` file in user's home directory.

Arguments

`--help | -?`

Displays the help text for the subcommand.

Examples

The following example logs into a domain located on another machine. Options are specified before the `login` subcommand.

```
asadmin --host foo --port 8282 login
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.12.7 remove-library

Removes one or more library JAR files from Java EE Server.

Synopsis

```
asadmin [asadmin-options] remove-library [--help]
    [--type={common|ext|app}]
    library-name [library-name ...]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `remove-library` subcommand of `asadmin` removes one or more library JAR files from Java EE Server. You can specify whether the libraries are removed from the `Common class loader` directory, the `Java optional package` directory, or the `application-specific class loader` directory. This subcommand is supported in remote mode only.

The library archive file is removed from the DAS. For common and extension libraries, you must restart the DAS so the library removals are picked up by the server runtime. To remove the libraries from other server instances, synchronize the instances with the DAS by restarting them.

Precondition

- Domain Administration Server (DAS) is running.
- A library must already be added to Java EE Server.

Files

- For type `-common`,

  Removes the library files from the `Common class loader` directory, `domain-dir/lib`.
- For type `-ext`,

  Removes the library files from the `Java optional package` directory, `domain-dir/lib/ext`.
- For type `-app`,

  Removes the library files from the `application-specific class loader` directory, `domain-dir/lib/applibs`. From the above directories `Space separated list of library JAR file name` which will be deleted from the DAS.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--type={common|ext|app}`

Specifies the library type and Java EE Server directory from which the library is to be removed.

Type: String

The following values can be specified:

- `common`

  Removes the library files from the `Common class loader` directory: `domain-dir/lib`. This is the default value.
- `ext`

  Removes the library files from the `Java optional package` directory, `domain-dir/lib/ext`.
- `app`

  Removes the library files from the `application-specific class loader` directory, `domain-dir/lib/applibs`.

Default value: `common`

`library_name`

Specifies the names of the JAR files that contain the libraries that are to be removed.

Type: String

The following values can be specified:

- *Name of the archive*

Default value: N/A

Examples

The following example removes the library in the archive file `mylib.jar` from the `application-specific class loader` directory on the default server instance.

```
asadmin remove-library --type app mylib.jar
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.12.8 version

Displays the version information for Java EE Server.

Synopsis

```
asadmin [asadmin-options] version [--help]
    [--verbose={false|true}] [--local={false|true}]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `version` subcommand of `asadmin` displays the version information for Java EE Server. By default, if the subcommand cannot contact the DAS, the subcommand retrieves the version information locally and displays a warning message. This subcommand is supported in remote mode and local mode.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`{--verbose|-v}={false|true}`

Provides the version of the Java Runtime Environment (JRE) in which the server is running. If this option is set to `true`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--local={false|true}`

Specifies whether the version information is retrieved locally.

If this option is set to `true`, the subcommand obtains the version locally from the installation of Java EE Server on the host where the subcommand is run. If this option is set to `false` (default), the subcommand attempts to contact the DAS to obtain the version. If the attempt to contact the DAS fails, the subcommand retrieves the version locally and displays a warning message.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

Examples

The following example obtains the Version Information from a running DAS.

```
asadmin version
```

Exit Status

Exit Status	Explanation
`0`	subcommand executed successfully.
`1`	error in executing the subcommand.

2.13 Commands used for internet connection administration

This section describes the syntax and functionality of the commands used for internet connection administration.

2.13.1 create-protocol

Adds a new protocol.

Synopsis

```
asadmin [asadmin-options] create-protocol [--help]
    [--securityenabled={false|true}] [--target target] protocol-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-protocol` subcommand creates a protocol. This command is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--securityenabled={false|true}`

If set to `true`, the protocol runs SSL. Also it can turn SSL2 or SSL3 ON or OFF and set ciphers using an SSL element. The security setting globally enables or disables SSL by making certificates available to the server instance.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--target` *target*

Creates the protocol only on the specified target.

Type: String

The following values can be specified:

- `server`
  Creates the protocol on the default server instance. This is the default value.
- *configuration-name*
  Creates the protocol in the specified configuration.

- *cluster-name*
  Creates the protocol on all server instances in the specified cluster.
- *standalone-instance-name*
  Creates the protocol on the specified standalone server instance.

Default value: `server`

protocol-name

Specifies the name of the protocol.

Type: String

The following values can be specified:

- *Specify the protocol name*

Default value: N/A

Examples

The following example creates a protocol on the server instance `instance1`.

```
asadmin create-protocol --target instance1 --securityenabled=true http-1
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.13.2 create-ssl

Creates and configures the SSL element in the selected HTTP listener, IIOP listener, or IIOP service.

Synopsis

```
asadmin [asadmin-options] create-ssl [--help] [--target target]
    --type listener_or_service_type --certname cert_name
    [--ssl2enabled={false|true}] [--ssl2ciphers ssl2ciphers]
    [--ssl3enabled={true|false}] [--tlsenabled={true|false}]
    [--ssl3tlsciphers ssl3tlsciphers]
    [--clientauthenabled={false|true}] [listener_id]
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `create-ssl` subcommand creates and configures the SSL element in the selected HTTP listener, IIOP listener, or IIOP service to enable secure communication on that listener/service.

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Files

For `--type` option, when the type is `iiop-service`, the `ssl-client-config` along with the embedded ssl element is created in the `domain.xml` file.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target on which you are configuring the SSL element.

Type: String

The following values can be specified:

- `server`
  Specifies the server in which the `iiop-service` or HTTP/IIOP listener is to be configured for SSL.
- *configuration_name*
  Specifies the configuration that contains the HTTP/IIOP listener or `iiop-service` for which SSL is to be configured.
- *cluster_name*
  Specifies the cluster in which the HTTP/IIOP listener or `iiop-service` is to be configured for SSL. All the server instances in the cluster will get the SSL configuration for the respective listener or `iiop-service`.
- *instance_name*
  Specifies the instance in which the HTTP/IIOP listener or `iiop-service` is to be configured for SSL.

Default value: `server`

`--type` *listener_or_service_type*

Specifies the type of service or listener for which the SSL is created. If the `--type` is `iiop-service`, then the `listener_id` is not required. When the type is `iiop-service`, the `ssl-client-config` along with the embedded ssl element is created in `domain.xml`.

Type: String

The following values can be specified:

- `http-listener`
- `iiop-listener`
- `iiop-service`
- `jmx-connector`
- `network-listener`

Default value: N/A

`--certname` *cert_name*

Specifies the nickname of the server certificate in the certificate database or the PKCS#11 token. The format of the name in the certificate is `tokenname:nickname`. For this property, the `tokenname:` is optional.

Type: String

The following values can be specified:

- *Nickname of the server certificate in the certificate database*

Default value: N/A

`--ssl2enabled={false|true}`

Specifies the property required to enable the SSL2. If both SSL2 and SSL3 are enabled for a virtual server, the server tries SSL3 encryption first. In the event SSL3 encryption fails, the server then tries SSL2 encryption.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--ssl2ciphers` *`ss12ciphers`*

Specifies a comma-separated list of the SSL2 ciphers to be used. Ciphers that are not explicitly listed will be disabled for the target, even if those Ciphers are available in the particular Cipher suite being used currently. If this option is not used, all the supported Ciphers are assumed to be enabled.

Type: String

The following values can be specified:

- `rc4`
- `rc4export`
- `rc2`
- `rc2export`
- `idea`
- `des`
- `desede3`

Default value:

- `rc4`
- `rc4export`
- `rc2`
- `rc2export`
- `idea`
- `des`
- `desede3`

`--ssl3enabled={true|false}`

Specifies the property required to enable the SSL3. If both SSL2 and SSL3 are enabled for a virtual server, the server tries SSL3 encryption first. In the event SSL3 encryption fails, the server then tries SSL2 encryption.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--tlsenabled={true|false}`

Specifies the property required to disable the TLS. It is good practice to enable TLS, which is a more secure version of SSL.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--ssl3tlsciphers` *`ssl3tlsciphers`*

Specifies a comma-separated list of the SSL3 and/or TLS ciphers to be used. Ciphers that are not explicitly listed will be disabled for the target, even if those Ciphers are available in the particular Cipher suite being used currently. If this option is not used, all the supported Ciphers are assumed to be enabled.

If `iiop-listener` or `iiop-service` is specified for the `--type` option, specify a plus sign (+) before the encryption algorithm that is specified by the `--ssl3tlsciphers` option.

Type: String

The following values can be specified:

- `SSL_RSA_WITH_RC4_128_MD5`
- `SSL_RSA_WITH_3DES_EDE_CBC_SHA`
- `SSL_RSA_WITH_DES_CBC_SHA`
- `SSL_RSA_EXPORT_WITH_RC4_40_MD5`
- `SSL_RSA_WITH_NULL_MD5`
- `SSL_RSA_WITH_RC4_128_SHA`
- `SSL_RSA_WITH_NULL_SHA`

Default value:

- `SSL_RSA_WITH_RC4_128_MD5`
- `SSL_RSA_WITH_3DES_EDE_CBC_SHA`
- `SSL_RSA_WITH_DES_CBC_SHA`
- `SSL_RSA_EXPORT_WITH_RC4_40_MD5`
- `SSL_RSA_WITH_NULL_MD5`
- `SSL_RSA_WITH_RC4_128_SHA`
- `SSL_RSA_WITH_NULL_SHA`

`--clientauthenabled={false|true}`

Specifies the property to enable the SSL3 client authentication performed on every request independent of ACL-based access control.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

listener_id

Specifies the ID of the HTTP or IIOP listener for which the SSL element is to be created. The `listener_id` is not required if the `--type` is `iiop-service`.

Type: String

The following values can be specified:

- *ID of the HTTP or IIOP listener*

Default value: N/A

Examples

The following example shows how to create an SSL element for an HTTP listener named `http-listener-1` on the server instance `instance1`.

```
asadmin create-ssl --type http-listener --target instance1
--certname sampleCert http-listener-1
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.13.3 delete-protocol

Removes a protocol.

Synopsis

```
asadmin [asadmin-options] delete-protocol [--help]
    [--target target] protocol-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-protocol` subcommand removes the specified protocol. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target from which you are deleting the protocol.

Type: String

The following values can be specified:

- `server`
  Deletes the protocol from the default `server` instance. This is the default value.
- *configuration-name*
  Deletes the protocol from the specified configuration.
- *cluster-name*
  Deletes the protocol from all server instances in the specified cluster.
- *standalone-instance-name*
  Deletes the protocol from the specified standalone server instance.

Default value: `server`

protocol-name

Specifies the name of the protocol.

Type: String

The following values can be specified:

- *Name of the protocol*

Default value: N/A

Examples

The following command deletes the protocol named `http-1` on the server instance `instance1`.

```
asadmin delete-protocol --target instance1 http-1
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.13.4 delete-ssl

Deletes the SSL element in the selected HTTP listener, IIOP listener, or IIOP service.

Synopsis

```
asadmin [asadmin-options] delete-ssl [--help]
    [--target target] --type listener_or_service_type
    listener_id
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-ssl` subcommand deletes the SSL element in the selected HTTP listener, IIOP listener, or IIOP service. The *listener_id* is not required if the `--type` is `iiop-service`.

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target`

Specifies the target on which you are configuring the SSL element.

Type: String

The following values can be specified:

- `server`

  Specifies the server in which the iiop-service or HTTP/IIOP listener is to be unconfigured for SSL.
- *configuration_name*

  Specifies the configuration that contains the HTTP/IIOP listener or iiop-service for which SSL is to be unconfigured.
- *cluster_name*

  Specifies the cluster in which the HTTP/IIOP listener or iiop-service is to be unconfigured for SSL. All the server instances in the cluster will get SSL unconfigured for the respective listener or iiop-service.
- *instance_name*

  Specifies the instance in which the HTTP/IIOP listener or iiop-service is to be unconfigured for SSL.

Default value: `server`

`--type`

Specifies the type of service or listener for which the SSL is deleted.

Type: String

The following values can be specified:

- `http-listener`
- `iiop-listener`
- `iiop-service`

Default value: N/A

listener_id

Specifies the ID of the listener from which the SSL element is to be deleted.

The *listener_id* operand is not required if the `--type` is `iiop-service`.

Type: String

The following values can be specified:

- *ID of the listener*

Default value: N/A

Examples

The following example shows how to delete an SSL element from an HTTP listener named `http-listener-1` on the server instance `instance1`.

```
asadmin delete-ssl --target instance1 --type http-listener http-listener-1
```

Exit Status

Exit Status	Explanation
`0`	command executed successfully.
`1`	error in executing the command.

2.13.5 list-network-listeners

Lists the existing network listeners.

Synopsis

```
asadmin [asadmin-options] list-network-listeners [--help]
    [target]
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `list-network-listeners` subcommand of asadmin lists the existing network listeners. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

target

Restricts the listing to network listeners for a specified target.

Type: String

The following values can be specified:

- `server`
  Lists the network listeners for the default server instance. This is the default value.
- *configuration-name*
  Lists the network listeners for the specified configuration.
- *cluster-name*
  Lists the network listeners for all the server instances in the specified cluster.
- *instance-name*
  Lists the network listeners for the specified server instance.

Default value: `server`

Examples

The following command lists all the network listeners for the server instance.

```
asadmin list-network-listeners
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.13.6 list-protocols

Lists the existing protocols.

Synopsis

```
asadmin [asadmin-options] list-protocols [--help]
    [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-protocols` subcommand of `asadmin` lists the existing protocols. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`target`

Restricts the listing of protocols for a specific target.

Type: String

The following values can be specified:

- `server`
  Lists the protocols for the default server instance. This is the default value.
- *configuration-name*
  Lists the protocols for the specified configuration.
- *cluster-name*
  Lists the protocols for all the server instances in the specified cluster.
- *instance-name*
  Lists the protocols for the specified server instance.

Default value: `server`

Examples

The following example lists all the protocols for the server instance.

```
asadmin list-protocols
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.13.7 list-transports

Lists the existing transports.

Synopsis

```
asadmin [asadmin-options] list-transports [--help]
    target
```

Storage location

`Application Server installation directory``/javaee/glassfish/bin`

Function

The `list-transports` subcommand of asadmin lists the existing transports. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`target`

Limits the listing of transports for a specified target.

Type: String

The following values can be specified:

- `server`
  Lists the transports for the default server instance. This is the default value.
- *configuration-name*
  Lists the transports for the specified configuration.
- *cluster-name*
  Lists the transports for all server instances in the specified cluster.
- *instance-name*
  Lists the transports for the specified server instance.

Default value: `N/A`

Examples

The following example lists all the transports for the server instance.

```
asadmin list-transports server
```

Exit Status

Exit Status	Explanation
`0`	subcommand executed successfully.
`1`	error in executing the subcommand.

2.13.8 list-virtual-servers

Lists the existing virtual servers.

Synopsis

```
asadmin [asadmin-options] list-virtual-servers [--help]
    [--target target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-virtual-servers` subcommand of asadmin lists the existing virtual servers. This subcommand is supported in remote mode only.

Precondition

- Domain Administration Server (DAS) is running.
- Virtual servers must exist on the specified *target*.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Limits the listing to virtual servers for a specified target.

Type: String

The following values can be specified:

- `server`
  Lists the virtual servers for the default server instance. This is the default value.
- *configuration-name*
  Lists the virtual servers for the specified configuration.
- *cluster-name*
  Lists the virtual servers for all the server instances in the specified cluster.
- *instance-name*
  Lists the virtual servers for the specified server instance.

Default value: `server`

Examples

The following example lists all the virtual servers for the server instance `instance1`.

```
asadmin list-virtual-servers --target instance1
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.

Exit Status	Explanation
1	error in executing the subcommand.

2.14 Commands used for ORB administration

This section describes the syntax and functionality of the commands used for ORB administration.

2.14.1 create-iiop-listener

Adds an IIOP listener.

Synopsis

```
asadmin [asadmin-options] create-iiop-listener [--help]
    --listeneraddress address [--iiopport iiop-port-number]
    [--securityenabled={false|true}] [--enabled={true|false}]
    [--target target] listener_id
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-iiop-listener` subcommand of `asadmin` creates an IIOP listener.

This command is supported in remote mode only.

Precondition

DAS instance has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--listeneraddress` *address*

Displays either the IP address or the hostname (resolvable by DNS).

Type: String

The following values can be specified:

- *Specify the IP address*

Default value: N/A

`--iiopport` *iiop-port-number*

Specifies the IIOP port number. The default value is `1072`.

Type: Integer

The following values can be specified:

- `1` to `65535`

Default value: `1072`

`--securityenabled={false|true}`

Specifies that the IIOP listener runs SSL, if set to `true`. You can turn `ON` or `OFF` the SSL2 or SSL3 and set ciphers using an SSL element.

The security setting globally enables or disables SSL by making certificates available to the server instance. The default value is `false`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--enabled={true|false}`

Enables the IIOP listener at runtime, if set to `true`. The default value is `true`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--target` *`target`*

Specifies the target for which an IIOP listener is created.

Type: String

The following values can be specified:

- `server`
  Creates the listener for the default server instance `server`. This is the default value.
- *configuration_name*
  Creates the listener for the named configuration.
- *cluster_name*
  Creates the listener for every server instance in the cluster.
- *stand-alone_instance_name*
  Creates the listener for a particular standalone server instance.

Default value: `server`

`listener_id`

Creates a unique identifier for the IIOP listener.

Type: String

The following values can be specified:

- *Specify the listener ID*

Default value: N/A

Examples

The following example creates an IIOP listener on the server instance `instance1`.

```
asadmin create-iiop-listener --target instance1 --listeneraddress 192.168.1.100
--iiopport 1400 sample_iiop_listener
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.14.2 delete-iiop-listener

Removes an IIOP listener.

Synopsis

```
asadmin [asadmin-options] delete-iiop-listener [--help]
    [--target target] listener_id
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-iiop-listener` subcommand removes the specified IIOP listener.

This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`
Displays the help text for the subcommand.

`--target` *target*
Specifies the target from which you are deleting the IIOP listener.
Type: String
The following values can be specified:

- `server`
  Deletes the listener from the default server instance `server` and is the default value.
- *configuration_name*
  Deletes the listener from the named configuration.
- *cluster_name*
  Deletes the listener from every server instance in the cluster.

- *instance_name*
  Deletes the listener from a particular server instance.

Default value: `server`

`listener_id`

Specifies the unique identifier for the IIOP listener to be deleted.

Type: String

The following values can be specified:

- *The unique identifier for the IIOP listener*

Default value: N/A

Examples

The following command deletes the IIOP listener named `sample_iiop_listener` on the server instance `instance1`.

```
asadmin> delete-iiop-listener --target instance1 sample_iiop_listener
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.14.3 list-iiop-listeners

Lists the existing IIOP listeners.

Synopsis

```
asadmin [asadmin-options] list-iiop-listeners [--help] [target]
```

Storage location

`Application Server installation directory``/javaee/glassfish/bin`

Function

The `list-iiop-listeners` subcommand lists the existing IIOP listeners. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`target`

Specifies the target for which the IIOP listeners are to be listed.

Type: String

The following values can be specified:

- `server`
  Lists the listeners in the default server instance `server` and is the default value.
- *configuration_name*
  Lists the listeners in the specified configuration.
- *cluster_name*
  Lists the listeners in the specified cluster.
- *instance_name*
  Lists the listeners in a particular server instance.

Default value: `server`

Examples

The following command lists all the IIOP listeners for the server instance.

```
asadmin list-iiop-listeners
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.15 Commands used for resource administration

This section describes the syntax and functionality of the commands used for resource administration.

2.15.1 add-resources

Creates the resources named in a specified XML file.

Synopsis

```
asadmin [asadmin-options] add-resources [--help] [--target target]
    [--upload={false|true}] xml-file-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `add-resources` subcommand creates the resources named in a specified XML file. The `DOCTYPE` has to be specified as `http://glassfish.org/dtds/glassfish-resources_1_5.dtd` in the `resources.xml` file.

This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Files

Specify an XML file in any location by using an absolute path.

Arguments

`--help | -?`
Displays the help text for the subcommand.

`--target` *target*
Specifies the target for which you are creating the resources.
Type: String

- `server`
  Creates resources for the default server instance. This is the default value.
- `domain`
  Creates resources for a domain.
- *cluster-name*
  Creates resources for every server instance in a cluster.
- *instance-name*
  Creates resources for a specific server instance.

Default value: `server`

`--upload={false|true}`

Specifies whether the subcommand uploads the file to the DAS

Type: Boolean

The following values can be specified:

- `true`

  The subcommand uploads the file to the DAS over a network connection. The default value depends on whether the DAS is on the host where the subcommand is run or is on a remote host.

- `false`

  The subcommand does not upload the file and attempts to access the file through a specified file name. If the DAS cannot access the file, the subcommand fails. For example, the DAS may be running as a different user than the administration user and does not have read access to the file. In this situation, the subcommand fails if the `--upload` option is `false`.

Default value:

If the DAS is on the host where the subcommand is run, the default is `false`.

If the DAS is on a remote host, the default is `true`.

`xml-file-name`

Specifies the name of the xml file that contains the resources.

Type: String

The following values can be specified:

- *Name of the .xml file*

Default value: N/A

Examples

The following example creates resources using the contents of the XML file `resource.xml` on the server instance `instance1`.

```
asadmin add-resources --target instance1
domain-dir/domain_name/config/resource.xml
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.15.2 create-resource-ref

Creates a reference to a resource.

Synopsis

```
asadmin [asadmin-options] create-resource-ref [--help]
    [--target target] [--enabled={false|true}] reference_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-resource-ref` subcommand creates a reference from a cluster or an unclustered server instance to a previously created resource. The target instance or instances making up the cluster need not be running or available for this subcommand to succeed. If one or more instances are not available, they will receive the new resource the next time they start.

This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target for which you are creating the resource reference.

Type: String

The following values can be specified:

- `server`
  Creates the resource reference for the default server instance. This is the default target.
- *cluster_name*
  Creates the resource reference for every server instance in the cluster.
- *instance_name*
  Creates the resource reference for the named unclustered server instance.

Default value: `server`

`--enabled={false|true}`

Indicates whether the resource should be enabled. This value will take effect only if the resource is enabled at the global level.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

reference_name

Specifies the name or JNDI name of the resource.

Type: String

The following values can be specified:

- *Name or JNDI name of the resource*

Default value: N/A

Examples

The following example creates a reference to the JDBC resource named `jdbc/DerbyPool` on the cluster `Cluster1`.

```
asadmin create-resource-ref --target Cluster1 jdbc/DerbyPool
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.15.3 delete-resource-ref

Removes a reference to a resource.

Synopsis

```
asadmin [asadmin-options] delete-resource-ref [--help]
    [--target target] reference_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-resource-ref` subcommand removes from a cluster or an unclustered server instance a reference to a resource (for example, a JDBC resource). This effectively results in the removal of the resource from the JNDI tree of the targeted instance or cluster. The target instance or instances making up the cluster need not be running or available for this subcommand to succeed. If one or more instances are not available, they will no longer load the resource in the JNDI tree the next time they start.

Removal of the reference does not result in removal of the resource from the domain. The resource is removed only by the `delete` subcommand for that resource (for example, `delete-jdbc-resource`).

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target from which you are removing the resource reference.

Type: String

The following values can be specified:

- `server`
  Removes the resource reference from the default server instance `server` and is the default value.
- *cluster_name*
  Removes the resource reference from every server instance in the cluster.
- *instance_name*
  Removes the resource reference from the named unclustered server instance.

Default value: `server`

reference_name

Specifies the name or JNDI name of the resource.

Type: String

The following values can be specified:

- *Name or JNDI name of the resource*

Default value: N/A

Examples

The following example removes a reference to the JDBC resource named `jdbc/DerbyPool` on the cluster `cluster1`.

```
asadmin delete-resource-ref --target cluster1 jdbc/DerbyPool
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.15.4 list-resource-refs

Lists the existing resource references.

Synopsis

```
asadmin [asadmin-options] list-resource-refs [--help]
    [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The list-resource-refs subcommand of asadmin lists all the resource references in a cluster or an unclustered server instance. This effectively lists all the resources (for example, JDBC resources) available in the JNDI tree of the specified target. This subcommand is supported in remote mode only.

The target instance or instances in the cluster need not be running or available for this command to succeed.

Precondition

Domain Administration Server (DAS) is running.

Arguments

--help | -?

Displays the help text for the subcommand.

target

Displays the target for which the resource references are to be listed.

Type: String

The following values can be specified:

- server
  Lists the resource references for the default server instance. This is the default value.
- *cluster_name*
  Lists the resource references for every server instance in the cluster.
- *instance_name*
  Lists the resource references for the named unclustered server instance.

Default value: server

Examples

The following example lists resource references for the cluster cluster1.

```
asadmin list-resource-refs cluster1
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.16 Commands used for database connection administration

This section describes the syntax and functionality of the commands used for database connection administration.

2.16.1 create-jdbc-connection-pool

Registers a JDBC connection pool.

Synopsis

```
asadmin [asadmin-options] create-jdbc-connection-pool [--help]
    [--datasourceclassname=datasourceclassname]
    [--restype=resourcetype]
    [--steadypoolsize=poolsize]
    [--maxpoolsize=maxpoolsize]
    [--maxwait=maxwaittime]
    [--poolresize=poolresizelimit]
    [--idletimeout=idletimeout]
    [--isolationlevel=isolationlevel]
    [--isisolationguaranteed={true|false}]
    [--isconnectvalidatereq={false|true}]
    [--validationmethod=validationmethod]
    [--validationtable=validationtable]
    [--nontransactionalconnections={false|true}]
    [--validateatmostonceperiod=validationinterval]
    [--leaktimeout=leaktimeout]
    [--leakreclaim={false|true}]
    [--creationretryattempts=creationretryattempts]
    [--creationretryinterval=creationretryinterval]
    [--statementtimeout=statementtimeout]
    [--lazyconnectionenlistment={false|true}]
    [--lazyconnectionassociation={false|true}]
    [--driverclassname=jdbcdriverclassname]
    [--matchconnections={false|true}]
    [--maxconnectionusagecount=maxconnectionusagecount]
    [--ping={false|true}]
    [--pooling={false|true}]
    [--statementcachesize=statementcachesize]
    [--validationclassname=validationclassname]
    [--description description]
    [--property name=value[:name=value]...]
    connectionpoolid
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-jdbc-connection-pool` subcommand registers a new Java Database Connectivity ("JDBC") software connection pool with the specified JDBC connection pool name.

A JDBC connection pool with authentication can be created by doing the following:

- Using a `--property` option to specify a user, password, or other connection information
- Specifying the connection information in the XML descriptor file.

Specify the following options depending on the database you want to connect to:

- Specify the following options to connect to Oracle:

Option		Value	Remark
--datasourceclassname		oracle.jdbc.pool.OracleDataSource	Refer to #1.
		oracle.jdbc.xa.client.OracleXADataSource	Refer to #2.
--restype		javax.sql.DataSource	Refer to #1.
		javax.sql.XADataSource	Refer to #2.
--validationmethod		custom-validation	Refer to #3.
--validationclassname		org.glassfish.api.jdbc.validation.OracleConnectionValidation	Refer to #3.
--property	databaseName	*Oracle SID*	Refer to #4.
	serverName	*Oracle-host-name-or-IP-address*	Refer to #4.
	portNumber	*port-number*	Refer to #4.
	user	*user-name*	-
	password	*password*	-
	driverType	thin	Refer to #4.
	url	jdbc:oracle:thin:@*Oracle-host-name-or-IP-address*:*port-number*:*Oracle SID*	Refer to #4.

#1:

Specify this option if the XA transactions must not be used.

#2:

Specify this option if the XA transactions must be used.

#3:

Specify this option only when the connection errors must be detected.

#4:

Specify the four properties (`databaseName`, `serverName`, `portNumber`, and `driverType`) or `url`. If you specify both, `url` takes effect but the other four properties are ignored.

This command is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Files

Specify the connection information in the XML descriptor when not using the `--property` option to create a jdbc connection pool.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--datasourceclassname=`*`datasourceclassname`*

Specifies the name of the vendor-supplied JDBC datasource resource manager. Any XA or global transactions capable datasource class will implement the `javax.sql.XADatasource` interface.

Type: String

The following values can be specified:

- *Specify the datasource class name*

Default value: N/A

`--restype=`*`resourcetype`*

Indicates when a datasource class implements two or more interfaces (`javax.sql.DataSource`, `javax.sql.XADataSource`, or `javax.sql.ConnectionPoolDataSource`) or when a driver classname must be provided.

If `--restype` = `java.sql.Driver`, then the `--driverclassname` option is required.

If `--restype` = `javax.sql.DataSource`, `javax.sql.XADataSource`, or `javax.sql.ConnectionPoolDataSource`, then the `--datasourceclassname` option is required.

If `--restype` is not specified, then either the `--driverclassname` or `--datasourceclassname` option must be specified, but not both.

Type: String

The following values can be specified:

- `java.sql.Driver`
- `javax.sql.DataSource`
- `javax.sql.XADataSource`
- `javax.sql.ConnectionPoolDataSource`

Default value: N/A

`--steadypoolsize=`*`poolsize`*

Specifies the minimum and initial number of connections maintained in the pool.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `24`

The default value of the `steady-pool-size` attribute of the `connector-connection-pool` element of the `glassfish-resources.xml` file is same as the default value of this option.

`--maxpoolsize=`*`maxpoolsize`*

Specifies the maximum number of connections that can be created.

Type: Integer

The following values can be specified:

- `1` to `2147483647`

Default value: `24`

The default value of the `max-pool-size` attribute of the `connection-pool` element is same as the default value of this option.

`--maxwait=`*`maxwaittime`*

Specifies the amount of time, in milliseconds, that a caller has to wait for before a connection timeout is sent.

A value of `0` forces the caller to wait indefinitely.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `60000`

`--poolresize=`*`poolresizelimit`*

Specifies the number of connections to be removed when `idle-timeout-in-seconds` timer expires.

Type: Integer

The following values can be specified:

- `1` to `2147483647`

Default value: `2`

`--idletimeout=`*`idletimeout`*

Specifies the maximum time, in seconds, that a connection can remain idle in the pool.

This timeout value must be shorter than the database server side timeout value to prevent the accumulation of unusable connections in the application.

If `0` is specified, then this option is disabled.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `300`

`--isolationlevel=`*`isolationlevel`*

Specifies the `transaction-isolation-level` that can be set on the pooled database connections.

Applications that change the isolation level on a pooled connection programmatically risk polluting the pool. This could lead to program errors.

Type: String

The following values can be specified:

- `read-uncommitted`

- `read-committed`
- `repeatable-read`
- `serializable`

Default value: Isolation level set by JDBC driver provider

`--isisolationguaranteed={true|false}`

Applies only when a specific isolation level is specified for the `transaction-isolation-level`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--isconnectvalidatereq={false|true}`

Validates or checks the connections to see if they are usable before being given out to the application, if set to `true`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--validationmethod=`*`validationmethod`*

Specifies the type of validation to be performed when the `--isconnectvalidatereq` option is `true`.

- If `auto-commit` is specified: Executes `Connection#setAutoCommit(Connection#getAutoCommit())`.
- If `meta-data` is specified: Executes `Connection#getMetaData()`.
- If `table` is specified: Execute the `SELECT` sentence to the table specified by the `--validationtable` option.
- If `custom-validation` is specified: Execute the user-defined class specified by the `--validationclassname` option.

Type: String

The following values can be specified:

- `auto-commit`
- `meta-data`
- `table`
- `custom-validation`

Default value: `table`

`--validationtable=`*`validationtable`*

Specifies the name of the validation table used to perform a query to validate a connection.

If the `--isconnectvalidatereq` option is set to `true` and the `--validationmethod` set to `table`, it is mandatory to set this option.

Type: String

The following values can be specified:

- *Specify the validation table name*

Default value: N/A

`--nontransactionalconnections={false|true}`

Returns non-transactional connections when a pool with this property is set to `true`. This connection does not get enlisted automatically with the transaction manager.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--validateatmostonceperiod=`*`validationinterval`*

Specifies the time interval, in seconds, between successive requests to validate a connection at most once. Setting this attribute to an appropriate value minimizes the number of validation requests by a connection.

If `0` is specified, then the connection is always validated.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `0`

`--leaktimeout=`*`leaktimeout`*

Specifies the amount of time, in seconds, for which connection leaks in a connection pool are to be traced. When a connection is not returned to the pool by the application within the specified period, it is assumed to be a potential leak, and the stack trace of the caller will be logged. This option only detects if there is a connection leak. The connection can be reclaimed only if `--leakreclaim` option is set to `true`.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `0`

`--leakreclaim={false|true}`

Specifies whether leaked connections are restored to the connection pool after leak connection tracing is complete.

Type: Boolean

The following values can be specified:

- `false`

  Leaked connections are not restored to the connection pool. This is the default value.

- `true`

  Leaked connections are restored to the connection pool.

Default value: `false`

`--creationretryattempts=`*`creationretryattempts`*

Specifies the maximum number of times that Java EE Server tries to re-create a connection if the initial attempt fails.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `0`

`--creationretryinterval=`*`creationretryinterval`*

Specifies the interval, in seconds, between successive attempts to create a connection.

If `--creationretryattempts` is `0`, the `--creationretryinterval` option is ignored.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `10`

`--statementtimeout=`*`satementleaktimeout`*

Specifies the length of time, in seconds, after which a query that is not completed is terminated.

Type: Integer

The following values can be specified:

- `-1` to `2147483647`

Default value: `-1`

`--lazyconnectionenlistment={false|true}`

Specifies whether a resource to a transaction is enlisted only when a method actually uses the resource.

Type: Boolean

The following values can be specified:

- `false`

  Resources to a transaction are always enlisted and not only when a method actually uses the resource. This is the default value.

- `true`

  Resources to a transaction are enlisted only when a method actually uses the resource.

Default value: `false`

`--lazyconnectionassociation={false|true}`

Specifies that a physical connection should be associated with the logical connection only when the physical connection is used, and disassociated when the transaction is completed. Such association and dissociation enable the reuse of physical connections.

Type: Boolean

The following values can be specified:

- `false`

  A physical connection is associated with the logical connection even before the physical connection is used, and is not disassociated when the transaction is completed. This is the default value.

- `true`

  A physical connection is associated with the logical connection only when the physical connection is used, and disassociated when the transaction is completed. The `--lazyconnectionenlistment` option must also be set to `true`.

Default value: `false`

`--driverclassname=`*`jdbcdriverclassname`*

Specifies the name of the vendor-supplied JDBC driver class. This driver should implement the `java.sql.Driver` interface.

Type: String

The following values can be specified:

- Java class implementing `java.sql.Driver` interface

Default value: N/A

`--matchconnections={false|true}`

Specifies whether a connection that is selected from the pool should be matched by the resource adaptor.

Type: Boolean

The following values can be specified:

- `false`

  A connection must not be matched by the resource adaptor. This is the default value.

- `true`

  A connection must be matched by the resource adaptor. If all the connections in the pool are homogenous, a connection picked from the pool need not be matched by the resource adapter. This means that this option can be set to `false`.

Default value: `false`

`--maxconnectionusagecount=`*`maxconnectionusagecount`*

Specifies the maximum number of times that a connection can be reused. When this limit is reached, the connection is closed. By limiting the maximum number of times that a connection can be reused, statement leaks can be avoided.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `0`

`--ping={false|true}`

Specifies if the pool is pinged during pool creation or reconfiguration to identify and warn of any erroneous values for its attributes.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--pooling={false|true}`

Specifies if connection pooling is enabled for a pool.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--statementcachesize=`*`statementcachesize`*

Specifies the number of SQL statements to be cached using the default caching mechanism.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `0`

`--validationclassname=`*`validationclassname`*

Specifies the name of the class that provides custom validation when the value of `--validationmethod` is `custom-validation`. This class must implement the `org.glassfish.api.jdbc.ConnectionValidation` interface, and it must be accessible to Java EE Server.

This option is mandatory if the connection validation type is set to custom validation.

Type: String

The following values can be specified:

- Implementers of `org.glassfish.api.jdbc.ConnectionValidation` interface

Default value: N/A

`--description` *`description`*

Specifies the text providing details about the specified JDBC connection pool.

Type: String

The following values can be specified:

- *Specify the description in text*

Default value: N/A

`--property` *`name`*`=`*`value`*`[:`*`name`*`=`*`value`*`]...`

Specifies the optional attribute `name`/`value` pairs for configuring the pool. The specification format is `name=`*`value`*. Use a colon (`:`) to separate property names. If you specify the same property name more than once, the property value specified last is valid.

For this option, you can specify the DataSource property supported by the JDBC driver, and a property that is mentioned below. For details on properties that can be specified, see the documentation for the database that you are using.

Type: String

The following values can be specified:

- `user=`*`value`*

  Specifies the user name required to connect to the database.

  Type: String

  Default value: N/A

  Range Value: N/A

- `password=`*`value`*

  Specifies the password required to connect to the database.

  Type: String

  Default value: N/A

  Range Value: N/A

- `databaseName=`*`value`*

  Specifies the database for this connection pool.

  Type: String

  Default value: N/A

  Range Value: N/A

- `serverName=`*`value`*

Specifies the database server for this connection pool.
Type: String
Default value: N/A
Range Value: N/A

- `portNumber=`*`value`*
  Specifies the port on which the database server listens for requests.
  Type: Integer
  Default value: N/A
  Range Value: `1` to `65535`
- `networkProtocol=`*`value`*
  Specifies the communication protocol.
  Type: String
  Default value: N/A
  Range Value: N/A
- `roleName=`*`value`*
  Specifies the initial SQL role name.
  Type: String
  Default value: N/A
  Range Value: N/A
- `datasourceName=`*`value`*
  Specifies an underlying XADataSource or a `connectionPoolDataSource` if the connection pooling is complete.
  Type: String
  Default value: N/A
  Range Value: N/A
- `description=`*`value`*
  Specifies a text description.
  Type: String
  Default value: N/A
  Range Value: N/A
- `url=`*`value`*
  Specifies the URL for this connection pool. Although this is not a standard property, it is commonly used.
  Type: String
  Default value: N/A
  Range Value: N/A
- `dynamic-reconfiguration-wait-timeout-in-seconds=`*`value`*
  Used to enable dynamic reconfiguration of the connection pool, transparently to the applications that are using the pool, so that applications need not be re-enabled for the attribute or property changes to the pool to take effect. Any in-flight transaction's connection requests will be allowed to complete with the old pool configuration as long as the connection requests are within the timeout period, so as to complete the transaction. New connection requests will wait for the pool reconfiguration to complete and connections will be acquired using the modified pool configuration.
  Type: Long

Default value: 0

Range Value: x to y (Where x*1000 is >= -9223372036854775808 and y*1000 is <= 9223372036854775807)

Default value: N/A

connectionpoolid

Specifies the name of the JDBC connection pool to be created.

Type: String

The following values can be specified:

- *Specify the connection pool ID*

Default value: N/A

Examples

The following example creates a jdbc connection pool.

```
asadmin create-jdbc-connection-pool
--datasourceclassname org.apache.derby.jdbc.ClientDataSource
--restype javax.sql.XADataSource
--property portNumber=1527:password=APP:user=APP:serverName=
localhost:databaseName=sun-appserv-samples:connectionAttributes=\;
create\\=true sample_derby_pool
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.16.2 create-jdbc-resource

Creates a JDBC resource with the specified JNDI name.

Synopsis

```
asadmin [asadmin-options] create-jdbc-resource [--help]
    --connectionpoolid connectionpoolid [--enabled={false|true}]
    [--description description][--target target] jndi_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The create-jdbc-resource subcommand creates a new JDBC resource.

This command is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--connectionpoolid` *`connectionpoolid`*

Specifies the name of the JDBC connection pool. If two or more JDBC resource elements point to the same connection pool element, they use the same pool connection at runtime.

Type: String

The following values can be specified:

- *Specify connection pool ID*

Default value: N/A

`--enabled={false|true}`

Determines whether the JDBC resource is enabled at runtime.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--description` *`description`*

Specifies the text providing descriptive details about the JDBC resource.

Type: String

The following values can be specified:

- *Provide the description*

Default value: N/A

`--target` *`target`*

Specifies the target to which an application is deployed.

The resource is always created for the domain as a whole but the `resource-ref` for the resource is only created for the specified `--target`. This means that although the resource is defined at the domain level, it is only available at the specified target level. Use the `create-resource-ref` subcommand to refer to the resource in multiple targets if needed.

Type: String

The following values can be specified:

- `server`

  Deploys the component to the default server instance. This is the default value.

- `domain`

  Deploys the component to a domain.

- *cluster_name*

  Deploys the component to every server instance in the cluster.

- *instance_name*

Deploys the component to a particular server instance.

Default value: `server`

jndi_name

Specifies the JNDI name of this JDBC resource.

Type: String

The following values can be specified:

- *Specify the JNDI name*

Default value: N/A

Examples

The following example creates a jdbc-resource on the server instance `instance1`.

```
asadmin create-jdbc-resource --target instance1
--connectionpoolid sample_derby_pool jdbc/DerbyPool
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.16.3 delete-jdbc-connection-pool

Removes the specified JDBC connection pool.

Synopsis

```
asadmin [asadmin-options] delete-jdbc-connection-pool [--help]
    [--cascade={false|true}] jdbc_connection_pool_id
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-jdbc-connection-pool` subcommand deletes a JDBC connection pool. Before running this subcommand, all associations to the JDBC connection pool must be removed.

This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--cascade={false|true}`

Specifies the property to delete the connection pool and all JDBC resources associated with the pool.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`jdbc_connection_pool_id`

Specifies the name of the JDBC resource.

Type: String

The following values can be specified:

- *Name of the JDBC resource*

Default value: N/A

Examples

The following example deletes the `sample_derby_pool` JDBC connection pool.

```
asadmin delete-jdbc-connection-pool
 --cascade=false sample_derby_pool
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.16.4 delete-jdbc-resource

Removes a JDBC resource with the specified JNDI name.

Synopsis

```
asadmin [asadmin-options] delete-jdbc-resource [--help]
   [--target target] jndi_name
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `delete-jdbc-resource` subcommand removes a JDBC resource. Ensure that all associations to the JDBC resource are removed before running this subcommand. This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target from which you are removing the JDBC resource.

Resources are always created for a domain as a whole but are only active for targets for which a `resource-ref` has been created using the `--target` option when the resource was created. This means that deleting a resource only deletes the `resource-ref` element for the specified `--target`, and does not delete the resource from the domain as a whole unless domain is specified as the `--target` for the deletion.

Type: String

The following values can be specified:

- `server`
  Removes the resource from the default server instance. This is the default value.
- `domain`
  Removes the resource from the domain.
- *cluster_name*
  Removes the resource from every server instance in the cluster.
- *instance_name*
  Removes the resource from a particular server instance.

Default value: `server`

jndi_name

Specifies the JNDI name of the JDBC resource.

Type: String

The following values can be specified:

- *JNDI name of the JDBC resource*

Default value: N/A

Examples

The following example deletes the JDBC resource named `jdbc/DerbyPool` on the server instance `instance1`.

```
asadmin delete-jdbc-resource --target instance1 jdbc/DerbyPool
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.16.5 list-jdbc-connection-pools

Lists all JDBC connection pools.

Synopsis

```
asadmin [asadmin-options] list-jdbc-connection-pools [--help]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-jdbc-connection-pools` subcommand of asadmin lists the current JDBC connection pools. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

Examples

The following example lists the existing JDBC connection pools.

```
asadmin list-jdbc-connection-pools
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.16.6 list-jdbc-resources

Lists all JDBC resources.

Synopsis

```
asadmin [asadmin-options] list-jdbc-resources [--help]
    [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-jdbc-resources` subcommand of asadmin displays a list of all the existing JDBC resources. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

target

Specifies the JDBC resources you want to list.

Type: String

The following values can be specified:

- `server`
  Lists all the JDBC resources in the current server. This is the default value.
- `domain`
  Lists all the JDBC resources in the current domain.
- *cluster_name*
  Lists all the JDBC resources in a cluster.
- *instance_name*
  Lists all the JDBC resources for a particular instance.

Default value: `server`

Examples

The following example lists the current JDBC resources.

```
asadmin list-jdbc-resources
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.17 Commands used for JavaMail service administration

This section describes the syntax and functionality of the commands used for JavaMail service administration.

2.17.1 create-javamail-resource

Creates a JavaMail session resource.

Synopsis

```
asadmin [asadmin-options] create-javamail-resource [--help]
    [--target target] --mailhost hostname --mailuser username
    --fromaddress address [--storeprotocol storeprotocol]
    [--storeprotocolclass storeprotocolclass]
    [--transprotocol transprotocol]
    [--transprotocolclass transprotocolclass]
    [--enabled={true|false}] [--description resource-description]
    [--property name=value[:name=value]...] jndi-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-javamail-resource` subcommand of `asadmin` creates a JavaMail session resource. This command is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target for which the JavaMail session resource is created.

Type: String

The following values can be specified:

- `server`

  Creates the resource for the default server instance. This is the default value.

- `domain`

  Creates the resource for the domain.

- *cluster_name*

  Creates the resource for every server instance in the cluster.

- *instance_name*

  Creates the resource for a specific server instance.

Default value: `server`

`--mailhost` *`hostname`*

Specifies the DNS name of the default mail server. The connect methods of the `Store` and `Transport` objects use this value if a `protocol-specific` host property is not supplied.

Type: String

The following values can be specified:

- *Specify the mail hostname*

Default value: N/A

`--mailuser` *`username`*

Specifies the name of the mail account user provided when connecting to a mail server.

Type: String

The following values can be specified:

- *Specify the mail user name*

Default value: N/A

`--fromaddress` *`address`*

Specifies the email address of the default user, in the format `username@host.domain`.

Type: String

The following values can be specified:

- *Specify the From address*

Default value: N/A

`--storeprotocol` *`storeprotocol`*

Specifies the mail server store protocol.

Type: String

The following values can be specified:

- `imap`
- `pop3`

Default value: `imap`

`--storeprotocolclass` *`storeprotocolclass`*

Specifies the mail server store protocol class name.

Type: String

The following values can be specified:

- `com.sun.mail.imap.IMAPStore`
- `com.sun.mail.pop.POPStore`

Default value: `com.sun.mail.imap.IMAPStore`

`--transprotocol` *`transprotocol`*

Specifies the mail server transport protocol.

Type: String

The following values can be specified:

- `smtp`
- `lmtp`

Default value: smtp

--transprotocolclass *transprotocol*

Specifies the mail server transport protocol class name.

Type: String

The following values can be specified:

- com.sun.mail.smtp.SMTPTransport
- com.sun.mail.lmtp.LMTPTransport

Default value: com.sun.mail.smtp.SMTPTransport

--enabled={true|false}

Enables the resource at runtime, if set to true.

Type: Boolean

The following values can be specified:

- true
- false

Default value: true

--description *resource-description*

Specifies text that provides some details of the JavaMail resource.

Type: String

The following values can be specified:

- *Provide description of java mail resource*

Default value: N/A

--property *name*=*value*[:*name*=*value*]...

Specifies the optional attribute name/value pairs for configuring the JavaMail resource. Java EE Server-specific mail- prefix is converted to the standard mail prefix. The specification format is *name*=*value*. You can specify multiple values separated by a colon (:). If the same property name is specified multiple times, then the property name, which is specified at the end is valid.

For the detail about the value that you can specify the *name*, see the package summary of the javax.mail in the JavaDoc (http://docs.oracle.com/javaee/7/api/javax/mail/package-summary.html). The default value is nothing. For the *value*, you need to specify in the range of specification of the *property*.

name

Type: String

Default value: N/A

value

Type: String

The following values can be specified:

- See the package summary of javax.mail package in the JavaDoc of Java Mail Specification (JavaDoc http://docs.oracle.com/javaee/7/api/javax/mail/package-summary.html).

Default value: N/A

jndi-name

Specifies the JNDI name of the JavaMail resource to be created. It is a recommended practice to use the naming sub-context prefix mail/ for JavaMail resources.

Type: String

The following values can be specified:

- *Specify the JNDI name*

Default value: N/A

Examples

The following example creates a JavaMail resource on the server instance `instance1`.

```
asadmin create-javamail-resource --target instance1 --mailhost localhost
--mailuser sample --fromaddress sample@sun.com mail/MyMailSession
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.17.2 delete-javamail-resource

Removes a JavaMail session resource.

Synopsis

```
asadmin [asadmin-options] delete-javamail-resource [--help]
    [--target target] jndi_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-javamail-resource` subcommand removes the specified JavaMail session resource. Ensure that you remove all references to this resource before running this subcommand. This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target from which you are deleting the JavaMail session resource.

Type: String

The following values can be specified:

- `server`
  Deletes the resource from the default server instance. This is the default value.
- `domain`
  Deletes the resource from the domain.
- *cluster_name*
  Deletes the resource from every server instance in the cluster.
- *instance_name*
  Deletes the resource from a particular server instance.

Default value: `server`

jndi_name

Specifies the JNDI name of the JavaMail session resource.

Type: String

The following values can be specified:

- *JNDI name of the resource*

Default value: N/A

Examples

The following example deletes the JavaMail session resource named `mail/MyMailSession` on the server instance `instance1`.

```
asadmin delete-javamail-resource --target instance1 mail/MyMailSession
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.17.3 list-javamail-resources

Lists the existing JavaMail session resources.

Synopsis

```
asadmin [asadmin-options] list-javamail-resources [--help]
    [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-javamail-resources` subcommand lists the existing JavaMail session resources. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`target`

Specifies the target for which the JavaMail session resources are to be listed.

Type: String

The following values can be specified:

- `server`
  Lists the resources for the default server instance. This is the default value.
- `domain`
  Lists the resources for the domain.
- *cluster_name*
  Lists the resources for the specified cluster.
- *instance_name*
  Lists the resources for a particular server instance.

Default value: `server`

Examples

This example lists the JavaMail session resources for the server instance.

```
asadmin list-javamail-resources
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.18 Commands used for JNDI service administration

This section describes the syntax and functionality of the commands used for JNDI service administration.

2.18.1 create-custom-resource

Creates a custom resource.

Synopsis

```
asadmin [asadmin-options] create-custom-resource [--help]
    --restype type --factoryclass classname
    [--enabled={true|false}] [--description text]
    jndi-name [--target target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-custom-resource` subcommand creates a custom resource. A custom resource specifies a custom server-wide resource object factory that implements the `javax.naming.spi.ObjectFactory` interface.

This command is supported in remote mode only.

Restart Java EE Server after creating the custom resource.

Precondition

- DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the availability of the target on which the custom resource is being created.

The resource is always created for the domain as a whole, but the `resource-ref` for the resource is only created for the specified `--target`. This means that although the resource is defined at the domain level, it is only available at the specified target level. Use the `create-resource-ref` subcommand to refer to the resource in multiple targets if needed.

Type: String

The following values can be specified:

- `server`

  The resource will be available on the default server instance and on all domains hosted on the instance. This is the default value.

- `domain`

  The resource will be available on the specified domain only.

- *cluster_name*
  The resource will be available on every server instance in the cluster.
- *instance_name*
  The resource will be available on the specified server instance only.

Default value: `server`

`--restype` *`type`*

Specifies the type of custom resource to be created.

Type: String

The following values can be specified:

- *Custom resource type*

Default value: N/A

`--factoryclass` *`classname`*

Specifies the factory class name for the custom resource.

Type: String

The following values can be specified:

- *A class name which implements the javax.naming.spi.ObjectFactory interface.*

Default value: N/A

`--enabled={true|false}`

Confirms whether the custom resource is enabled at runtime.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--description` *`text`*

Displays the text providing details about the custom resource. This description is a string value and can include a maximum of `250` characters.

Type: String

The following values can be specified:

- *Specify the description in text*

Default value: N/A

`jndi-name`

Specifies the JNDI name of this resource.

Type: String

The following values can be specified:

- *JNDI name*

Default value: N/A

Examples

The following example creates a custom resource on the server instance `instance1`.

```
asadmin create-custom-resource --target instance1 --restype topic
--factoryclass com.imq.topic mycustomresource
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.18.2 create-jndi-resource

Registers a JNDI resource.

Synopsis

```
asadmin [asadmin-options] create-jndi-resource [--help]
    [--target target] --restype restype --factoryclass factoryclass
    --jndilookupname jndilookupname [--enabled={true|false}]
    [--description description] jndi-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-jndi-resource` subcommand registers a JNDI resource.

This command is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target for which you are registering a JNDI resource.

The resource is always created for the domain as a whole but the `resource-ref` for the resource is only created for the specified `--target`. This means that although the resource is defined at the domain level, it is only available at the specified target level. Use the `create-resource-ref` subcommand to refer to the resource in multiple targets if required.

Type: String

The following values can be specified:

- `server`

  Creates the resource for the default server instance. This is the default value.

- `domain`
  Creates the resource for the domain.
- *cluster-name*
  Creates the resource for every server instance in the cluster.
- *instance-name*
  Creates the resource for a particular server instance.

Default value: `server`

`--restype` *restype*

Displays the JNDI resource type.

Type: String

The following values can be specified:

- `topic`
- `queue`

Default value: N/A

`--factoryclass` *factoryclass*

Displays the class that creates the JNDI resource.

Type: String

The following values can be specified:

- A class name implementing the `javax.naming.spi.InitialContextFactory` interface.

Default value: N/A

`--jndilookupname` *jndilookupname*

Displays the lookup name that the external container uses.

Type: String

The following values can be specified:

- *Specify the jndi lookup name*

Default value: N/A

`--enabled={true|false}`

Determines whether the resource is enabled at runtime.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--description` *description*

Provides details about the JNDI resource.

Type: String

The following values can be specified:

- *Text description*

Default value: N/A

jndi-name

Displays the unique name of the JNDI resource to be created.

Type: String

The following values can be specified:

- *Specify the JNDI name*

Default value: N/A

Examples

The following example creates a JNDI resource on the server instance `instance1`.

```
asadmin create-jndi-resource --target instance1
--restype com.example.jndi.MyResourceType
--factoryclass com.example.jndi.MyInitialContextFactoryClass
--jndilookupname remote-jndi-name --description "sample JNDI resource"
my-jndi-resource
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.18.3 delete-custom-resource

Deletes a custom resource.

Synopsis

```
asadmin [asadmin-options] delete-custom-resource [--help]
    [--target target] jndi-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-custom-resource` subcommand removes a custom resource.

This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target from which you are deleting the custom resources.

Type: String

The following values can be specified:

- `server`
  Deletes the resource for the default server instance. This is the default value.
- `domain`
  Deletes the resource for the domain.
- *cluster-name*
  Deletes the resource for every server instance in the cluster.
- *instance-name*
  Deletes the resource for a particular server instance.

Default value: `server`

jndi_name

Specifies the JNDI name of the resource.

Type: String

The following values can be specified:

- *JNDI name of the resource*

Default value: N/A

Examples

The following example deletes a custom resource named `mycustomresource` on the server instance `instance1`.

```
asadmin delete-custom-resource --target instance1 mycustomresource
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.18.4 delete-jndi-resource

Removes a JNDI resource.

Synopsis

```
asadmin [asadmin-options] delete-jndi-resource [--help]
    [--target target] jndi_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-jndi-resource` subcommand removes the specified JNDI resource. Ensure that all associations to the JNDI resource are removed before running this subcommand. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target from which you are removing the JNDI resource.

Resources are always created for a domain as a whole but are only active for targets for which a `resource-ref` has been created using the `--target` option when the resource was created. This means that deleting a resource only deletes the `resource-ref` element for the specified `--target` and does not delete the resource from the `domain` as a whole unless `domain` is specified as the `--target` for the deletion.

Type: String

The following values can be specified:

- `server`
  Deletes the resource from the default server instance. This is the default value.
- `domain`
  Deletes the resource from the domain.
- *cluster_name*
  Deletes the resource for every server instance in the cluster.
- *instance_name*
  Deletes the resource from the specified server instance.

Default value: `server`

jndi_name

Specifies the name of the JNDI resource.

Type: String

The following values can be specified:

- *Name of the JNDI resource*

Default value: N/A

Examples

The following example removes an existing JNDI resource named `sample_jndi_resource` on the server instance `instance1`.

```
asadmin delete-jndi-resource --target instance1 sample_jndi_resource
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.18.5 list-custom-resources

Gets all the custom resources.

Synopsis

```
asadmin [asadmin-options] list-custom-resources [--help]
    [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-custom-resources` subcommand lists the custom resources. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

target

Specifies the location of the custom resources.

Type: String

The following values can be specified:

- `server`

  Lists the resources on the default server instance. This is the default value.

- `domain`

  Lists the resources in the domain.

- *cluster_name*
  Lists the resources for every server instance in the cluster.
- *instance_name*
  Lists the resources for a particular server instance.

Default value: `server`

Examples

This example lists the current custom resources.

```
asadmin list-custom-resources
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.18.6 list-jndi-resources

Lists all the existing JNDI resources.

Synopsis

```
list-jndi-resources [--help] [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-jndi-resources` subcommand of asadmin identifies all the existing JNDI resources. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`
Displays the help text for the subcommand.

target
Specifies the target for which the JNDI resources are to be listed.
Type: String
The following values can be specified:

- `server`
  Lists the JNDI resources on the default `server` instance. This is the default value.
- *configuration-name*
  Lists the JNDI resources for the specified configuration.
- *cluster-name*
  Lists the JNDI resources on all the server instances in the specified cluster.
- *instance-name*
  Lists the JNDI resources on a specified server instance.

Default value: `server`

Examples

The following example lists the JNDI resources on the default server instance.

```
asadmin list-jndi-resources
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.19 Commands used for JMS administration

This section describes the syntax and functionality of the commands used for JMS administration.

2.19.1 create-jms-host

Creates a JMS host.

Synopsis

```
asadmin [asadmin-options] create-jms-host [--help] --mqhost mq-host
    --mqport mq-port --mquser mq-user --mqpassword mq-password
    [--target target] [--force={false|true}] jms_host_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-jms-host` subcommand creates a Java Message Service (JMS) host within the JMS service.

This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`
Displays the help text for the subcommand.

`--mqhost` *mq-host*
Specifies the host name for the JMS service.
Type: String
The following values can be specified:

- *Name of the host*

Default value: `localhost`

`--mqport` *mq-port*
Specifies the port number used by the JMS service.
Type: Integer
The following values can be specified:

- `1` to `65535`

Default value: `7676`

`--mquser` *mq-user*
Specifies the user name for the JMS service.

Type: String

The following values can be specified:

- *Specify user name*

Default value: `admin`

`--mqpassword` *`mq-password`*

Specifies the password for the JMS service.

Type: String

The following values can be specified:

- *Specify user password*

Default value: `admin`

`--target` *`target`*

Creates a JMS host only for the specified target.

Type: String

The following values can be specified:

- `server`
  Creates a JMS host for the default server instance. This is the default value.
- *configuration-name*
  Creates a JMS host in the specified configuration.
- *cluster-name*
  Creates a JMS host for every server instance in the specified cluster.
- *instance-name*
  Creates a JMS host for every server instance in the specified cluster.

Default value: `server`

`--force={false|true}`

Specifies whether the subcommand overwrites the existing JMS host of the same name.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`jms_host_name`

Specifies a unique identifier for the JMS host to be created.

Type: String

The following values can be specified:

- *Unique name for JMS host*

Default value: N/A

Examples

The following example creates a JMS host named `MyNewHost` on the system pigeon on the server instance `instance1`.

```
asadmin create-jms-host --target instance1 --mqhost pigeon.example.com --mqport 7677
--mquser admin --mqpassword admin MyNewHost
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.19.2 create-jms-resource

Creates a JMS resource.

Synopsis

```
asadmin [asadmin-options] create-jms-resource [--help] --restype type
    [--target target] [--enabled={true|false}]
    [--property name=value[:name=value]...] [--force={false|true}]
    jndi_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-jms-resource` subcommand creates a Java Message Service (JMS) connection factory resource or a JMS destination resource.

This subcommand is supported in remote mode only.

To specify the `addresslist` property (in the format `host:mqport,host2:mqport,host3:mqport`) for the `asadmin create-jms-resource` command, escape the `:` character using the characters `\\`.

For example, `host1\\:mqport,host2\\:mqport,host3\\:mpqport`.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`
Displays the help text for the subcommand.

`--restype` *type*
Specifies the type of JMS resource.
Type: String
The following values can be specified:

- `javax.jms.Topic`

- `javax.jms.Queue`
- `javax.jms.ConnectionFactory`
- `javax.jms.TopicConnectionFactory`
- `javax.jms.QueueConnectionFactory`

Default value: N/A

`--target` *`target`*

Creates a JMS resource only for the specified target.

The resource is always created for the domain as a whole but the `resource-ref` element for the resource is only created for the specified `--target`. This means that although the resource is defined at the domain level, it is only active at the specified `--target`.

Type: String

The following values can be specified:

- `server`
  Creates a JMS resource for the default server instance. This is the default value.
- `domain`
  Creates a JMS resource for the domain.
- *cluster_name*
  Creates a JMS resource for every server instance in the specified cluster.
- *instance_name*
  Creates a JMS resource for the specified server instance.

Default value: `server`

`--enabled={true|false}`

Specifies whether the resource is enabled at runtime or not.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--property` *`name=value`*`[:`*`name=value`*`]...`

Specifies the optional attribute *name*/*value* pairs for configuring the JMS resource. The specification format is `name=`*`value`*. Use a colon (`:`) to separate property names. If you specify the same property name more than once, the property value specified last is valid.

Type: String

The following values can be specified:

- `ClientId=`*`value`*
  A client ID for a connection factory that will be used by a durable subscriber.
  Type: String
  Default value: N/A
  Range Value: ID of the JMS resource
- `AddressList=`*`value`*

A comma-separated list of message queue addresses that specify the host names (and, optionally, port numbers) of a message broker instance, or instances with which the application will communicate. For example, the value could be `earth` or `earth:7677`. Specify the port number if the message broker is running on a port other than the default (`7676`). composed from the JMS hosts defined in the server's JMS service configuration. The default value is `localhost` and the default port number is `7676`. The client will attempt a connection to a broker on port `7676` of the local host.

Type: String

Default value: N/A

Range Value: Comma separated Host names and optional port numbers

- `UserName=`*`value`*

  The user name for the connection factory. The default value is guest.

  Type: String

  Default value: `guest`

  Range Value: String (User name)

- `Password=`*`value`*

  The password for the connection factory. The default value is guest.

  Type: String

  Default value: `guest`

  Range Value: String (User name)

- `ReconnectEnabled=`*`value`*

  A value of `true` indicates that the client runtime attempts to reconnect to a message server (or the list of addresses in the `AddressList`) when a connection is lost. The default value is false.

  Type: Boolean

  Default value: `false`

  Range Value: `true`/`false`

- `ReconnectAttempts=`*`value`*

  The number of attempts to connect (or reconnect) for each address in the `AddressList` before the client runtime tries 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). The default value is `6`.

  Type: Integer

  Default value: `6`

  Range Value: `-1` to `2147483647`

- `ReconnectInterval=`*`value`*

  The interval in milliseconds between reconnect attempts. This applies to attempts on each address in the `AddressList` and for successive addresses in the list. If the interval is too short, the broker does not have time to recover. If it is too long, the reconnect might represent an unacceptable delay. The default value is `30000` milliseconds.

  Type: Long

  Default value: `30000` (milli seconds)

  Range Value: `1` to `9223372036854775807L`

- `AddressListIterations=`*`value`*

  The number of times the client runtime iterates through the `AddressList` in an effort to establish (or re-establish) a connection). A value of `-1` indicates that the number of attempts is unlimited. The default value is

-1. Additionally, you can specify connector-connection-pool attributes as connector resource properties.

Type: Integer

Default value: -1

Range Value: -1 to 2147483647

You can specify the following properties for a destination resource:

- Name=*value*

  The name of the physical destination to which the resource will refer. The physical destination is created automatically when you run an application that uses the destination resource. You can also create a physical destination with the create-jmsdest subcommand. If you do not specify this property, the JMS service creates a physical destination with the same name as the destination resource (replacing any forward slash in the JNDI name with an underscore).

  Type: String

  Default value: JNDI name of the destination resource with _ replacing /.

  Range Value: String (Name of the Physical destination)

- Description=*value*

  A description of the physical destination.

  Type: String

  Default value: N/A

  Range Value: String (Description of the physical destination).

Default value: N/A

Specify the following properties for a connection factory resource:

--force={false|true}

Specifies whether the subcommand overwrites the existing JMS resource of the same name.

Type: Boolean

The following values can be specified:

- true
- false

Default value: false

jndi_name

Specifies the JNDI name of the JMS resource to be created.

Type: String

The following values can be specified:

- *JNDI name*

Default value: N/A

Examples

The following example creates a connection factory resource of the type javax.jms.ConnectionFactory whose JNDI name is jms/DurableConnectionFactory on the server instance instance1. The ClientId property sets a client ID on the connection factory so that it can be used for durable subscriptions. The JNDI name for a JMS resource customarily includes the jms/ naming subcontext.

```
asadmin create-jms-resource --restype javax.jms.ConnectionFactory --target instance1
--property ClientId=MyID jms/DurableConnectionFactory
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.19.3 create-jmsdest

Creates a JMS physical destination.

Synopsis

```
asadmin [asadmin-options] create-jmsdest [--help] --desttype dest_type
    [--property name=value[:name=value]...] [--target target]
    [--force={false|true}] dest_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-jmsdest` subcommand creates a Java Message Service (JMS) physical destination.

This subcommand is supported in remote mode only.

Typically, you use the `create-jms-resource` subcommand to create a JMS destination resource that has a `Name` property that specifies the physical destination. The physical destination is created automatically when you run an application that uses the destination resource. Use the `create-jmsdest` subcommand if you want to create a physical destination with non-default property settings.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--desttype` *dest_type*

Specifies the type of the JMS destination.

Type: String

The following values can be specified:

- `topic`
- `queue`

Default value: N/A

`--property` *`name=value`*`[:`*`name=value`*`]...`

Specifies the optional attribute `name`/`value` pairs required to configure the physical destination. The specification format is `name=`*`value`*. Use a colon (`:`) to separate property names. If you specify the same property name more than once, the property value specified last is valid.

To modify the value of these properties, you can use them *`Application Server installation directory`*`/javaee/glassfish/mq/bin/imqcmd` command.

Type: String

The following values can be specified:

- `MaxNumMsgs=`*`value`*

  Allows you to select the maximum number of unconsumed messages permitted for the destination. A value of `-1` denotes an unlimited number of messages. The default value is `-1`. For the dead message queue, the default value is `1000`. If the `limitBehavior` property is set to `FLOW_CONTROL`, it is possible for the specified message limit to be exceeded because the broker cannot react quickly enough to stop the flow of incoming messages. In such cases, the value specified for `maxNumMsgs` serves as merely a hint for the broker rather than a strictly enforced limit.

  Type: Integer

  Default value: `1000` for dead message queue and `-1` for others.

  Range Value: `-1` to `2147483647`

- `MaxBytesPerMsg=`*`value`*

  Displays the maximum size (in bytes) of a single message. The rejection of a persistent message is reported to the producing client with an exception; no notification is sent for non-persistent messages. The value may be expressed in bytes, kilobytes, or megabytes, using the following suffixes:

  b: Bytes

  k: Kilobytes (1024 bytes)

  m: Megabytes (1024 x 1024 = 1,048,576 bytes)

  A value with no suffix is expressed in bytes; a value of `-1` denotes an unlimited message size. The default value is `-1`.

  Type: Integer

  Default value: `-1`

  Range Value: `-1` to `2147483647`

- `MaxTotalMsgBytes=`*`value`*

  Displays the maximum total memory (in bytes) for unconsumed messages. The default value is `-1`. The syntax is the same as for `maxBytesPerMsg`. For the dead message queue, the default value is `10 m`.

  Type: Integer

  Default value: `10m` for dead message queue, `-1` for others.

  Range Value: `-1` to `2147483647`

- `LimitBehavior=`*`value`*

  Displays the behavior of the message queue broker when the `memory-limit` threshold is reached.

  The following values can be specified:

  `REJECT_NEWEST`:

  Rejects the newest messages and notifies the producing client with an exception only if the message is persistent. This is the default value.

  `FLOW_CONTROL`:

Slows the rate at which message producers send messages. If the `limitBehavior` property is set to `FLOW_CONTROL`, it is possible for the specified message limit to be exceeded the `maxNumMsgs` because the broker cannot react quickly enough to stop the flow of incoming messages. In such cases, the value specified for `maxNumMsgs` serves as merely a hint for the broker rather than a strictly enforced limit.

`REMOVE_OLDEST`:

Removes the oldest messages.

`REMOVE_LOW_PRIORITY`:

Removes the lowest-priority messages according to age, with no notification to the producing client. If the value is `REMOVE_OLDEST` or `REMOVE_LOW_PRIORITY` and the `useDMQ` property is set to true, the excess messages are moved to the dead message queue. For the dead message queue itself, the default limit behavior is `REMOVE_OLDEST`, and the value cannot be set to `FLOW_CONTROL`.

Type: String

Default value: `REJECT_NEWEST`

Range: `String`

- `MaxNumProducers=`*`value`*

  Displays the maximum number of message producers for a destination. When this limit is reached, no new producers can be created. A value of `-1` denotes an unlimited number of producers. The default value is `100`. This property does not apply to the dead message queue.

  Type: Integer

  Default value: `100`

  Range Value: `-1 and 1` to `2147483647`

- `ConsumerFlowLimit=`*`value`*

  Displays the maximum number of messages that can be delivered to a consumer in a single batch. A value of `-1` denotes an unlimited number of messages. The default value is `1000`. The client runtime can override this limit by specifying a lower value on the connection factory object. In load-balanced queue delivery, this is the initial number of queued messages routed to active consumers before load balancing begins.

  Type: Integer

  Default value: `1000`

  Range Value: `-1` to `2147483647`

- `UseDMQ=`*`value`*

  If set to `true`, the dead messages go to the dead message queue. If set to `false`, the dead messages are discarded. The default value is `true`.

  Type: Boolean

  Default value: `true`

  Range Value:

  The following values can be specified:

  `true`

  `false`

- `ValidateXMLSchemaEnabled=`*`value`*

  If set to `true`, XML schema validation is enabled for the destination. The default value is `false`. When XML validation is enabled, the `Message Queue` client runtime will attempt to validate an XML message against the specified XSDs (or against the DTD, if no XSD is specified) before sending it to the broker. If the specified schema cannot be located or the message cannot be validated, the message is not sent, and an exception is thrown. This property should be set when a destination is inactive, that is, when it has no consumers or producers and when there are no messages in the destination. Otherwise the producer must reconnect.

Type: Boolean

Default value: `false`

Range Value:

The following values can be specified:

`true`

`false`

- `XMLSchemaURIList=`*`value`*

  Comprises a space-separated list of XML schema document (XSD) URI strings. The URLs point to the location of one or more XSDs to use for XML schema validation, if `validateXMLSchemaEnabled` is set to `true`. The default value is `null`. Use double quotes around this value if multiple URIs are specified, as in the following example:

  ```
  http://foo/flap.xsd http://test.com/test.xsd
  ```

  If this property is not set or is `null` and the XML validation is enabled. The XML validation is performed using a DTD specified in the XML document. If an XSD is changed as a result of changing application requirements, all client applications that produce XML messages based on the changed XSD must reconnect to the broker.

  Type: String

  Default value: `null`

  Range Value: *String (list of space-separated XSD URI strings)*

Default value: N/A

`--target` *`target`*

Creates the physical destination only for the specified target. Although the `create-jmsdest` subcommand is related to resources, a physical destination is created using the JMS Service (JMS Broker), which is part of the configuration.

Type: String

The following values can be specified:

- `server`

  Creates the physical destination for the default server instance. This is the default value.

- *configuration-name*

  Creates the physical destination in the specified configuration.

- *cluster-name*

  Creates the physical destination for every server instance in the specified cluster.

- *instance-name*

  Creates the physical destination for the specified server instance.

Default value: `server`

`--force={false|true}`

Specifies whether the subcommand overwrites the existing JMS physical destination of the same name.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

dest_name

Displays a unique identifier for the JMS destination to be created.

Type: String

The following values can be specified:

- *Name of the physical destination*

Default value: N/A

Examples

The following example creates a JMS physical queue named `PhysicalQueue` with non-default property values on the server instance `instance1`.

```
asadmin create-jmsdest --target instance1 --desttype queue --property
maxNumMsgs=1000:maxBytesPerMsg=5k PhysicalQueue
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.19.4 delete-jms-host

Removes a JMS host.

Synopsis

```
asadmin [asadmin-options] delete-jms-host [--help] [--target target]
    jms_host_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-jms-host` subcommand is supported in remote mode only. Remote asadmin subcommands require a running domain administration server (DAS).

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Deletes the JMS host, only from the specified target.

Type: String

The following values can be specified:

- `server`
  Deletes the JMS host from the default server instance. This is the default value.
- *configuration-name*
  Deletes the JMS host from the specified configuration.
- *cluster-name*
  Deletes the JMS host from every server instance in the specified cluster.
- *instance_name*
  Deletes the JMS host from the specified server instance.

Default value: `server`

jms_host_name

Specifies the name of the host to be deleted.

Type: String

The following values can be specified:

- *The name of the host*

Default value: N/A

Examples

The following example deletes the JMS host named `MyNewHost` on the server instance `instance1`.

```
asadmin delete-jms-host --target instance1 MyNewHost
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.19.5 delete-jms-resource

Removes a JMS resource.

Synopsis

```
asadmin [asadmin-options] delete-jms-resource [--help]
    [--target target]
    jndi_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The delete-jms-resource subcommand is supported in remote mode only. Remote asadmin subcommands require a running domain administration server (DAS).

Precondition

Domain Administration Server (DAS) is running.

Arguments

--help|-?

Displays the help text for the subcommand.

--target *target*

Deletes the JMS resource only from the specified target.

Resources are always created for a domain as a whole but are only active for targets for which a resource-ref has been created using the --target option when the resource was created. This means that deleting a resource only deletes the resource-ref element for the specified --target, and does not delete the resource from the domain as a whole unless domain is specified as the --target for the deletion.

Type: String

The following values can be specified:

- server

  Deletes the JMS resource from the default server instance. This is the default value.
- domain

  Deletes the JMS resource from the domain.
- *cluster-name*

  Deletes the JMS resource from every server instance in the specified cluster.
- *instance-name*

  Deletes the JMS resource from the specified server instance.

Default value: server

jndi_name

Specifies the JNDI name of the JMS resource to be deleted.

Type: String

The following values can be specified:

- *The name of the resource*

Default value: N/A

Examples

The following subcommand deletes the JMS destination resource named jms/MyQueue on the server instance instance1.

```
asadmin delete-jms-resource --target instance1 jms/MyQueue
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.19.6 delete-jmsdest

Removes a JMS physical destination.

Synopsis

```
asadmin [asadmin-options] delete-jmsdest [--help]
    --desttype type
    [--target target]
    dest_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-jmsdest` subcommand is supported in remote mode only. Remote `asadmin` subcommands require a running Domain Administration Server (DAS).

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--desttype` *dest_type*

Specifies the type of the JMS destination.

Type: String

The following values can be specified:

- `topic`
- `queue`

Default value: N/A

`--target` *target*

Deletes the physical destination only from the specified target.

Type: String

The following values can be specified:

- `server`
  Deletes the physical destination from the default `server` instance. This is the default value.
- *configuration-name*
  Deletes the physical destination from the specified configuration.
- *cluster-name*
  Deletes the physical destination from every server instance in the specified cluster.
- *instance-name*
  Creates the physical destination from the specified server instance.

Default value: `server`

dest_name

Specifies the unique identifier of the JMS destination to be deleted.

Type: String

The following values can be specified:

- *The name of the JMS destination*

Default value: N/A

Examples

The following subcommand deletes the queue named `PhysicalQueue` on the server instance `instance1`.

```
asadmin delete-jmsdest --target instance1 --desttype queue PhysicalQueue
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.19.7 flush-jmsdest

Purges messages in a JMS destination.

Synopsis

```
asadmin [asadmin-options] flush-jmsdest [--help]
    --desttype {topic|queue} [--target target] destname
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `flush-jmsdest` subcommand purges the messages from a physical destination in the server's Java Message Service (JMS) configuration. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--desttype {topic|queue}`

Specifies the type of physical destination from which you want to purge messages.

Type: String

The following values can be specified:

- `topic`
- `queue`

Default value: N/A

`--target` *`target`*

Purges messages from the physical destination only for the specified target.

Type: String

The following values can be specified:

- `server`
  Purges messages from the physical destination for the default `server` instance. This is the default value.
- *configuration-name*
  Purges messages from the physical destination in the specified configuration.
- *cluster-name*
  Purges messages from the physical destination for every server instance in the specified cluster.
- *instance-name*
  Purges messages from the physical destination for the specified server instance.

Default value: `server`

`dest_name`

Specifies the unique identifier of the JMS destination to be purged.

Type: String

The following values can be specified:

- *Name of the physical destination*

Default value: N/A

Examples

The following subcommand purges messages from the queue named `PhysicalQueue` on the server instance `instance1`.

```
asadmin flush-jmsdest --target instance1 --desttype queue PhysicalQueue
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.19.8 list-jms-hosts

Lists the existing JMS hosts.

Synopsis

```
asadmin [asadmin-options] list-jms-hosts [--help] [--target target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-jms-hosts` subcommand of asadmin lists the existing Java Message Service (JMS) hosts for the JMS service. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`
Displays the help text for the subcommand.

`--target` *target*
Restricts the listing to JMS hosts for a specified target.
Type: String
The following values can be specified:

- `server`
  Lists the JMS hosts for the default `server` instance. This is the default value.
- *configuration-name*
  Lists the JMS hosts for the specified configuration.
- *cluster-name*
  Lists the JMS hosts for all the server instances in the specified cluster.
- *instance-name*
  Lists the JMS hosts for the specified server instance.

Default value: `server`

Examples

The following example lists the JMS hosts for the JMS service on the server instance `instance1`.

```
asadmin list-jms-hosts --target instance1
```

Exit Status

Exit Status	Explanation
`0`	subcommand executed successfully.
`1`	error in executing the subcommand.

2.19.9 list-jms-resources

Lists the JMS resources.

Synopsis

```
asadmin [asadmin-options] list-jms-resources [--help]
    [--restype type] [target]
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `list-jms-resources` subcommand of asadmin lists the existing Java Message Service (JMS) resources (destination and connection factory resources). This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--restype` *type*

Specifies the JMS resource type to be listed.

Type: String

The following values can be specified:

- `javax.jms.Topic`
- `javax.jms.Queue`
- `javax.jms.ConnectionFactory`

- `javax.jms.TopicConnectionFactory`
- `javax.jms.QueueConnectionFactory`

Default value: N/A

target

Restricts the listing to resources for a specified target.

Type: String

The following values can be specified:

- `server`
  Lists the resources for the default `server` instance. This is the default value.
- `domain`
  Lists the resources for the domain.
- *cluster-name*
  Lists the resources for every server instance in the specified cluster.
- *instance-name*
  Lists the resources for the specified server instance.

Default value: `server`

Examples

The following example lists all `javax.jms.ConnectionFactory` resources.

```
asadmin list-jms-resources --restype javax.jms.ConnectionFactory
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.19.10 list-jmsdest

Lists the existing JMS physical destinations.

Synopsis

```
asadmin [asadmin-options] list-jmsdest [--help]
    [--desttype type]
    target
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-jmsdest` subcommand of asadmin lists the Java Message Service (JMS) physical destinations.

This subcommand is supported in remote mode only. Remote `asadmin` subcommands require a running Domain Administration Server (DAS).

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--desttype` *type*

Specifies the type of JMS destination to be listed. Valid values are topic and queue.

Type: String

The following values can be specified:

- `topic`
- `queue`

Default value: `queue`

target

Restricts the listing to physical destinations for a specified target.

Type: String

The following values can be specified:

- `server`
  Lists the physical destinations for the default server instance. This is the default value.
- *configuration-name*
  Lists the physical destinations in the specified configuration.
- *cluster-name*
  Lists the physical destinations for every server instance in the specified cluster.
- *instance-name*
  Lists the physical destinations for the specified server instance.

Default value: `server`

Examples

The following example lists all physical topics.

```
asadmin list-jmsdest --desttype topic
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.

Exit Status	Explanation
1	error in executing the subcommand.

2.20 Commands used for EIS connection administration

This section describes the syntax and functionality of the commands used for EIS connection administration.

2.20.1 create-admin-object

Creates an administered object with the specified JNDI name for a resource adapter.

Synopsis

```
asadmin [asadmin-options] create-admin-object [--help]
    [--target target] --restype restype [--classname classname]
    --raname raname [--enabled={true|false}]
    [--description description] [--property name=value[:name=value]...]
    jndi_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-admin-object` subcommand creates the administered object with the specified JNDI name and the interface definition for a resource adapter.

This command is supported in remote mode only.

Precondition

- DAS has to be in a running state.
- The resource adapter has to be deployed before running the `create-admin-object` subcommand.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target on which you are creating the administered object.

The resource is always created for the domain as a whole, but the `resource-ref` for the resource is created only for the specified `--target`. This means that although the resource is defined at the domain level, it is only available at the specified target level. Use the `create-resource-ref` subcommand to refer to the resource in multiple targets if needed.

Type: String

The following values can be specified:

- `server`
  Creates the administered object for the default server instance `server`. This is the default value.
- *configuration_name*
  Creates the administered object for the named configuration.

- *cluster_name*

  Creates the administered object for every server instance in the cluster.

- *instance_name*

  Creates the administered object for a particular server instance.

Default value: `server`

`--restype` *`restype`*

Specifies the interface definition for the administered object. The resource type has to be an interface definition that is specified in the `ra.xml` file of the resource adapter.

Type: String

The following values can be specified:

- *Type of the resource, the resource type has to be an interface definition that is specified in the ra.xml file of the resource adapter*

Default value: N/A

`--classname` *`classname`*

Specifies the class name of the administered object. Required if multiple administered objects use the same interface definition.

Type: String

The following values can be specified:

- *Class name of administered object*

Default value: N/A

`--raname` *`raname`*

Specifies the name of the resource adapter associated with this administered object.

Type: String

The following values can be specified:

- *Name of the resource adapter*

Default value: N/A

`--enabled={true|false}`

Specifies if this object is enabled. Default is true.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--description` *`description`*

Displays the text string describing the administered object.

Type: String

The following values can be specified:

- *Text describing the administered object to be created*

Default value: N/A

`--property` *`name=value`*`[:`*`name=value`*`]...`

Describes the name or value pairs for configuring the resource. Dependent on the resource adapter.

After creating the admin object, some properties require a server restart. If required, restart the server.

The specification format is *name*=*value*. Use a colon (`:`) to separate property names. If you specify the same property name more than once, the property value specified last is valid.

Type: String

The following values can be specified:

- *name-value pairs of the properties supported by the resource adapter*

Default value: N/A

jndi_name

Specifies the JNDI name of the administered object to be created.

Type: String

The following values can be specified:

- *JNDI name of the administered object to be created*

Default value: N/A

Examples

In the following example, `jmsra` is a system resource adapter with the `javax.jms.Queue` and `javax.jms.Topic` admin object interfaces. This `jmsra` system resource adapter is created on the server instance `instance1`.

```
asadmin create-admin-object --target instance1 --restype javax.jms.Queue
--raname jmsra --description "sample administered object"
--property Name=sample_jmsqueue jms/samplequeue
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.20.2 create-connector-connection-pool

Adds a connection pool with the specified connection pool name.

Synopsis

```
asadmin [asadmin-options] create-connector-connection-pool [--help]
    --raname raname
    --connectiondefinition connectiondefinitionname
    [--steadypoolsize steadypoolsize]
    [--maxpoolsize maxpoolsize]
    [--maxwait maxwait]
    [--poolresize poolresize]
    [--idletimeout idletimeout]
    [--isconnectvalidatereq={false|true}]
    [--leaktimeout=timeout]
    [--leakreclaim={false|true}]
    [--creationretryattempts=attempts]
    [--creationretryinterval=interval]
```

```
    [--lazyconnectionenlistment={false|true}]
    [--lazyconnectionassociation={false|true}]
    [--matchconnections={true|false}]
    [--maxconnectionusagecount=count]
    [--validateatmostonceperiod=interval]
    [--transactionsupport transactionsupport]
    [--description description]
    [--ping {false|true}]
    [--pooling {true|false}]
    [--property name=value[:name=value]...]
    poolname
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-connector-connection-pool` subcommand of asadmin, defines a pool of connections to an Enterprise Information System (EIS). The named pool can be referred by multiple connector resources. Each defined pool is initiated at server startup and is populated when accessed for the first time. If two or more connector resources point to the same connector connection pool, they use the same pool of connections at run time. There can be more than one pool for a connection definition in a single resource adapter. A connector connection pool with authentication can be created either by specifying, the connection information in the XML descriptor file or by using the `--property` option to specify user, password or other connection information.

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--connectiondefinition` *connectiondefinitionname*

Specifies the name of the connection definition.

Type: String

The following values can be specified:

- *Name of connection definition*

Default value: N/A

`--creationretryattempts=`*attempts*

Specifies the maximum number of times that the server retries to create a connection if the initial attempt fails.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `0`

`--creationretryinterval=`*interval*

Specifies the interval, in seconds, between successive attempts to create a connection.

If the value for the `--creationretryattempts` option is `0`, the `--creationretryinterval` option is ignored.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `10`

`--description` *description*

Displays the text providing descriptive details about the connector connection pool.

Type: String

The following values can be specified:

- *Description in text*

Default value: N/A

`--idletimeout` *idletimeout*

Specifies the maximum time (in seconds) that a connection can remain idle in the pool.

Type: Integer

The following values can be specified:

- `1` to `2147483647`

Default value: `300`

`--isconnectvalidatereq={false|true}`

Confirms whether the connections are usable, if the value is set to `true`.

The connection failure detection at a regular interval is always enabled, regardless of the specified value.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--lazyconnectionenlistment={false|true}`

Specifies whether a resource to a transaction is enlisted only when a method actually uses the resource.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--lazyconnectionassociation={false|true}`

Specifies whether a physical connection should be associated with the logical connection only when the physical connection is used.

Type: Boolean

The following values can be specified:

- `true`

A physical connection is associated with the logical connection only when the physical connection is used, and disassociated when the transaction is completed. The `--lazyconnectionenlistment` option must also be set to `true`.

- `false`

  A physical connection is associated with the logical connection even before the physical connection is used, and is not disassociated when the transaction is completed. This is the default value.

Default value: `false`

`--leakreclaim={false|true}`

Specifies whether leaked connections are restored to the connection pool after leak connection tracing is complete.

Type: Boolean

The following values can be specified:

- `true`

  Leaked connections are restored to the connection pool.

- `false`

  Leaked connections are not restored to the connection pool. This is the default value.

Default value: `false`

`--leaktimeout=`*`timeout`*

Specifies the amount of time, in seconds, for which connection leaks in a connection pool are to be traced.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `0`

`--matchconnections={false|true}`

Specifies whether a connection that is selected from the pool should be matched with the resource adaptor.

If all connections in the pool are identical, matching between connections and resource adapters is not required.

Type: Boolean

The following values can be specified:

- `true`

  A connection should be matched with the resource adaptor. This is the default value.

- `false`

  A connection should not be matched with the resource adaptor.

Default value: `true`

`--maxconnectionusagecount=`*`count`*

Specifies the maximum number of times that a connection can be reused.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `0`

`--maxpoolsize` *`maxpoolsize`*

Specifies the maximum number of connections that can be created to satisfy client requests.

Type: Integer

The following values can be specified:

- 1 to 2147483647

Default value: 24

--maxwait *maxwait*

Specifies the time, in milliseconds, that a caller must wait before a connection is created, if a connection is not available.

If 0 is specified, a caller waits indefinitely.

Type: Integer

The following values can be specified:

- 0 to 2147483647

Default value: 60000

--ping {false|true}

Contacts a pool with this attribute (set to true), during creation.

Type: Boolean

The following values can be specified:

- true
- false

Default value: false

--pooling {true|false}

Disables connection pooling, when set to false.

Type: Boolean

The following values can be specified:

- true
- false

Default value: true

--poolresize *poolresize*

Specifies the quantity by which, the pool will scale up or scale down the number of connections.

Type: Integer

The following values can be specified:

- 1 to 2147483647

Default value: 2

--property *name*=*value*[:*name*=*value*]...

Optional attribute name/value pairs for configuring the pool. The specification format is name=*value*. Use a colon (:) to separate property names. If you specify the same property name more than once, the property value specified last is valid.

This option can be used to overwrite the setting value retained by the ManagedConnectionFactory class based on the JavaBeans specifications. Specify a property name according to the JavaBeans specifications that corresponds to setter of the ManagedConnectionFactory class specified for the managedconnectionfactory-class element of the ra.xml file. If the same property is set to this option and the config-property element of the ra.xml file, this option takes priority.

Type: String

`--raname` *raname*

Specifies the name of the resource adapter.

Type: String

The following values can be specified:

- *resource adapter name*

Default value: N/A

`--steadypoolsize` *steadypoolsize*

Specifies the minimum and initial number of connections maintained in the pool.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `24`

`--transactionsupport` *transactionsupport*

Confirms the level of transaction support that this pool will have.

Type: String

The following values can be specified:

- `XATransaction`
- `LocalTransaction`
- `NoTransaction`

Default value: N/A

`--validateatmostonceperiod=`*interval*

Specifies the time interval, in seconds, between successive requests to validate a connection at most once.

If `0` is specified, the connection is always validated.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `0`

poolname

Specifies the name of the connection pool to be created.

Type: String

The following values can be specified:

- *Name of pool*

Default value: N/A

Examples

The following example creates a connector connection pool.

```
asadmin create-connector-connection-pool --raname jmsra
--connectiondefinition javax.jms.QueueConnectionFactory --steadypoolsize 20
--maxpoolsize 100 --poolresize 2 --maxwait 60000 jms/qConnPool
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.20.3 create-connector-resource

Registers the connector resource with the specified JNDI name.

Synopsis

```
asadmin [asadmin-options] create-connector-resource [--help]
    --poolname connectorConnectionPoolName [--enabled={true|false}]
    [--description description] [--objecttype objecttype]
    [--target target] jndi_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-connector-resource` subcommand registers the connector resource with the specified JNDI name.

This command is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--poolname` *connectorConnectionPoolName*

Specifies the name of the connection pool.

Type: String

The following values can be specified:

- *Specify the name of the connection pool*

Default value: N/A

`--enabled={true|false}`

Specifies whether the resource is enabled at runtime.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--objecttype` *`objecttype`*

Defines the type of connector resource.

Type: String

The following values can be specified:

- `system-all`

  A system resource for all server instances and the Domain Administration Server (DAS).

- `system-admin`

  A system resource only for the DAS.

- `system-instance`

  A system resource for all server instances only.

- `user`

  A user resource.

Default value: `user`

`--description` *`description`*

Displays the text providing details about the connector resource.

Type: String

The following values can be specified:

- *Description in text*

Default value: N/A

`--target` *`target`*

Specifies the ending location of the connector resources.

The resource is always created for the domain as a whole, but the `resource-ref` for the resource is only created for the specified `--target`. This means that although the resource is defined at the domain level, it is only available at the specified target level. Use the `create-resource-ref` subcommand to refer to the resource in multiple targets if required.

Type: String

The following values can be specified:

- `server`

  Creates the connector resource in the default server instance. This is the default value.

- `domain`

  Creates the connector resource in the domain.

- *cluster_name*

  Creates the connector resource in every server instance in the cluster.

- *instance_name*

  Creates the connector resource in the specified server instance.

Default value: `server`

`jndi_name`

Specifies the JNDI name of this connector resource.

Type: String

The following values can be specified:

- *JNDI name*

Default value: N/A

Examples

The following example creates a connector resource on the server instance `instance1`.

```
asadmin create-connector-resource  --target instance1 --poolname jms/qConnPool
--description "sample connector resource" jms/qConnFactory
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.20.4 create-connector-security-map

Creates a security map for the specified connector connection pool.

Synopsis

```
asadmin [asadmin-options] create-connector-security-map [--help]
    --poolname connector_connection_pool_name
    [--principals principal-name1[,principal-name2]...]
    [--usergroups user-group1[,user-group2]...]
    --mappedusername user-name mapname
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-connector-security-map` subcommand creates a security map for the specified connector connection pool. If the security map is not available, a new one is created. This command can also map the caller identity of the application (principal or user group) to a suitable Enterprise Information System (EIS) principal, in container-managed authentication scenarios. One or more named security maps can be associated with a connector connection pool. The connector security map configuration supports the use of the wild card asterisk (`*`) to indicate all users or all user groups. To specify the EIS password,`AS_ADMIN_MAPPEDPASSWORD` entry can be added to the password file, then specify the file by using the `--passwordfile asadmin` utility option. This command is supported in remote mode only.

For this command to succeed, a connector connection pool using the `create-connector-connection-pool` subcommand has to be created.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--poolname` *`connector_connection_pool_name`*

Specifies the name of the connector connection pool to which the security map belongs.

Type: String

The following values can be specified:

- *Pool name*

Default value: N/A

`--principals` *`principal-name1`*`[,`*`principal-name2`*`]...`

Specifies a list of backend EIS principals.

- Use either the `--principals` or `--usergroups` options, but not both in the same command.
- More than one principal can be specified using a comma separated list.

Type: String

The following values can be specified:

- *Backend EIS principle name*

Default value: N/A

`--usergroups` *`user-group1`*`[,`*`user-group2`*`]...`

Specifies a list of backend EIS user group.

- Use either the `--principals` or `--usergroups` options, but not both in the same command.
- More than one user groups can be specified using a comma separated list.

Type: String

The following values can be specified:

- *EIS user group*

Default value: N/A

`--mappedusername` *`user-name`*

Specifies the EIS username.

Type: String

The following values can be specified:

- *User name*

Default value: N/A

`mapname`

Specifies the name of the security map to be created.

Type: String

The following values can be specified:

- *Name of map*

Default value: N/A

Examples

The following example creates a connector security map.

```
asadmin create-connector-security-map --poolname connector-pool1
--principals principal1,principal2 --mappedusername backend-username
securityMap1
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.20.5 create-connector-work-security-map

Creates a work security map for the specified resource adapter.

Synopsis

```
asadmin [asadmin-options] create-connector-work-security-map [--help]
    --raname raname [--principalsmap eis-principal1=principal_name1
    [,eis-principal2=principal_name2]...|
    --groupsmap eis-group1=server-group1[,eis-group2=server-group2]...]
    mapname
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-connector-work-security-map` subcommand maps the caller identity of the work submitted by the resource adapter EIS principal or EIS user group to a suitable principal or user group in Java EE Server security domain.

One or more work security maps can be associated with a resource adapter. The connector work security map configuration supports the use of the wild card asterisk (`*`) to indicate all users or all user groups.

The Enterprise Information System (EIS) can be any system that holds the data of an organization. It can be a mainframe, a messaging system, a database system, or an application.

This command is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--groupsmap` *`eis-group1=server-group1`*`[,`*`eis-group2=server-group2`*`]...`

Specifies a map of the backend EIS user group to Java EE Server user group.

Use a comma-separated list to specify more than one mapping. Use either the `--principalsmap` option or the `--groupsmap` option, but not both.

Type: String

The following values can be specified:

- *Name of map of backend EIS*

Default value: N/A

`--principalsmap` *`eis-principal1=principal_name1`*`[,`*`eis-principal2=principal_name2`*`]...`

Specifies a map of the backend EIS principal to the Java EE Server principal.

Use a comma-separated list to specify more than one mapping. Use either the `--principalsmap` option or the `--groupsmap` option, but not both.

Type: String

The following values can be specified:

- *Principle map name*

Default value: N/A

`--raname` *`raname`*

Indicates the connector module name that is the name of the resource adapter.

Type: String

The following values can be specified:

- *Connector module name*

Default value: N/A

`mapname`

Specifies the name of the work security map to be created.

Type: String

The following values can be specified:

- *Name of the map*

Default value: N/A

Examples

The following example creates a connector work security map.

```
asadmin create-connector-work-security-map --raname my-resource-adapter
--principalsmap eis-principal-1=server-principal-1,eis-principal-2
=server-principal-2,eis-principal-3=server-principal-1
workSecurityMap1
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.20.6 create-resource-adapter-config

Creates the configuration information for the connector module.

Synopsis

```
asadmin [asadmin-options] create-resource-adapter-config [--help]
    [--objecttype object-type] raname
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-resource-adapter-config` subcommand creates configuration information for the connector module. This subcommand can be run before deploying a resource adapter, so that the configuration information is available at the time of deployment. The resource adapter configuration can also be created after the resource adapter is deployed. In this case, the resource adapter is restarted with the new configuration. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Files

For *raname* operand,

It is the value of the `resource-adapter-name` in the `domain.xml` file.

Arguments

`--help | -?`

Displays the help text for the subcommand

`--objecttype` *object-type*

Specifies the type of object.

Type: String

The following values can be specified:

- *Name of the objecttype*

Default value: `user`

raname

Indicates the connector module name. It is the value of the `resource-adapter-name` in the `domain.xml` file.

Type: String

The following values can be specified:

- *Name of connector module*

Default value: N/A

Examples

The following example creates a resource adapter configuration for `ra1`.

```
asadmin create-resource-adapter-config ra1
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.20.7 delete-admin-object

Removes the administered object with the specified JNDI name.

Synopsis

```
asadmin [asadmin-options] delete-admin-object [--help]
    [--target target] jndi_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-admin-object` subcommand removes an administered object with the specified JNDI name.

This subcommand is supported in remote mote only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the name of the targets for which the administered object is to be deleted.

Resources are always created for a domain as a whole but are only active for targets for which a `resource-ref` has been created using the `--target` option when the resource was created. This means that deleting a resource only deletes the `resource-ref` element for the specified `--target`, and does not delete the resource from the domain as a whole unless domain is specified as the `--target` for the deletion.

Type: String

The following values can be specified:

- `server`
  Deletes the administered object for the default server instance "`server`" and is the default value.
- *configuration_name*
  Deletes the administered object for the specified configuration.
- *cluster_name*
  Deletes the administered object for the specified cluster.
- *instance_name*
  Deletes the administered object for a particular server instance.

Default value: `server`

jndi_name

Specifies the JNDI name of the administered object.

Type: String

The following values can be specified:

- *JNDI name of the administered object*

Default value: N/A

Examples

The following example deletes the administered object named `jdbc/samplelqueue` on the server instance `instance1`.

```
asadmin delete-admin-object --target instance1 jdbc/samplequeue
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.20.8 delete-connector-connection-pool

Removes the specified connector connection pool.

Synopsis

```
asadmin [asadmin-options] delete-connector-connection-pool [--help]
    [--cascade={false|true}] poolname
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-connector-connection-pool` subcommand removes the specified connector connection pool.

This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--cascade={false|true}`

Specifies the property to delete the connection pool and all connector resources associated with the pool.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

poolname

Specifies the name of the connection pool.

Type: String

The following values can be specified:

- *Name of the connection pool*

Default value: N/A

Examples

The following example deletes the connector connection pool named `jdbc/qConnPool`.

```
asadmin delete-connector-connection-pool --cascade=false jdbc/qConnPool
```

Exit Status

Exit Status	Explanation
0	command executed successfully.

Exit Status	Explanation
1	error in executing the command.

2.20.9 delete-connector-resource

Removes the connector resource for the specified JNDI name.

Synopsis

```
asadmin [asadmin-options] delete-connector-resource [--help]
    [--target target] jndi_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-connector-resource` subcommand removes the connector resource with the specified JNDI name.

This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target from which you want to remove the connector resource.

Resources are always created for a domain as a whole but are only active for targets for which a `resource-ref` has been created using the `--target` option, when the resource was created. This means that deleting a resource only deletes the `resource-ref` element for the specified `--target`, and does not delete the resource from the domain as a whole unless domain is specified as the `--target` for the deletion.

Type: String

The following values can be specified:

- `server`
  Deletes the connector resource from the default server instance. This is the default value.
- `domain`
  Deletes the connector resource from the domain.
- *cluster_name*
  Deletes the connector resource from every server instance in the cluster.
- *instance_name*
  Deletes the connector resource from a specified server instance.

Default value: `server`

jndi_name

Specifies the JNDI name of the connector resource.

Type: String

The following values can be specified:

- *JNDI name of the connector resource*

Default value: N/A

Examples

The following example deletes a connector resource named `jdbc/qConnFactory` on the server instance `instance1`.

```
asadmin delete-connector-resource --target instance1 jdbc/qConnFactory
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.20.10 delete-connector-security-map

Deletes a security map for the specified connector connection pool.

Synopsis

```
asadmin [asadmin-options] delete-connector-security-map [--help]
    --poolname connector_connection_pool_name mapname
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `delete-connector-security-map` subcommand deletes a security map for the specified connector connection pool.

For this subcommand to succeed, you must have first created a connector connection pool using the `create-connector-connection-pool` subcommand.

This subcommand is supported in remote mode only.

Precondition

- DAS has to be in a running state.
- For `--poolname` option, a connector connection pool should be created.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--poolname` *connector_connection_pool_name*

Specifies the name of the connector connection pool to which the security map that is to be deleted belongs.

Type: String

The following values can be specified:

- *Name of the connector connection pool*

Default value: N/A

mapname

Specifies the name of the security map.

Type: String

The following values can be specified:

- *Name of the security map*

Default value: N/A

Examples

The following example deletes `securityMap1` for the existing connection pool named `connector-pool1`.

```
asadmin delete-connector-security-map --poolname connector-pool1 securityMap1
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.20.11 delete-connector-work-security-map

Deletes a work security map for the specified resource adapter.

Synopsis

```
asadmin [asadmin-options] delete-connector-work-security-map [--help]
    --raname raname mapname
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `delete-connector-work-security-map` subcommand deletes a security map associated with the specified resource adapter. For this subcommand to succeed, you must have first created and deployed the specified

resource adapter. The enterprise information system (EIS) is any system that holds the data of an organization. It can be a mainframe, a messaging system, a database system, or an application.

This subcommand is supported in remote mode only.

Precondition

- DAS has to be in a running state.
- For `--raname` option, the specified resource adapter should be created and deployed.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--raname` *raname*

Indicates the connector module name with which the work security map is associated.

Type: String

The following values can be specified:

- *Name of the resource adapter*

Default value: N/A

mapname

Specifies the name of the work security map.

Type: String

The following values can be specified:

- *Name of the work security map*

Default value: N/A

Examples

The following example deletes the work security map named `work_security_map_name` for the resource adapter named `ra_name`.

```
asadmin delete-connector-work-security-map
--raname ra_name work_security_map_name
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.20.12 delete-resource-adapter-config

Deletes the resource adapter configuration.

Synopsis

```
asadmin [asadmin-options] delete-resource-adapter-config [--help]
    raname
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-resource-adapter-config` subcommand deletes the configuration information for the connector module.

This command is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

raname

Specifies the connector module name.

Type: String

The following values can be specified:

- *Name of the resource adapter*

Default value: N/A

Examples

The following example deletes the configuration information for `ra1`.

```
asadmin delete-resource-adapter-config ra1
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.20.13 flush-connection-pool

Reinitializes all connections established in the specified connection pool.

Synopsis

```
asadmin [asadmin-options] flush-connection-pool [--help]
    [--appname application [--modulename module]]
    [--target target] pool_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `flush-connection-pool` subcommand resets a JDBC connection pool or a connector connection pool to its initial state. Delete the unused connections. The subcommand then recreates the initial connections for the pool, and restores the pool to its steady pool size. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--appname` *application*

Specifies the name of the application in which the application scoped resource is defined.

Type: String

The following values can be specified:

- *Application name*

Default value: N/A

`--modulename` *module*

Specifies the name of the module in which the module scoped resource is defined.

Type: String

The following values can be specified:

- *Module name*

Default value: N/A

`--target` *target*

Specifies the connection pool to be reinitialized. If you create a server instance and then specify the name of the server instance that has never started, then the command is run successfully. The target specified using the `--target` option does not depend on the pool name specified by the parameter.

Type: String

The following values can be specified:

- `server`

  Sets a Domain Administration Server as the target. `server` is the name of the Domain Administration Server.

- *instance-name*

  Sets the server instance of the specified name as the target.

Default value: `server`

pool_name

Specifies the name of the connection pool to be reinitialized.

Type: String

The following values can be specified:

- *Name of the connection pool*

Default value: N/A

Examples

The following example reinitializes the JDBC connection pool named `__TimerPool` on the server instance `instance1`.

```
asadmin flush-connection-pool --target instance1 __TimerPool
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.20.14 list-admin-objects

Gets all the administered objects.

Synopsis

```
asadmin [asadmin-options] list-admin-objects [--help]
    [target]
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `list-admin-objects` subcommand lists all the administered objects. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

target

Specifies the target for which administered objects are to be listed.

Type: String

The following values can be specified:

- `server`
  Lists the administered objects on the default `server` instance. This is the default value.
- *configuration-name*
  Lists the administered objects in the specified configuration.
- *cluster-name*
  Lists the administered objects on all server instances in the specified cluster.
- *instance-name*
  Lists the administered objects on a specified server instance.

Default value: `server`

Examples

The following example lists all the administered objects.

```
asadmin list-admin-objects
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.20.15 list-connector-connection-pools

Lists the existing connector connection pools.

Synopsis

```
asadmin [asadmin-options] list-connector-connection-pools [--help]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-connector-connection-pools` subcommand lists the connector connection pools that have been created. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

Examples

The following example lists the existing connector connection pools.

```
asadmin list-connector-connection-pools
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.20.16 list-connector-resources

Lists all connector resources.

Synopsis

```
asadmin [asadmin-options] list-connector-resources [--help]
    [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-connector-resources` subcommand lists all the connector resources. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

target

Specifies the target for which the connector resources are to be listed.

Type: String

The following values can be specified:

- `server`

Lists the connector resources on the default server instance. This is the default value.

- `domain`
  Lists the connector resources for the domain.
- *cluster-name*
  Lists the connector resources on all server instances in the specified cluster.
- *instance-name*
  Lists the connector resources on a specified server instance.

Default value: `server`

Examples

The following example lists all existing connector resources.

```
asadmin list-connector-resources
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.20.17 list-connector-security-maps

Lists the security maps belonging to the specified connector connection pool.

Synopsis

```
asadmin [asadmin-options] list-connector-security-maps [--help]
    [--securitymap securitymap]
    [--verbose={false|true}]
    pool-name
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `list-connector-security-maps` subcommand lists the security maps belonging to the specified connector connection pool. For this subcommand to succeed, you must have first created a connector connection pool using the `create-connector-connection-pool` subcommand. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--securitymap` *securitymap*

Specifies the name of the security map contained within the connector connection pool from which the identity and principals should be listed. With this option, `--verbose` is redundant.

Type: String

The following values can be specified:

- *Name of the security map*

Default value: N/A

`--verbose={false|true}`

Returns a list including the identity, principals, and security name.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

pool-name

Specifies the name of the connector connection pool for which you want to list security maps.

Type: String

The following values can be specified:

- *Name of the connector connection pool*

Default value: N/A

Examples

The following example lists the existing connector security maps for the pool named `connector-Pool1`.

```
asadmin list-connector-security-maps connector-Pool1
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.20.18 list-connector-work-security-maps

Lists the work security maps belonging to the specified resource adapter.

Synopsis

```
asadmin [asadmin-options] list-connector-work-security-maps [--help]
    [--securitymap securitymap]
    resource_adapter_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-connector-work-security-maps` subcommand lists the work security maps belonging to the specified resource adapter. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--securitymap` *securitymap*

Specifies the name of the security map contained within the resource adapter from which the identity and principals should be listed.

Type: String

The following values can be specified:

- *Name of the security map*

Default value: N/A

resource_adapter_name

Specifies the name of the resource adapter for which you want to list security maps.

Type: String

The following values can be specified:

- *Name of the resource adapter*

Default value: N/A

Examples

The following example lists the current connector work security maps for the resource adapter named `my_resource_adapter`.

```
asadmin list-connector-work-security-maps my_resource_adapter
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.

Exit Status	Explanation
1	error in executing the subcommand.

2.20.19 list-resource-adapter-configs

Lists the names of the current resource adapter configurations.

Synopsis

```
asadmin [asadmin-options] list-resource-adapter-configs [--help]
    [--raname raname] [--verbose={false|true}]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-resource-adapter-configs` subcommand of asadmin lists the configuration information in the `domain.xml` for the connector module. It lists an entry called `resource-adapter-config` in the `domain.xml` file. If the `--raname` option is specified, only the resource adapter configurations for a specific connector module are listed. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--raname` *raname*

Specifies the connector module name.

Type: String

The following values can be specified:

- *Name of the Connector module*

Default value: N/A

`--verbose={false|true}`

Lists the properties that are configured. The default value is `false`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

Examples

The following example lists the current resource adapter configurations.

```
asadmin list-resource-adapter-configs
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.20.20 ping-connection-pool

Tests a connection pool to determine if it is usable.

Synopsis

```
asadmin [asadmin-options] ping-connection-pool [--help]
    [--appname application [--modulename module]]
    [--target target] pool_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `ping-connection-pool` subcommand of `asadmin` tests if an existing JDBC or connector connection pool is usable. The `ping-connection-pool` command can target resources that are scoped to a specific application or module, as defined in the `glassfish-resources.xml` for Java EE Server domain. This subcommand is supported in remote mode only.

The `jndi-name` for `application-scoped-resources` or `module-scoped-resources` is specified using the format `java:app/jdbc/myDataSource` or `java:module/jdbc/myModuleLevelDataSource`. This naming scope is defined in the Java EE 6 Specification.

Precondition

- Domain Administration Server (DAS) is running.
- Before testing the availability of a connection pool, you must create the connection pool with authentication and ensure that the server or database is started.

Files

The `ping-connection-pool` subcommand can target resources that are scoped to a specific application or module, as defined in the `glassfish-resources.xml` for Java EE Server domain.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--appname` *`application`*

Specifies the name of the application in which the application scoped resource is defined. To reference the `jndi-name` for an application scoped resource, perform the lookup using the `java:app` prefix.

Type: String

The following values can be specified:

- *Name of the application*

Default value: N/A

`--modulename` *`module`*

Specifies the name of the module in which the module scoped resource is defined. To reference the `jndi-name` for a module scoped resource, perform the lookup using the `java:module` prefix.

Type: String

The following values can be specified:

- *Name of the module*

Default value: N/A

`--target` *`target`*

Specifies the target of this subcommand.

When you create a server instance and then specify the name of the server instance that has never been started, the command is run successfully.

The target specified by the `target` option does not depend on the pool name specified by the parameter.

Type: String

The following values can be specified:

- `server`

  Sets a DAS as the target. `server` is the name of the DAS.

- *instance-name*

  Sets the server instance of the specified name as the target.

Default value: `server`

`pool_name`

Specifies the name of the connection pool to be tested.

Type: String

The following values can be specified:

- *Name of the connection pool*

Default value: N/A

Examples

The following example tests to see if the connection pool named `DerbyPool` is usable on server instance `instance1`.

```
asadmin ping-connection-pool --target instance1 DerbyPool
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.20.21 update-connector-security-map

Modifies a security map for the specified connector connection pool.

Synopsis

```
asadmin [asadmin-options] update-connector-security-map [--help]
    --poolname connector_connection_pool_name
    [--addprincipals principal_name1[,principal_name2]...]
    [--addusergroups user_group1[,user_group2]...]
    [--removeprincipals principal_name1[,principal_name2]...]
    [--removeusergroups user_group1[,user_group2]...]
    [--mappedusername username] mapname
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `update-connector-security-map` subcommand of `asadmin` modifies a security map for the specified connector connection pool. This subcommand is supported in remote mode only.

Precondition

- A connector connection pool must exist.
- Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--poolname` *connector_connection_pool_name*

Specifies the name of the connector connection pool to which the security map that is to be updated, belongs.

Type: String

The following values can be specified:

- *Name of the connection pool*

Default value: N/A

`--addprincipals` *principal_name1*`[,`*principal_name2*`]...`

Specifies a comma-separated list of EIS-specific principals to be added.

Type: String

The following values can be specified:

- *Name of the principal*

Default value: N/A

`--addusergroups` *`user_group1`*`[,`*`user_group2`*`]...`

Specifies a comma-separated list of EIS user groups to be added.

Type: String

The following values can be specified:

- *Name of the group*

Default value: N/A

`--removeprincipals` *`principal_name1`*`[,`*`principal_name2`*`]...`

Specifies a comma-separated list of EIS-specific principals to be removed.

Type: String

The following values can be specified:

- *Name of the principal*

Default value: N/A

`--removeusergroups` *`user_group1`*`[,`*`user_group2`*`]...`

Specifies a comma-separated list of EIS user groups to be removed.

Type: String

The following values can be specified:

- *Name of the group*

Default value: N/A

`--mappedusername` *`username`*

Specifies the EIS username.

Type: String

The following values can be specified:

- *User name*

Default value: N/A

`mapname`

Specifies the name of the security map to be updated.

Type: String

The following values can be specified:

- *Name of the application module*

Default value: N/A

Examples

The following example adds principals to the existing security map named `securityMap1`.

```
asadmin update-connector-security-map --poolname connector-pool1
--addprincipals principal1,principal2 securityMap1
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.20.22 update-connector-work-security-map

Modifies a work security map for the specified resource adapter.

Synopsis

```
asadmin [asadmin-options] update-connector-work-security-map [--help]
    --raname raname
    [--addprincipals eis-principal1=server-principal1
    [, eis-principal2=server-principal2]...]
    --addgroups eis-group1=server-group1
    [, eis-group2=server-group2]...
    [--removeprincipals eis-principal1[,eis-principal2]...]
    [--removegroups eis-group1[, eis-group2]...] mapname
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `update-connector-work-security-map` subcommand of `asadmin` modifies a security map for the specified resource adapter. This subcommand is supported in remote mode only.

Precondition

- A connector connection pool must be existing.
- Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--addgroups` *eis-group1=server-group1*`[,` *eis-group2=server-group2*`]...`

Specifies a comma-separated list of EIS groups to be added.

Type: String

The following values can be specified:

- *Name of the group*

Default value: N/A

`--addprincipals` *eis-principal1=server-principal1* `[,` *eis-principal2=server-principal2*`]...`

Specifies a comma-separated list of EIS-specific principals to be added.

Type: String

The following values can be specified:

- *Name of the principal*

Default value: N/A

`--removegroups` *eis-group1*`[,` *eis-group2*`]...`

Specifies a comma-separated list of EIS groups to be removed.

Type: String

The following values can be specified:

- *Name of the group*

Default value: N/A

`--removeprincipals` *eis-principal1*`[,`*eis-principal2*`]...`

Specifies a comma-separated list of EIS-specific principals to be removed.

Type: String

The following values can be specified:

- *Name of the principal*

Default value: N/A

`--raname` *raname*

Specifies the connector module name with which the work security map is associated.

Type: String

The following values can be specified:

- *Resource adapter name*

Default value: N/A

mapname

Specifies the name of the work security map to be updated.

Type: String

The following values can be specified:

- *Name of the work security map*

Default value: N/A

Examples

The following example removes a principal from a work security map.

```
asadmin update-connector-work-security-map --raname generic-ra
--removeprincipals eis-foo generic-ra-principals-map
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.21 Commands used for thread pool administration

This section describes the syntax and functionality of the commands used for thread pool administration.

2.21.1 create-threadpool

Creates a thread pool with the specified name.

Synopsis

```
asadmin [asadmin-options] create-threadpool [--help] [--target target]
    [--maxthreadpoolsize maxthreadpoolsize]
    [--minthreadpoolsize minthreadpoolsize]
    [--idletimeout idletimeout] [--maxqueuesize maxqueuesize]
    threadpool-id
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-threadpool` subcommand creates a thread pool with the specified name. You can specify maximum and minimum number of threads in the pool, the quantity of messages, and the idle timeout of a thread. The created thread pool can be used for servicing IIOP requests and for resource adapters to service work management requests. A thread pool can be used in multiple resource adapters. This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target on which you are creating the thread pool.

Type: String

The following values can be specified:

- `server`
  Creates the thread pool for the server instance "`server`" and is the default value.
- *configuration-name*
  Creates the thread pool for the named configuration.
- *cluster-name*
  Creates the thread pool for every instance in the cluster.
- *instance-name*
  Creates the thread pool for a particular instance.

Default value: `server`

`--maxthreadpoolsize` *`maxthreadpoolsize`*

Specifies the maximum number of threads the pool can contain.

Type: Integer

The following values can be specified:

- `1` to `2147483647`

Default value: `5`

`--minthreadpoolsize` *`minthreadpoolsize`*

Specifies the minimum number of threads in the pool.

Type: Integer

The following values can be specified:

- `1` to `2147483647`

Default value: `2`

`--idletimeout` *`idletimeout`*

Specifies the amount of time in seconds after which idle threads are removed from the pool.

Type: Integer

The following values can be specified:

- `1`to `2147483647`

Default value: `900`

`--maxqueuesize` *`maxqueuesize`*

Specifies the maximum number of messages that can be queued until the threads are available to process them for a network listener or IIOP listener.

Type: Integer

The following values can be specified:

- `-1` and `1` to `2147483647`

Default value: `4096`

`threadpool-id`

Specifies an ID for the work queue.

Type: String

The following values can be specified:

- *ID of thread pool*

Default value: N/A

Examples

The following example creates a new thread pool called threadpool-l on the server instance `instance1`.

```
asadmin create-threadpool --target instance1 --maxthreadpoolsize 100
--minthreadpoolsize 20 --idletimeout 2 threadpool-1
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.21.2 delete-threadpool

Removes a thread pool.

Synopsis

```
asadmin [asadmin-options] delete-threadpool [--help]
    [--target target] threadpool-id
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The delete-threadpool subcommand removes the thread pool with the specified ID.

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

--help | -?
Displays the help text for the subcommand.

--target *target*
Specifies the target from which you are removing the thread pool.
Type: String
The following values can be specified:

- server
  Deletes the thread pool for the default server instance server and is the default value.
- *configuration-name*
  Deletes the thread pool for the named configuration.
- *cluster-name*
  Deletes the thread pool for every instance in the cluster.
- *instance-name*
  Deletes the thread pool for a particular instance.

Default value: server

threadpool-id

Specifies an ID for the work queue.

Type: String

The following values can be specified:

- *ID of thread pool*

Default value: N/A

Examples

The following example deletes `threadpool-1` on the server instance `instance1`.

```
asadmin delete-threadpool --target instance1 threadpool-1
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.21.3 list-threadpools

Lists all the thread pools.

Synopsis

```
asadmin [asadmin-options] list-threadpools [--help]
    target
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-threadpools` subcommand of asadmin lists Java EE Server thread pools. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

target

Specifies the target for which you are listing the thread pools.

Type: String

The following values can be specified:

- `server`
  Lists the thread pools for the default server instance.
- *configuration-name*
  Lists the thread pools for the named configuration.
- *cluster-name*
  Lists the thread pools for every instance in the cluster.
- *instance-name*
  Lists the thread pools for a particular instance.

Default value: `server`

Examples

The following example lists the current thread pools for the default instance server.

```
asadmin list-threadpools server
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.22 Commands used for transaction administration

This section describes the syntax and functionality of the commands used for transaction administration.

2.22.1 recover-transactions

Recovers pending transactions manually.

Synopsis

```
asadmin [asadmin-options] recover-transactions [--help]
    [--transactionlogdir transaction_log_dir]
    [--target target] server_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `recover-transactions` subcommand of `asadmin` recovers transactions after a server failure. For an installation of multiple server instances, you can run the `recover-transactions` command from a surviving server instance. To use this command, the following conditions must be met:

- Delegated transaction recovery is disabled.
- Transaction logs are stored on a shared file system or a database that is accessible to all the server instances.

For a stand-alone server, do not use this command to recover transactions after a server failure.

This subcommand is supported in remote mode only.

For a stand-alone server, the `recover-transactions` command can recover transactions only when a resource fails, but the server still running.

Precondition

Domain Administration Server (DAS) is running.

For an installation of multiple server instances, you can run the `recover-transactions` command from a surviving server instance to recover transactions after a server failure. To use this command in this way, the following conditions must be met:

- Delegated transaction recovery is disabled.
- Transaction logs are stored on a shared file system or a database that is accessible to all the server instances.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--transactionlogdir` *`transaction_log_dir`*

Specifies the location of the transaction logs for a server for which the transaction recovery is requested. This option applies only if transaction logs are stored on a shared file system.

Type: String

The following values can be specified:

- *Location of the logs*

Default value: N/A

`--target` *`target`*

Specifies the target server that performs the recovery for the server, which is specified by the *server_name* operand.

Type: String

The following values can be specified:

- *Name of the target server instance*

Default value: N/A

`server_name`

Specifies the value of this operand, which is typically server. This is for a stand-alone server. If this server is running, recovery is performed by the same server. In this situation, the `--transactionlogdir` and `--target` options must be omitted. If the server is not running, the `--target` option is required, and the `--transactionlogdir` option is also required if transaction logs are stored on a shared file system.

Type: String

The following values can be specified:

- *Name of the server instance*

Default value:

- For a stand-alone server, the value of this operand is typically `server`.
- For an installation of multiple server instances, the value of this operand is the name of the server for which the recovery is required.

Examples

The following example shows how to recover transactions for a server that is not running.

```
asadmin recover-transactions --transactionlogdir /logs/tx
--target server1 server2
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.23 Commands used for batch job administration

This section describes the syntax and functionality of the commands used for batch job administration.

2.23.1 list-batch-job-executions

Lists the batch job executions and execution details.

Synopsis

```
asadmin [asadmin-options] list-batch-job-executions [--help]
    [--target target]
    [--executionid execution-id]
    [--long={false|true}]
    [--output output]
    [--header={false|true}]
    [instance_ID]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-batch-job-executions` subcommand lists the batch job executions and execution details.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target for which to list batch job executions and execution details.

Type: String

The following values can be specified:

- `server`
  Lists executions for the default server instance `server` and is the default value.
- *cluster-name*
  Lists executions for every server instance in the cluster.
- *instance-name*
  Lists executions for a particular server instance.

Default value: `server`

`--executionid` *execution-id* | `-x` *execution-id*

Specifies the execution ID of a specific batch job execution.

Type: String

The following values can be specified:

- *The ID of the job instance*

Default value: N/A

`--long={false|true} | -l={false|true}`

Displays detailed information about batch job executions. The default value is `false`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--output` *output* `| -o` *output*

Displays specific details about batch job executions. Use a comma-separated list to specify the details to display and their order. The values are case-insensitive.

Type: String

The following values can be specified:

- `jobname`

  Displays the name of the job.

- `executionid`

  Displays the ID assigned to the execution of the batch job. A new execution is created the first time a job is started and every time the existing execution is restarted.

- `starttime`

  Displays the start time of the execution.

- `endtime`

  Displays the finish time of the execution.

- `batchstatus`

  Displays the status of the execution as set by the batch runtime.

- `exitstatus`

  Displays the status of the execution as set by the Job XML for the job or by the batch application. By default, the `exitstatus` and the `batchstatus` are the same unless the `exitstatus` is explicitly overridden.

- `jobparameters`

  Displays the properties passed to the batch runtime for the batch job execution, listed as name/value pairs.

- `stepcount`

  Displays the number of steps in the batch job execution.

Default value: A subset of all possible headings is displayed by default.

`--header={false|true} | -h={false|true}`

Specifies whether column headings are displayed when the `--long` option is used. The default value is `true`. To suppress the headings, set the `--header` option to `false`.

Type: Boolean

The following values can be specified:

- `true`

- `false`

Default value: `true`

instance_ID

Specifies the ID of the job instance for which to list execution details.

Type: Integer

The following values can be specified:

- *The ID of the job instance*

Default value: N/A

Examples

The following example lists batch job executions for the server instance `instance1` and displays specific details.

```
asadmin list-batch-job-executions --target instance1
-o=jobname,executionid,batchstatus,exitstatus
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.23.2 list-batch-job-steps

Lists the steps for a specific batch job execution.

Synopsis

```
asadmin [asadmin-options] list-batch-job-steps [--help]
    [--long={false|true}]
    [--target target]
    [--output output]
    [--header={false|true}]
    execution_id
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-batch-job-steps` subcommand lists the steps for a specific batch job execution.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target for which the subcommand must list the batch job steps.

Type: String

The following values can be specified:

- `server`

  Lists steps for the default server instance `server` and is the default value.

- *cluster-name*

  Lists steps for every server instance in the cluster.

- *instance-name*

  Lists steps for a particular server instance.

Default value: `server`

`--long={false|true} | -l={false|true}`

Displays detailed information about batch job steps.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--output` *output* `| -o` *output*

Displays specific details about batch job steps. Use a comma-separated list to specify the details to display and their order. The values are case-insensitive. A subset of all possible headings is displayed by default.

Type: String

The following values can be specified:

- `stepname`

  Displays the name of the step.

- `stepid`

  Displays the step ID.

- `starttime`

  Displays the start time of the step.

- `endtime`

  Displays the finish time of the step.

- `batchstatus`

  Displays the status of the step as set by the batch runtime.

- `exitstatus`

  Displays the status of the step as set by the Job XML for the job or by the batch application. By default, the `exitstatus` and the `batchstatus` are the same unless the `exitstatus` is explicitly overridden.

Default value: A subset of all possible headings is displayed by default

`--header={false|true} | -h={false|true}`

Specifies whether column headings are displayed when the `--long` option is used. The default value is `true`. To suppress the headings, set the `--header` option to `false`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`execution_id`

Specifies the ID of the execution for which subcommand must list batch job steps and details.

Type: Integer

The following values can be specified:

- *The ID of the job instance*

Default value: N/A

Examples

The following example lists batch job steps and specific step details for a job execution with the execution ID of `7`. The target is the server instance `instance1`.

```
asadmin list-batch-job-steps --target instance1 o=stepname,stepid,batchstatus, 7
```

Exit Status

Exit Status	Explanation
`0`	subcommand executed successfully.
`1`	error in executing the subcommand.

2.23.3 list-batch-jobs

Lists the batch jobs.

Synopsis

```
asadmin [asadmin-options] list-batch-jobs [--help]
    [--target target]
    [--long={false|true}]
    [--output output]
    [--header={false|true}]
    [job_name]
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `list-batch-jobs` subcommand lists the batch jobs and job details.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target for which the subcommand must list batch jobs and job details.

Type: String

The following values can be specified:

- `server`
  Lists batch jobs for the default server instance `server` and is the default value.
- *cluster-name*
  Lists batch jobs for every server instance in the cluster.
- *instance-name*
  Lists batch jobs for a particular server instance.

Default value: `server`

`--long={false|true} | -l={false|true}`

Displays detailed information about batch jobs. The default value is `false`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--output` *output* `| -o` *output*

Displays specific details about batch jobs. Use a comma-separated list to specify the details to display and their order. The values are case-insensitive. The `jobname` and `instancecount` column headings are displayed by default.

Type: String

The following values can be specified:

- `jobname`
  Displays the name of the job.
- `appname`
  Displays the name of the application.
- `instancecount`
  Displays the number of job instances.
- `instanceid`

Displays the ID assigned to the job instance.

- `executionid`

  Displays the ID assigned to the execution of the batch job. A new execution is created the first time a job is started and every time the existing execution is restarted.

- `batchstatus`

  Displays the status of the job as set by the batch runtime.

- `starttime`

  Displays the start time of the job.

- `endtime`

  Displays the finish time of the job.

- `exitstatus`

  Displays the status of the job as set by the Job XML for the job or by the batch application. By default, the `exitstatus` and the `batchstatus` are the same unless the `exitstatus` is explicitly overridden.

Default value: `jobname`, `instancecount`

`--header={false|true} | -h={false|true}`

Specifies whether column headings are displayed when the `--long` option is used. The default value is `true`. To suppress the headings, set the `--header` option to `false`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`job_name`

Specifies the name of the job for which to list details.

Type: String

The following values can be specified:

- *Name of the job*

Default value: N/A

Examples

The following example lists batch jobs for the server instance `instance1`.

```
asadmin list-batch-jobs --target instance1
```

Exit Status

Exit Status	Explanation
`0`	subcommand executed successfully.
`1`	error in executing the subcommand.

2.23.4 list-batch-runtime-configuration

Displays the configuration of the batch runtime.

Synopsis

```
asadmin [asadmin-options] list-batch-runtime-configuration [--help]
    [--target target]
    [--output output]
    [--header={false|true}]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-batch-runtime-configuration` subcommand displays the configuration of the batch runtime. Batch runtime configuration data is stored in the `config` element in the `domain.xml` file.

Precondition

Domain Administration Server (DAS) is running.

Files

Batch runtime configuration data is stored in the `config` element in `domain.xml`.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target for which to list the batch runtime configuration.

Type: String

The following values can be specified:

- `server`
  Lists the batch runtime configuration for the default server instance `server` and is the default value.
- *cluster-name*
  Lists the batch runtime configuration for every server instance in the cluster.
- *instance-name*
  Lists the batch runtime configuration for a particular server instance.

Default value: `server`

`--output` *output* | `-o` *output*

Displays specific details about the batch runtime configuration.

Use a comma-separated list to specify the details to display and their order. The values are case-insensitive.

The `datasourcelookupname` and `executorservicelookupname` column headings are displayed by default.

Type: String

The following values can be specified:

- `datasourcelookupname`

  The JNDI lookup name of the data source used to store job information. By default, the batch runtime uses the default data source `jdbc/__BatchPool`.

- `executorservicelookupname`

  The JNDI lookup name of the managed executor service used to provide threads to jobs. By default, the batch runtime uses the default managed executor service `concurrent/__defaultManagedExecutorService`.

Default value: `datasourcelookupname`, `executorservicelookupname`

`--header={false|true} | -h={false|true}`

Specifies whether column headings are displayed when the `--long` option is used. The default value is `true`. To suppress the headings, set the `--header` option to `false`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

Examples

The following example lists the configuration of the batch runtime for the server instance `instance1`.

```
asadmin list-batch-runtime-configuration --target instance1
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.23.5 set-batch-runtime-configuration

Configures the batch runtime.

Synopsis

```
asadmin [asadmin-options] set-batch-runtime-configuration [--help]
    [--target target]
    [--datasourcelookupname datasource-lookup-name]
    [--executorservicelookupname executor-service-lookup-name]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `set-batch-runtime-configuration` subcommand of `asadmin` configures the batch runtime. The runtime uses a data source and a managed executor service to execute batch jobs. Batch runtime configuration data is stored in the `config` element in `domain.xml`.

Precondition

Domain Administration Server (DAS) is running.

Files

Batch runtime configuration data is stored in the `config` element in `domain.xml`.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *`target`*

Specifies the target for which the batch runtime is to be configured.

Type: String

The following values can be specified:

- `server`
  Configures the batch runtime for the default server instance `server` and is the default value.
- *cluster-name*
  Configures the batch runtime for every server instance in the cluster.
- *instance-name*
  Configures the batch runtime for a particular server instance.

Default value: `server`

`{--datasourcelookupname | -d}` *`datasource-lookup-name`*

Specifies the JNDI lookup name of the data source to be used to store job information. The default data source is `jdbc/__BatchPool`.

If the data source must be changed, stop and restart the domain and then make the change before any jobs are started or restarted. However, once the data source has been changed, information stored in the previous data source becomes inaccessible.

Type: String

The following values can be specified:

- *Name of the JNDI lookup name*

Default value: `jdbc/__BatchPool`

`{--executorservicelookupname | -x}` *`executor-service-lookup-name`*

Specifies the JNDI lookup name of the managed executor service to be used to provide threads to jobs.

Type: String

The following values can be specified:

- *JNDI lookup name of the managed executor service*

Default value: `concurrent/__defaultManagedExecutorService`

Examples

The following example configures the batch runtime for the server instance `instance1` to use an existing managed executor service named `concurrent/myExecutor`.

```
asadmin set-batch-runtime-configuration --target instance1 --
executorservicelookupname
concurrent/myExecutor
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.24 Commands used for concurrent resource administration

This section describes the syntax and functionality of the commands used for concurrent resource administration.

2.24.1 create-context-service

Creates a context service resource.

Synopsis

```
asadmin [asadmin-options] create-context-service [--help]
    [--enabled={false|true}] [--contextinfoenabled={false|true}]
    [--contextinfo={Classloader|JNDI|Security|WorkArea}]
    [--description description] [--target target]
    jndi_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-context-service` subcommand creates a context service resource.

This command is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--enabled={false|true}`

Determines whether the resource is enabled at runtime.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--contextinfoenabled={false|true}`

Determines whether container contexts are propagated to threads.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--contextinfo={Classloader|JNDI|Security|WorkArea}`

Specifies individual container contexts to propagate to threads

Type: String

The following values can be specified:

- `Classloader`
- `JNDI`
- `Security`
- `WorkArea`

Default value:

- `Classloader`
- `JNDI`
- `Security`
- `WorkArea`

`--description` *`description`*

Displays descriptive details about the resource

Type: String

The following values can be specified:

- *Descriptive details about the resource*

Default value: N/A

`--target` *`target`*

Specifies the target for which you are creating the resource.

Type: String

The following values can be specified:

- `server`
  Creates the resource for the default server instance. This is the default value.
- `domain`
  Creates the resource for the domain.
- *cluster_name*
  Creates the resource for every server instance in the specified cluster.
- *instance_name*
  Creates the resource for the specified server instance.

Default value: `server`

`jndi_name`

Specifies the JNDI name of this resource.

Type: String

The following values can be specified:

- *JNDI name*

Default value: N/A

Examples

The following example creates a context service on the server instance `instance1`.

```
asadmin create-context-service --target instance1 concurrent/myContextService
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.24.2 create-managed-executor-service

Creates a managed executor service resource.

Synopsis

```
asadmin [asadmin-options] create-managed-executor-service [--help]
    [--enabled={false|true}] [--contextinfoenabled={false|true}]
    [--contextinfo={Classloader|JNDI|Security|WorkArea}]
    [--threadpriority threadpriority] [--longrunningtasks={false|true}]
    [--hungafterseconds hungafterseconds] [--corepoolsize corepoolsize]
    [--maximumpoolsize maximumpoolsize]
    [--keepaliveseconds keepaliveseconds]
    [--threadlifetimeseconds threadlifetimeseconds]
    [--taskqueuecapacity taskqueuecapacity]
    [--target target] jndi_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-managed-executor-service` subcommand creates a managed executor service resource. This command is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--enabled={false|true}`

Determines whether the resource is enabled at runtime.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--contextinfoenabled={false|true}`

Determines whether container contexts are propagated to threads.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--contextinfo={Classloader|JNDI|Security|WorkArea}`

Specifies whether individual container contexts to propagate to threads.

Type: String

The following values can be specified:

- `Classloader`
- `JNDI`
- `Security`
- `WorkArea`

Default value: `Classloader`, `JNDI`, `Security`, and `WorkArea`

`--threadpriority` *`threadpriority`*

Specifies the priority to assign to the created threads.

Type: Integer

The following values can be specified:

- `1` to `10`

Default value: `5`

`--longrunningtasks={false|true}`

Specifies whether the resource should be used for long-running tasks.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--hungafterseconds` *`hungafterseconds`*

Specifies the number of seconds that a task can execute before it is considered unresponsive.

Type: Integer

The following values can be specified:

- `0` to `2147483647`.

Default value: `0`

`--corepoolsize` *`corepoolsize`*

Specifies the number of threads to keep in a thread pool, even if they are idle.

Type: Integer

The following values can be specified:

- `0` to `2147483647`.

Default value: `0`

`--maximumpoolsize` *`maximumpoolsize`*

Specifies the maximum number of threads that a thread pool can contain.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `2147483647`

`--keepaliveseconds` *`keepaliveseconds`*

Specifies the number of seconds that threads can remain idle when the number of threads is greater than `corepoolsize`.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `60`

`--threadlifetimeseconds` *`threadlifetimeseconds`*

Specifies the number of seconds that threads can remain in a thread pool before being purged, regardless of whether the number of threads is greater than `corepoolsize` or whether the threads are idle.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `0`

`--taskqueuecapacity` *`taskqueuecapacity`*

Specifies the number of submitted tasks that can be stored in the task queue awaiting execution.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `2147483647`

`--target` *`target`*

Specifies the target for which you are creating the resource.

Type: String

The following values can be specified:

- `server`

  Creates the resource for the default server instance. This is the default value.

- `domain`

  Creates the resource for the domain.

- *cluster_name*

Creates the resource for every server instance in the specified cluster.

- *instance_name*
  Creates the resource for the specified server instance.

Default value: `servers`

jndi_name

Specifies the JNDI name of this resource.

Type: String

The following values can be specified:

- *Specify the JNDI name*

Default value: N/A

Examples

The following example creates a managed executor service on the server instance `instance1`.

```
asadmin create-managed-executor-service --target instance1 concurrent/myExecutor
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.24.3 create-managed-scheduled-executor-service

Creates a managed scheduled executor service resource.

Synopsis

```
asadmin [asadmin-options] create-managed-scheduled-executor-service
    [--help] [--enabled={false|true}]
    [--contextinfoenabled={false|true}]
    [--contextinfo={Classloader|JNDI|Security|WorkArea}]
    [--threadpriority threadpriority] [--longrunningtasks={false|true}]
    [--hungafterseconds hungafterseconds] [--corepoolsize corepoolsize]
    [--keepaliveseconds keepaliveseconds]
    [--threadlifetimeseconds threadlifetimeseconds]
    [--description description] [--target target] jndi_name
```

Storage location

Application Server installation directory`/javaee/glassfish/bin`

Function

The `create-managed-scheduled-executor-service` subcommand creates a managed scheduled executor service resource. This command is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--enabled={false|true}`

Determines whether the resource is enabled at runtime.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--contextinfoenabled={false|true}`

Determines whether container contexts are propagated to threads.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--contextinfo={Classloader|JNDI|Security|WorkArea}`

Specifies individual container contexts to propagate to threads.

Type: String

The following values can be specified:

- `Classloader`
- `JNDI`
- `Security`
- `WorkArea`

Default value: `Classloader`, `JNDI`, `Security`, and `WorkArea`

`--threadpriority` *`threadpriority`*

Specifies the priority to assign for the created threads.

Type: Integer

The following values can be specified:

- `1` to `10`

Default value: `5`

`--longrunningtasks={false|true}`

Specifies whether the resource should be used for long-running tasks.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false`

`--hungafterseconds` *`hungafterseconds`*

Specifies the number of seconds that a task can execute before it is considered unresponsive.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `0`

`--corepoolsize` *`corepoolsize`*

Specifies the number of threads to keep in a thread pool, even if they are idle.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `0`

`--keepaliveseconds` *`keepaliveseconds`*

Specifies the number of seconds that threads can remain idle when the number of threads is greater than `corepoolsize`.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `60`

`--threadlifetimeseconds` *`threadlifetimeseconds`*

Specifies the number of seconds that threads can remain in a thread pool before being purged, regardless of whether the number of threads is greater than `corepoolsize` or whether the threads are idle.

Type: Integer

The following values can be specified:

- `0` to `2147483647`

Default value: `0`

`--description` *`description`*

Specifies descriptive details about the resource.

Type: String

The following values can be specified:

- *Provide the resource description*

Default value: N/A

`--target` *`target`*

Specifies the target for which you are creating the resource.

Type: String

The following values can be specified:

- `server`

Creates the resource for the default server instance. This is the default value.

- `domain`
  Creates the resource for the domain.
- *cluster_name*
  Creates the resource for every server instance in the specified cluster.
- *instance_name*
  Creates the resource for the specified server instance.

Default value: `server`

jndi_name

Specifies the JNDI name of this resource.

Type: String

The following values can be specified:

- *Specify the JNDI name*

Default value: N/A

Examples

The following example creates a managed scheduled executer service on the server instance `instance1`.

```
asadmin create-managed-scheduled-executor-service --target instance1
concurrent/myScheduledExecutor
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.24.4 create-managed-thread-factory

Creates a managed thread factory resource.

Synopsis

```
asadmin [asadmin-options] create-managed-thread-factory [--help]
    [--enabled={false|true}] [--contextinfoenabled={false|true}]
    [--contextinfo={Classloader|JNDI|Security|WorkArea}]
    [--threadpriority threadpriority] [--description description]
    [--target target] jndi_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-managed-thread-factory` subcommand creates a managed thread factory resource. This command is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--enabled={false|true}`

Determines whether the managed thread factory is enabled at runtime.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--contextinfoenabled={false|true}`

Determines whether container contexts are propagated to threads.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `true`

`--contextinfo={Classloader|JNDI|Security|WorkArea}`

Specifies individual container contexts to propagate to threads.

Type: String

The following values can be specified:

- `Classloader`
- `JNDI`
- `Security`
- `WorkArea`

Default value: `Classloader`, `JNDI`, `Security`, and `WorkArea`

`--threadpriority` *`threadpriority`*

Specifies the priority to assign for the created threads.

Type: Integer

The following values can be specified:

- `1` to `10`

Default value: `5`

`--description` *description*

Specifies the descriptive details about the resource.

Type: String

The following values can be specified:

- *Provide the description of the resource*

Default value: N/A

`--target` *target*

Specifies the target for which you are creating the resource.

Type: String

The following values can be specified:

- `server`
  Creates the resource for the default server instance. This is the default value.
- `domain`
  Creates the resource for the domain.
- *cluster_name*
  Creates the resource for every server instance in the specified cluster.
- *instance_name*
  Creates the resource for the specified server instance.

Default value: `server`

jndi_name

Specifies the JNDI name of this resource.

Type: String

The following values can be specified:

- *Specify the JNDI name*

Default value: N/A

Examples

The following example creates a `managed-thread-factory` on the server instance `instance1`.

```
asadmin create-managed-thread-factory --target instance1
concurrent/myThreadFactory
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.24.5 delete-context-service

Deletes a context service resource with the specified JNDI name.

Synopsis

```
asadmin [asadmin-options] delete-context-service [--help]
    [--target target] context-service-name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-context-service` subcommand removes a context service resource with the specified JNDI name.

This subcommand is supported in remote mode only.

Precondition

DAS has to be in a running state.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target from which you are deleting the resource.

Type: String

The following values can be specified:

- `server`
  Deletes the resource from the default server instance. This is the default value.
- `domain`
  Deletes the resource from the domain.
- *cluster-name*
  Deletes the resource from every server instance in the specified cluster.
- *instance-name*
  Deletes the resource from the specified server instance.

Default value: `server`

context-service-name

Specifies the JNDI name of the resource.

Type: String

The following values can be specified:

- *JNDI name of the resource*

Default value: N/A

Examples

The following example deletes the context service resource named `concurrent/myContextService` on the server instance `instance1`.

```
asadmin delete-context-service --target instance1 concurrent/myContextService
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.24.6 delete-managed-executor-service

Removes a managed executor service resource with the specified JNDI name.

Synopsis

```
asadmin [asadmin-options] delete-managed-executor-service [--help]
    [--target target]
    managed_executor_service_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-managed-executor-service` subcommand removes a managed executor service resource with the specified JNDI name. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help|-?`
Displays the help text for the subcommand.

`--target` *target*
Specifies the target from which you are deleting the resource.
Type: String
The following values can be specified:

- `server`
  Deletes the resource from the default server instance. This is the default value.
- `domain`
  Deletes the resource from the domain.
- *cluster_name*
  Deletes the resource from every server instance in the specified cluster.
- *instance_name*

Deletes the resource from the specified server instance.

Default value: `server`

`managed_executor_service_name`

Specifies the JNDI name of the resource.

Type: String

The following values can be specified:

- *JNDI name of the resource*

Default value: N/A

Examples

The following example deletes the managed executor service resource named `concurrent/myExecutor` on the server instance `instance1`.

```
asadmin delete-managed-executor-service --target instance1 concurrent/myExecutor
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.24.7 delete-managed-scheduled-executor-service

Removes a managed scheduled executor service resource with the specified JNDI name.

Synopsis

```
asadmin [asadmin-options] delete-managed-scheduled-executor-service
    [--help] [--target target]
    managed_scheduled_executor_service_name
```

Storage location

`Application Server installation directory``/javaee/glassfish/bin`

Function

The `delete-managed-scheduled-executor-service` subcommand removes a managed scheduled executor service resource with the specified JNDI name. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target from which you are deleting the resource.

Type: String

The following values can be specified:

- `server`
  Deletes the resource from the default `server` instance. This is the default value.
- `domain`
  Deletes the resource from the domain.
- *cluster_name*
  Deletes the resource from every server instance in the specified cluster.
- *instance_name*
  Deletes the resource from the specified server instance.

Default value: `server`

managed_scheduled_executor_service_name

Specifies the JNDI name of the resource.

Type: String

The following values can be specified:

- *JNDI name of the resource*

Default value: N/A

Examples

The following example deletes the managed scheduled executor service resource named `concurrent/myScheduledExecutor` on the server instance `instance1`.

```
asadmin delete-managed-scheduled-executor-service --target instance1
concurrent/myScheduledExecutor
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.24.8 delete-managed-thread-factory

Removes a managed thread factory resource with the specified JNDI name.

Synopsis

```
asadmin [asadmin-options] delete-managed-thread-factory [--help]
    [--target target]
    managed_thread_factory_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-managed-thread-factory` subcommand removes a managed thread factory resource with the specified JNDI name.

This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target from which you are deleting the resource.

Type: String

The following values can be specified:

- `server`
  Deletes the resource from the default server instance. This is the default value.
- `domain`
  Deletes the resource from the domain.
- *cluster_name*
  Deletes the resource from every server instance in the specified cluster.
- *instance_name*
  Deletes the resource from the specified server instance.

Default value: `server`

managed_thread_factory_name

Specifies the JNDI name of the resource.

Type: String

The following values can be specified:

- *JNDI name of the resource*

Default value: N/A

Examples

The following example deletes the managed thread factory resource named `concurrent/myThreadFactory` on the server instance `instance1`.

```
asadmin delete-managed-thread-factory --target instance1
concurrent/myThreadFactory
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.24.9 list-context-services

Lists the context service resources.

Synopsis

```
asadmin [asadmin-options] list-context-services [--help]
    [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-context-services` subcommand lists context service resources. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

target

Specifies the target for which context service resources are to be listed.

Type: String

The following values can be specified:

- `server`

  Lists the resources on the default server instance. This is the default value.

- `domain`

  Lists the resources for the domain.

- *cluster-name*
  Lists the resources on all server instances in the specified cluster.
- *instance-name*
  Lists the resources on a specified server instance.

Default value: `server`

Examples

This example lists context service resources on the default server instance.

```
asadmin list-context-services
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.24.10 list-managed-executor-services

Lists the managed executor service resources.

Synopsis

```
asadmin [asadmin-options] list-managed-executor-services [--help]
    [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-managed-executor-services` subcommand of asadmin lists the managed executor service resources. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`
Displays the help text for the subcommand.

target
Specifies the target for which the managed executor service resources are to be listed.
Type: String

The following values can be specified:

- `server`
  Lists the resources on the default server instance. This is the default value.
- `domain`
  Lists the resources for the domain.
- *cluster-name*
  Lists the resources on all the server instances in the specified cluster.
- *instance-name*
  Lists the resources on a specified server instance.

Default value: `server`

Examples

The following example lists the managed executor service resources on the default server instance.

```
asadmin list-managed-executor-services
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.24.11 list-managed-scheduled-executor-services

Lists the managed scheduled executor service resources.

Synopsis

```
asadmin [asadmin-options] list-managed-scheduled-executor-services
    [--help] [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-managed-scheduled-executor-services` subcommand of asadmin lists the managed scheduled executor service resources. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

target

Specifies the target for which the managed scheduled executor service resources are to be listed.

Type: String

The following values can be specified:

- `server`
  Lists the resources on the default server instance. This is the default value.
- `domain`
  Lists the resources for the domain.
- *cluster-name*
  Lists the resources on all the server instances in the specified cluster.
- *instance-name*
  Lists the resources on a specified server instance.

Default value: `server`

Examples

The following example lists the managed scheduled executor service resources on the default server instance.

```
asadmin list-managed-scheduled-executor-services
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.24.12 list-managed-thread-factories

Lists the managed thread factory resources.

Synopsis

```
asadmin [asadmin-options] list-managed-thread-factories [--help]
    [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-managed-thread-factories` subcommand of asadmin lists the managed thread factory resources. This subcommand is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`target`

Specifies the target for which the managed thread factory resources are to be listed.

Type: String

The following values can be specified:

- `server`
  Lists the resources on the default server instance. This is the default value.
- `domain`
  Lists the resources for the domain.
- *cluster-name*
  Lists the resources on all the server instances in the specified cluster.
- *instance-name*
  Lists the resources on a specified server instance.

Default value: `server`

Examples

The following example lists the managed thread factory resources on the default server instance.

```
asadmin list-managed-thread-factories
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.25 Commands used for message security provider administration

This section describes the syntax and functionality of the commands used for message security provider administration.

2.25.1 create-message-security-provider

Creates a message security provider that specifies how SOAP messages will be secured. This option is applicable for administrators.

Synopsis

```
asadmin [asadmin-options] create-message-security-provider [--help]
    [--target target] --classname provider_class
    --layer message_layer [--providertype provider_type]
    [--requestauthsource request_auth_source]
    [--requestauthrecipient request_auth_recipient]
    [--responseauthsource response_auth_source]
    [--responseauthrecipient response_auth_recipient]
    [--isdefaultprovider]
    [--property name=value[:name=value]...]
    provider_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-message-security-provider` subcommand enables the administrator to create a message security provider for the security service which specifies how SOAP messages will be secured. This command is supported in remote mode only.

Precondition

The DAS should be in a running state.

Files

You can specify the security configuration file `security.config` by using the `-property` option.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target for which you are creating the message security provider.

Type: String

The following values can be specified:

- `server`

  Creates the provider for the default server instance and is the default value.

- *cluster_name*
  Creates the provider for every server instance in the cluster.
- *instance_name*
  Creates the provider for a particular sever instance.

Default value: `server`

`--classname` *provider_class*

Defines the Java implementation class of the provider. Client authentication providers must implement the `com.sun.enterprise.security.jauth.ClientAuthModule` interface. Server-side providers must implement the `com.sun.enterprise.security.jauth.ServerAuthModule` interface. A provider may implement both interfaces, but it must implement the interface corresponding to its provider type.

Type: String

The following values can be specified:

A Java class name which implements the interface

- `com.sun.enterprise.security.jauth.ClientAuthModule`
- `com.sun.enterprise.security.jauth.ServerAuthModule`

Default value: N/A

`--layer` *message_layer*

Specifies the `message-layer` entity used to define the value of the `auth-layer` attribute of `message-security-config` elements.

Type: String

The following values can be specified:

- `HttpServlet`
- `SOAP`

Default value: `HttpServlet`

`--providertype` *provider_type*

Establishes whether the provider is to be used as client authentication provider, server authentication provider, or both.

Type: String

The following values can be specified:

- `client`
- `server`
- `client-server`

Default value: N/A

`--requestauthsource` *request_auth_source*

Specifies the `auth-source` attribute that defines a requirement for the `message-layer` sender authentication (For example, `username password`) or the content authentication (For example, `digital signature`), to be applied to request messages.

Type: String

The following values can be specified:

- `sender`
- `content`

Default value: N/A

`--requestauthrecipient` *request_auth_recipient*

Specifies the `auth-recipient` attribute that defines a requirement for message-layer authentication of the receiver of the response message to its sender (for example, by XML encryption).

Type: String

The following values can be specified:

- `before-content`
- `after-content`

Default value: `after-content`

`--responseauthsource` *response_auth_source*

Specifies the `auth-source` attribute that defines a requirement for the `message-layer` sender authentication (For example, `username password`) or the content authentication (For example, digital signature) to be applied to response messages.

Type: String

The following values can be specified:

- `sender`
- `content`

Default value: N/A

`--responseauthrecipient` *response_auth_recipient*

Specifies the `auth-recipient` attribute that defines a requirement for message-layer authentication of the receiver of the response message to its sender (for example, by XML encryption).

Type: String

The following values can be specified:

- `before-content`
- `after-content`

Default value: `after-content`

`--isdefaultprovider={false|true}`

On the layer whose type is specified by the *providertype* argument, specifies the provider as the `default provider`. Default value: The provider is not specified as the `default provider`.

Type: Boolean

The following values can be specified:

- `true`
- `false`

Default value: `false` (The provider is not specified as the `default provider`)

`--property` *name*`=`*value*`[:`*name*`=`*value*`]...`

Passes `provider-specific` property values to the provider when it is initialized. Properties passed in this way might include key aliases to be used by the provider to get keys from keystores, signing, canonicalization, encryption algorithms,etc.

Type: String

The following values can be specified:

- `security.config =`*value*

Specifies the location of the message security configuration file. To point to a configuration file in the `domain-dir/config` directory, use the system property `${com.sun.aas.instanceRoot}/config/`, for example: `${com.sun.aas.instanceRoot}/config/wss-server-config-1.0.xml`. The default is `domain-dir/config/ wss-serverconfig-1.0.xml`.

Type: String

Default value: *domain-dir*`/config/ wss-serverconfig-1.0.xml`

Range Value: N/A

- `debug =`*value*

  Enables dumping of server provider debug messages to the server log, if `true`. The default is `false`.

  Type: Boolean

  Default value: `false`

  Range Value:

  `true`

  `false`

- `dynamic.username.password =`*value*

  Signals the provider runtime to collect the user name and password from the `CallbackHandler` for each request, if set to `true`. If `false`, the user name and password for `wsse:UsernameToken(s)` is collected once, during module initialization. This property is only applicable for a `ClientAuthModule`. The default is `false`.

  Type: boolean

  Default value: `false`

  Range Value:

  `true`

  `false`

- `encryption.key.alias =`*value*

  Specifies the encryption key used by the provider. The key is identified by its keystore alias. The default value is `s1as`.

  Type: String

  Default value: `s1as`

  Range Value: N/A

- `signature.key.alias =`*value*

  Specifies the signature key used by the provider. The key is identified by its keystore alias. The default value is `s1as`.

  Type: String

  Default value: `s1as`

  Range Value: N/A

Default value: N/A

provider_name

Specifies the name of the provider used to reference the `provider-config` element.

Type: String

The following values can be specified:

- *Specify the name of the provider*

Default value: N/A

Examples

The following example creates a message security provider on server instance `instance1`.

```
asadmin create-message-security-provider --target instance1
--classname com.sun.enterprise.security.jauth.ClientAuthModule --layer SOAP
--providertype client mySecurityProvider
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.25.2 delete-message-security-provider

Deletes a message security provider. This option is used by administrators.

Synopsis

```
asadmin [asadmin-options] delete-message-security-provider [--help]
    [--target target] --layer message_layer
    provider_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-message-security-provider` subcommand enables administrators to delete a message security provider. In terms of what happens when this subcommand is run, the `provider-config` sub-element for the given message layer (`message-security-config` element of `domain.xml` is deleted. The `domain.xml` file specifies parameters and properties to Java EE Server). The options specified in the list below apply to attributes within the `message-security-config` and `provider-config` sub-elements of the `domain.xml` file. If the message-layer (`message-security-config` attribute) does not exist, it is created, and then the `provider-config` is created under it.

This command is supported in remote mode only.

Precondition

Domain Administration Server (DAS) is running.

Files

When `delete-message-security-provider` subcommand is run, the `provider-config` sub-element for the given message layer (`message-security-config` element of `domain.xml` is deleted. The `domain.xml` file specifies parameters and properties to Java EE Server). The options specified in the list below apply to attributes within the `message-security-config` and `provider-config` sub-elements of the `domain.xml` file.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--target` *target*

Specifies the target from which you are deleting the message security provider.

Type: String

The following values can be specified:

- `server`
  Deletes the message security provider from the default server instance `server` and is the default value.
- `domain`
  Deletes the message security provider from the domain.
- *cluster_name*
  Deletes the message security provider from every server instance in the cluster.
- *instance_name*
  Deletes the message security provider from a particular sever instance.

Default value: `server`

`--layer` *message_layer*

Specifies the message layer from which the provider has to be deleted.

Type: String

The following values can be specified:

- `HttpServlet`
- `SOAP`

Default value: `HttpServlet`

provider_name

Specifies the name of the provider used to reference the `provider-config` element.

Type: String

The following values can be specified:

- *Name of the provider*

Default value: N/A

Examples

The following example shows how to delete a message security provider for a client on the server instance `instance1`.

```
asadmin delete-message-security-provider --target instance1
--layer SOAP mySecurityProvider
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
1	error in executing the command.

2.25.3 list-message-security-providers

Lists all the security message providers for the given message layer.

Synopsis

```
asadmin [asadmin-options] list-message-security-providers [--help]
    [--layer message_layer]
    [target]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-message-security-providers` subcommand of asadmin enables administrators to list all security message providers (`provider-config` sub-elements) for the given message layer (`message-security-config` element of `domain.xml`). This subcommand is supported in remote mode only.

Execution permission

Superuser

Precondition

Domain Administration Server (DAS) is running.

Arguments

`--help | -?`

Displays the help text for the subcommand.

`--layer` *message_layer*

Specifies the message-layer for which the provider has to be listed. The default value is `HttpServlet`.

Type: String

The following values can be specified:

- `SOAP`
- `HttpServlet`

Default value: `HttpServlet`

target

Restricts the listing to message security providers for a specific target.

Type: String

The following values can be specified:

- `server`

  Lists providers for the default server instance `server` and is the default value.
- `domain`

  Lists providers for the domain.
- *cluster_name*

Lists providers for the server instances in the cluster.

- *instance_name*
  Lists providers for a particular server instance.

Default value: `server`

Examples

The following example shows how to list message security providers for a message layer.

```
asadmin list-message-security-providers --layer SOAP
```

Exit Status

Exit Status	Explanation
0	subcommand executed successfully.
1	error in executing the subcommand.

2.26 Commands used to manage the event hook command

This section defines the syntax and functionality of the commands used to manage the event hook command.

2.26.1 create-event-hook

The `create-event-hook` subcommand registers an event hook command.

Synopsis

```
asadmin [asadmin_options] create-event-hook --messageid=messageid
    --script=script_file_path
    [--monitoring-time-span=monitoring_time_span
    --monitoring-max-count=monitoring_max_count] event_hook_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `create-event-hook` subcommand registers an event hook command, which is executed automatically when an event occurs in the domain administration server. You need to specify a script file created by the user for the event hook command.

When an event is detected, the registered command is executed in the location where the specified script file is placed.

You can specify any name for the event hook command. However, if another event hook command is already registered and uses the same value for the following argument, an error occurs even if the name is not yet registered:

- `--messageid`

This subcommand is supported only in the remote mode.

Execution permission

General user

Arguments

`--messageid=`*messageid*

Specifies the message ID that triggers the execution of the event hook command.

Type: String

You can specify the following values:

- Message ID published by the system.

Default value: (None. You must specify a value.)

`--script=`*script_file_path*

Specifies the script file to be registered as the event hook command. Specify an absolute path in the domain administration server.

Type: String

You can specify the following values:

- Absolute path of the existing file.

Default value: (None. You must specify a value.)

`--monitoring-time-span=`*`monitoring_time_span`*

Specifies the time interval to configure the maximum permitted number of executions.

The value is specified in seconds. If this value is set to `0`, the event hook command runs indefinitely without monitoring the execution.

Executing an event hook command when an event is detected might result in repeatedly issuing the event that is subject to detection. As a result, the event hook command might be executed indefinitely. You can prevent the command from running indefinitely by specifying this option and the `--monitoring-max-count` option simultaneously.

Type: Integer

You can specify the following values:

- `0` to `2147483647`

Default value: `0`

`--monitoring-max-count=`*`monitoring_max_count`*

Specify the maximum permitted number of executions for the same event type, within the time specified in the `--monitoring-time-span` option. If `0` is specified, the event hook command is executed indefinitely.

The events that satisfy the conditions below are referred to as "events of the same type".

Server events are of the same type if all the following items are identical among the server events:

- Message ID that triggered the event.
- Name of the server where the event occurred.

Node events are of the same type if all the following items are identical among the node events:

- Message ID that triggered the event.
- Name of the node where the event occurred.

Non-server events and non-node events are of the same type if all the following items are identical among those servers:

- Message ID that triggered the event.

Type: Integer

You can specify the following values:

- `0` to `256`

Default value: `0`

`event_hook_name`

Specifies the name of the event hook command.

Type: String

You can specify the following values:

- ASCII characters
- The name can contain the following characters:
  Lowercase letters: `a` to `z`
  Uppercase letters: `A` to `Z`
  Numbers: `0` to `9`

Hyphens: -

Underscores: _

- The first character can either be an uppercase or a lowercase letter.
- The following names that exist in the domain cannot be specified:
  Node name
  Java EE server name
  Web server name
  PRF server name
  Cluster name
  Name of a dependency relation between servers
  Name of the configuration of the Java EE server
- The following names that are listed cannot be specified because they are either used by the system or are reserved words:
  `domain`
  `server`
  `default`
  `server-config`
  `default-config`
  `default-webserver-config`
  `default-prf-config`
  `javaee`
  `webserver`
  `prf`
  `cluster`
  `redirect`
  `prf-relation`
  Name that begins with `HJES_`
  prebuilt_web_server_name`-config`
  prebuilt_PRF_name`-config`

Default value: (None. You must specify a value.)

Examples

The following example runs the event hook command when the server starts successfully.

```
asadmin create-event-hook --messageid KDKD10083-I
 --script /work/failure-script.sh Event01
```

The following example runs the event hook command when the server stops successfully (and the execution monitoring option of the event hook command is specified).

```
asadmin create-event-hook --messageid KDKD10094-I
--script /work/failure-script.sh --monitoring-time-span=180
--monitoring-max-count=3  Event02
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

Notes

- Do not delete the script file for the event hook command, until you unregister the event hook command.

2.26.2 delete-event-hook

The `delete-event-hook` subcommand unregisters the event hook command.

Synopsis

```
asadmin [asadmin_options] delete-event-hook event_hook_name
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `delete-event-hook` subcommand unregisters an event hook command.

This subcommand is supported only in the remote mode.

Execution permission

General user

Precondition

The event hook command has been registered.

Arguments

event_hook_name

Specifies the name of the event hook command to be unregistered.

Type: String

You can specify the following values:

- Name of an existing event hook command.

Default value: (None. You must specify a value.)

Examples

- The following example unregisters the event hook command `Event01`:

```
asadmin delete-event-hook Event01
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

2.26.3 list-event-hooks

The `list-event-hooks` subcommand lists the event hook commands.

Synopsis

```
asadmin [asadmin_options] list-event-hooks [--long={false|true}]
```

Storage location

Application Server installation directory/javaee/glassfish/bin

Function

The `list-event-hooks` subcommand lists the registered event hook commands.

This subcommand is supported only in the remote mode.

Execution permission

General user

Arguments

`--long={false|true}`

Specifies whether to display the header information.

Type: Boolean

You can specify the following values:

- `true`

  Displays all the event hook commands and header information. The list displays all the items.

- `false`

  Displays all the event hook commands. It does not display the header information. The list displays the values for names, message IDs, and script paths.

Default value: `false`

Output Format

The first line is the header of the output information and displays a fixed character string, which indicates the output format.

Between columns, the longest character string within a column is separated from the beginning of the next column by two spaces.

Each line, except the header, displays the `create-event-hook` subcommand in the order of its registration.

```
NAME   MESSAGEID   SCRIPT         MONITORING-TIME-SPAN  MONITORING-MAX-COUNT
name   message_ID  script_path  time_interval
maximum_permitted_number_of_executions
```

name: Displays the event hook command name.

message_ID: Displays the message ID.

script_path: Displays the absolute path of the script to be run.

time_interval: Displays the time interval to configure the maximum permitted number of executions.

maximum_permitted_number_of_executions: Displays the maximum permitted number of executions.

Examples

The following example displays all the registered event hook commands including the header.

```
list-event-hooks --long=true
```

Example of output:

```
NAME       MESSAGEID    SCRIPT        MONITORING-TIME-SPAN  MONITORING-MAX-COUNT
Event01    KDKD10083-I  /work/failure-script.sh  180          3
```

The following example displays all the registered event hook commands without the header.

```
list-event-hooks --long=false
```

Example of output:

```
Event01  KDKD10083-I  /work/failure-script.sh
```

Exit Status

Exit Status	Explanation
0	Normal termination.
1	Abnormal termination.

3 Commands used in the Web server

This chapter describes the syntax and functionality of the commands used in the Web server.

3.1 List of commands to be used in the Web Server

The following table shows a list of commands to be used in the Web Server.

Commands used in Web server operations

Command Name	Classification	Summary
htpasswd	Registering the user name and password in the password file and changing the password	Registers a user name and password in the password file. You can change the password or delete a user name registered in the password file.
httpsd	Starting Web Server	Starts up the Web server. You use this method to specify an httpsd.conf file name other than the default name or to specify the root directory and the httpsd.conf file when starting the server.
httpsdctl	Starts, restarts, or stops the Web server	The `httpsdctl` command starts, restarts, or stops the Web server.
hwstraceinfo	Collecting the internal trace	When an application program is executed and a request is received, the events that occur in the system are collected as internal trace.
logresolve	Converting the IP address of the access log file into a host name	The `logresolve` command converts the IP address of the access log file (that has IP address at the beginning of the record) into the host, and outputs to the new log file.

Commands used for authentication and encryption with SSL

Command Name	Classification	Summary
hwscertutil cert -hash	Creating a hash link	Creating a hash link.
hwscertutil cert -outform	Converting the certificate format	The `hwscertutil cert -outform` command converts the certificate format.
hwscertutil cert -text	Displaying certificate contents	Displaying certificate contents.
hwscertutil req	Displaying the contents of a Certificate Signing Request (CSR)	Displaying the contents of a Certificate Signing Request (CSR).
hwscertutil reqgen	Creating a Certificate Signing Request (CSR).	Creating a Certificate Signing Request (CSR). The created CSR file is submitted to the CA, which then issues the signed certificate.
hwskeygen	Creating a private key for the Web server	Creating a private key for the Web server. The created Web server private key file is specified in the `SSLCertificateKeyFile` directive.
sslpasswd	Creating password file for password-protected server private key	Creating password file for password-protected server private key. Before using the password-protected server private key, you can save the password in a file and then set the directive, so that you do not have to enter the password when the Web server starts.

Important note

- Specify the value which can be specified by each command as the argument of a command. Operation is not guaranteed when other values are specified.
- Even if a command is executed normally, errors may have occurred. Check the message log and stack trace log.

3.2 Commands used in Web server operations

This section describes the syntax and functionality of the commands used for operating a Web server.

3.2.1 htpasswd

The `htpasswd` command registers the user name and password in the password file, and changes the password.

Synopsis

```
htpasswd [-b] [-c|-D] password-file-name user-name [password]
```

Storage location

Application Server installation directory/httpsd/sbin

Function

Registers a user name and password in the password file. You can change the password or delete a user name registered in the password file.

Execution permission

Superuser

Arguments

`-b`

Specify this parameter when you specify the password in the command line.

`-c`

Specify this parameter when you create a new password file. You need not specify `-c` when you add a user and change the password in an already created password file.

`-D`

Specify this parameter when you delete a user registration. If the specified user is registered in the specified password file, the utility deletes the corresponding user from the password file.

password-file-name

Specify the password file that registers, changes, or deletes password.

user-name

Specify the user name for which password is to be registered, changed, or deleted.

password

Specify the password to be registered or changed. You can specify this parameter only when `-b` option is specified.

Examples

If you specify the password file name, the user name to be registered, or the user name for which the password is to be changed, and run the `htpasswd`, the input of respective password is requested. If you enter the password twice,

including the confirmation of password entry, the user name and the password of that user are registered in the password file:

```
$ "Application Server installation directory/httpsd/sbin/htpasswd" .passwd userxx ...
1.
New password:                    ...2.
Re-type new password:            ...3.
Adding password for user userxx  ...4.
$
```

1. Change the password of userxx
2. Enter a new password
3. Re-enter the new password
4. End the registration of new password

When deleting the registration, start the `htpasswd` utility by specifying the `-D` option, the password file name, and the user name that is to be deleted.

```
$ "Application Server installation directory/httpsd/sbin/htpasswd" -D .passwd
userxx ...1.
Deleting password for user userxx    ...2.
$
```

1. Delete registration of userxx
2. Delete the registration of userxx and exit

Exit Status

Exit Status	Explanation
`0`	command executed successfully.
Other than `0`	error in executing the command.

Notes

- The maximum length of a user name and password is 128 characters.
- When the `htpasswd` command is executed, a temporary work file is created in the directory in which the password file is created. The work file name is *password-file-name*.*process-ID*. The work file is deleted when the `htpasswd` command ends. However, the work file might not be deleted if you cancel to end the htpasswd utility while it is running. Manually delete the work file if it is not deleted automatically.

3.2.2 httpsd

The `httpsd` command starts the Web Server.

Synopsis

```
httpsd [[-d directory] [-f file-name] [-R directory]|-v|-t]
    [-D HWS_OPTION_HWS2]
```

Storage location

Application Server installation directory/httpsd/sbin

Function

Starts up the Web server.

You use this method to specify an httpsd.conf file name other than the default name or to specify the root directory and the httpsd.conf file when starting the server.

If you are using an encrypted private key for SSL, a request to enter the private key password is displayed when starting Web server.

Execution permission

Superuser or general user

Precondition

The PRF daemon needs to be running.

Environment variable

- LD_LIBRARY_PATH(Linux)
  Specify *Application Server installation directory*/common/PRF/lib.
- PRFSPOOL
  Sets the execution environment directory for the PRF daemon.
- TZ
  Sets the time zone.

Arguments

-d *directory*

You can specify the default value used when the ServerRoot directive is not specified in the configuration file.

-f *file-name*

You can specify the httpsd.conf file. Specify the httpsd.conf file with the absolute path or the relative path from the value specified in the ServerRoot directive.

-R *directory*

This option specifies the absolute path of the directory storing the DSO execution library.

-v

This option displays the version information. If you specify this option, Web Server will not start.

-t

This option checks the configuration file syntax. If there is a syntax error, an error message will be displayed on the screen. If you specify this option, Web Server will not start.

-D HWS_OPTION_HWS2

Specify this option to use the SSL functionality in Red Hat Enterprise Linux Server 6 (64-bit x86_64). For other platforms or if you do not use the SSL functionality, do not specify this option.

How to confirm the start

To confirm the start of Web server, check the control process. The process name (execution program name) is `httpsd`.

In addition, after starting Web server, confirm that the `AH00163` message was output to the error log.

How to restart

You can use the `kill` command to restart Web Server:

```
kill {-HUP|-USR1} `cat PidFile-directive-specified-value`
```

`-HUP`
Restart as when using the `restart` option of the `httpsdctl` command.

`-USR1`
Restart as when using the `graceful` option of the `httpsdctl` command.

`PidFile-directive-specified-value`
Specify a value (file name) that is specified in the PidFile directive.

How to end

When you use the `httpsd` command to start Web Server, execute the following command to end the process and stop Web Server:

```
kill {-TERM|-USR2} `cat PidFile-directive-specified-value`
```

`-TERM`
Stop as when using the `stop` option of the `httpsdctl` command.

`-USR2`
Stops as when using the `gracefulstop` option of the `httpsdctl` command.

Related topics

- 3.4 Operation by general user

3.2.3 httpsdctl

The `httpsdctl` command starts, restarts, or stops the Web server.

Synopsis

```
httpsdctl {start|stop|restart|graceful|gracefulstop|configtest|help}
```

Storage location

`Application Server installation directory``/httpsd/sbin`

Function

Starts, restarts, or stops the Web server.

Execution permission

Superuser or general user

Precondition

The PRF daemon needs to be running.

Environment variable

- `LD_LIBRARY_PATH(Linux)`
  Specify *Application Server installation directory*`/common/PRF/lib`.
- `PRFSPOOL`
  Sets the execution environment directory for the PRF daemon.
- `TZ`
  Sets the time zone.

Files

`httpsd.conf` file

Arguments

`start`
This option starts Web server. If you are using an encrypted private key for SSL, a request to enter the private key password is displayed when starting Web server.

`stop`
This option stops Web server.

`restart`
This option restarts Web server (server restart). End the running server processes immediately. After all the server processes end, restart Web server. When you restart Web server, the change in the value specified in the `MaxRequestWorkers` directive is not applied, and the value that is set earlier will continue to be used. If you changed the value specified in the `Listen` directive and the settings for the private key used in SSL communication (that is, the `SSLCertificateKeyFile` directive), stop and restart Web server.

`graceful`
Restart Web server. End the running server processes, and then stop the server. Start the server process with a new configuration file when required. When you restart Web server, the change in the value that is specified in the `MaxRequestWorkers` directive is not applied, and the value that is set earlier will continue to be used. If you changed the value specified in the `Listen` directive and the settings for the private key used in SSL communication (that is, the `SSLCertificateKeyFile` directive), stop and restart Web server.

`gracefulstop`
This option stops Web server. End the running server processes, and then stop the server. If you do not end the running processes, the processes will end when the waiting time specified in the `HWSGracefulStopTimeout` directive ends.

`configtest`
This option checks the configuration file syntax. If there is a syntax error, an error message is displayed on the screen. If you specify the configtest option, Web server will not start.

help

This option displays the httpsdctl help.

Examples

The following example starts Web server. If you are using an encrypted private key, enter the password.

```
/opt/hitachi/APServer/httpsd/sbin/httpsdctl start
Enter PEM pass phrase:
```

How to confirm the start

To confirm the start of Web server, check the control process. The process name (execution program name) is httpsd.

In addition, after starting Web server, confirm that the AH00163 message was output to the error log.

Notes

- When you stop the Web server with httpsdctl stop and gracefulstop, if the configuration file of Web server is not correctly defined, an error will occur in the httpsdctl execution and the Web server will not stop.
- When you restart the Web server with httpsdctl restart and graceful, if the configuration file of Web server is not defined correctly, an error will occur in the httpsdctl execution and the Web server will not restart.
- When you start, restart, and stop Web server with the httpsdctl command, the message showing the completion of start and stop is not output.

Related topics

- 3.4 Operation by general user

3.2.4 hwstraceinfo

The hwstraceinfo command collects the internal trace.

Synopsis

```
hwstraceinfo -i shared-memory-identifier {-l file-name|-r}
```

Storage location

Application Server installation directory/httpsd/sbin

Function

When an application program is executed and a request is received, the events that occur in the system are collected as internal trace. The internal trace is output once to the shared memory, and then it is output to the file as per the specification in the directive or command.

Collecting trace information

Internal traces are collected in the shared memory when various events occur in the Web server. The memory identifiers of shared memory are stored in the file specified in the HWSTraceIdFile directive.

How to output to a file

The internal trace that is collected in the shared memory is output to a file when the server process terminates abnormally or when the `hwstraceinfo` command is executed. When the server process terminates abnormally, the trace is output to the file specified in the `HWSTraceLogFile` directive.

Specify the memory identifier and file name of output destination, in the `hwstraceinfo` command.

The internal trace information output file size is as follows:

(size of output from the `ps -efl` command) + (size of output from the the `vmstat` command) + (size of output from thethe `ipcs -a` command) + (7KB x MaxRequestWorkers value)

Execution permission

Superuser or general user

Arguments

`-i` *`shared-memory-identifier`*

This parameter specifies the shared memory identifier that is output to the file specified in the `HWSTraceIdFile` directive.

`-l` *`file-name`*

This parameter specifies the file that outputs the trace corresponding to the shared memory identifier specified with `-i`.

`-r`

This parameter releases the shared memory allocated to the shared memory identifier specified in `-i`.

The shared memory for trace remains even if the Web server stops. Use this parameter to release the remaining shared memory.

Examples

The following is an example to output the trace corresponding to the shared memory identifier `1800_1133780652_0`, to the `traceinfo.log` file:

```
hwstraceinfo -i 1800_1133780652_0 -l traceinfo.log
```

Notes

To retain trace information, the Web server does not release the shared memory even when the Web server stops. The shared memory is reused when the server restarts.

When you stop the server and restart it later, the Web server releases the shared memory once and then restores it depending upon the file value specified in the `HWSTraceIdFile` directive. However, in the following cases, the shared memory used earlier cannot be released:

- When the same user does not restart the server (the User directive value or the Group directive value has changed)
- The value of the `HWSTraceIdFile` directive has changed
- The file specified in the `HWSTraceIdFile` directive is deleted

When you release the shared memory, execute the `hwstraceinfo` command in which `-r` is specified.

3.2.5 logresolve

The `logresolve` command converts the IP address of the access log file into a host name.

Synopsis

```
logresolve [-s file-name] [-c] < Access-log-file-name > New-log-file-name
```

Storage location

Application Server installation directory/httpsd/sbin

Function

The `logresolve` command converts the IP address of the access log file (that has IP address at the beginning of the record) into the host, and outputs to the new log file. The conversion rule depends on the reverse lookup of host name.

Execution permission

Superuser

Arguments

`-s` *file-name*

This parameter specifies the file to which conversion information is to be output.

`-c`

Use this option to check whether the host name after conversion matches the IP address before conversion.

Access-log-file-name

This parameter specifies the input log file name. Reverse the look up of the host name from the IP address mentioned in the input file. The IP address must be at the top of the record. If an attempt to retrieve the host name fails, the IP address is output to the new log file.

New-log-file-name

This parameter specifies a file name that outputs the access log with IP address converted to the host.

Examples

Convert the IP address of access log stored in the `logs\access.log`, into the host name.

Access log file: `logs\access.log`

New log file: `logs\new_access.log`

```
logresolve < logs\access.log > logs\new_access.log
```

Exit Status

Exit Status	Explanation
0	command executed successfully.
Other than 0	error in executing the command.

3.3 Commands used for authentication and encryption with SSL

This section describes the syntax and functionality of the commands used for authentication and encryption with SSL.

3.3.1 hwscertutil cert -hash

Creating a hash link

Synopsis

```
hwscertutil cert -noout -hash -in CA-certificate-file
```

Storage location

Application Server installation directory/httpsd/sbin

Function

To perform a certificate validity check, specify the certificate of the certificate issuer CA in the SSLCACertificateFile directive or SSLCACertificatePath directive. In the SSLCACertificatePath directive, specify the directory that stores the symbolic link (hash link) with the hash value that points to the certificate of the certificate issuer CA.

The hash value is created by using the hwscertutil cert command.

If the SSLCACertificatePath directive is specified, the certificate search can be performed efficiently on the Web server by using the hash value. If there are many CA certificates, we recommend that you specify the SSLCACertificatePath directive rather than the SSLCACertificateFile directive. Note that one hash value must be assigned per certificate, so you cannot specify a file with multiple certificates when creating the hash link.

When generating the symbolic link in the hash link directory that is specified in the SSLCACertificatePath directive, you must add .0 to the hash value. Grant the read and execution permissions to the directory to be specified in the SSLCACertificatePath directive so that the user specified in the User and Group directives can access the directory.

Execution permission

Superuser

Arguments

-in *CA-certificate-file*

Specify the CA certificate file for which the hash link value is created.

Usage example

An example of the hash link directory and CA certificate for the following directory and file is given below:

/opt/hitachi/APServer/httpsd/conf/ssl/cacerts: Hash link directory

/opt/hitachi/APServer/httpsd/conf/ssl/cacert/cacert.pem: Certificate of the CA

```
cd /opt/hitachi/APServer/httpsd/conf/ssl/cacerts
ln -s /opt/hitachi/APServer/httpsd/conf/ssl/cacert/cacert.pem `hwscertutil cert
 -noout -hash -in /opt/hitachi/APServer/httpsd/conf/ssl/cacert/cacert.pem`.0
```

This creates the `xxxxxxxx.0` hash link for `/opt/hitachi/APServer/httpsd/conf/ssl/cacert/cacert.pem`.

Exit Status

Exit Status	Explanation
`0`	command executed successfully.
Other than `0`	error in executing the command.

3.3.2 hwscertutil cert -outform

The `hwscertutil cert -outform` command converts the certificate format.

Synopsis

```
hwscertutil cert -inform input-format -outform output-format
    -in input-file -out output-file
```

Storage location

Application Server installation directory/httpsd/sbin

Function

This subsection explains how to convert the certificate format. Use this functionality as necessary.

Execution permission

Superuser

Arguments

`-inform` *input-format*

Specifies the input format.

The following values can be specified:

- `DER`
- `PEM`

Default value: `PEM`

`-outform` *output-format*

Specifies the output format.

The following values can be specified:

- `DER`

- PEM

Default value: PEM

-in *input-file*

Specify the certificate file before conversion.

-out *output-file*

Specify the certificate file after conversion.

3.3.3 hwscertutil cert -text

Displaying certificate contents

Synopsis

```
hwscertutil cert -in certificate file -text
```

Storage location

Application Server installation directory/httpsd/sbin

Function

This subsection explains how to display the contents of a certificate file.

The following command displays the part of the certificate file that begins with -----BEGIN CERTIFICATE----- and ends with -----END CERTIFICATE----.

Execution permission

Superuser

Arguments

-in *certificate file*

Specify the certificate file to be displayed.

Examples

```
hwscertutil cert -in httpsd.pem -text
```

httpsd.pem: Certificate file to be displayed

Exit Status

Exit Status	Explanation
0	command executed successfully.
Other than 0	error in executing the command.

3.3.4 hwscertutil req

Displaying the contents of a Certificate Signing Request (CSR)

Synopsis

```
hwscertutil req -in CSR-file -text
```

Storage location

Application Server installation directory/httpsd/sbin

Function

This subsection explains how to display the contents of a Certificate Signing Request (CSR).

Execution permission

Superuser

Arguments

`-in` *CSR-file*

Specify the CSR file to be displayed.

Examples

```
hwscertutil req -in httpsd.csr -text
```

`httpsd.csr`: CSR file to be displayed

Exit Status

Exit Status	Explanation
0	command executed successfully.
Other than 0	error in executing the command.

3.3.5 hwscertutil reqgen

Creating a Certificate Signing Request (CSR)

Synopsis

```
hwscertutil reqgen [-sign {MD5|SHA1|SHA224|SHA256|SHA384|SHA512}]
    -key key-file -out CSR-file
```

Storage location

Application Server installation directory/httpsd/sbin

Function

This subsection describes how to use the `hwscertutil reqgen` command to create a Certificate Signing Request (CSR). The created CSR file is submitted to the CA, which then issues the signed certificate. The CSR is created in the format conforming to PKCS #10.

Execution permission

Superuser

Arguments

`-sign {MD5|SHA1|SHA224|SHA256|SHA384|SHA512}`

Specify the signature algorithm used when the CSR is created.

The following values can be specified:

- `-MD5` : `md5WithRSAEncryption` is used.
- `-SHA1`: `sha1WithRSAEncryption` is used.
- `-SHA224`: `sha224WithRSAEncryption` is used.
- `-SHA256`: `sha256WithRSAEncryption` is used.
- `-SHA384`: `sha384WithRSAEncryption` is used.
- `-SHA512`: `sha512WithRSAEncryption` is used.

Default value: `SHA1`

`-key` *`key-file`*

Specify the Web server private key file.

`-out` *`CSR-file`*

Specify the file to which the created CSR is output.

Examples

To create a Certificate Signing Request (CSR).

```
hwscertutil reqgen -sign SHA1 -key httpsdkey.pem -out httpsd.csr
```

`httpsdkey.pem`: Key file

`httpsd.csr`: CSR file

Exit Status

Exit Status	Explanation
`0`	command executed successfully.
Other than `0`	error in executing the command.

3.3.6 hwskeygen

The `hwskeygen` command creates a private key for the Web server

Synopsis

```
hwskeygen -rand file-name[:file-name]... [-des|-des3] -out key-file
    [-bits {512|1024|2048|4096}]
```

Storage location

Application Server installation directory/httpsd/sbin

Function

This subsection describes how to use the `hwskeygen` command to create a private key for the Web server. The created Web server private key file is specified in the `SSLCertificateKeyFile` directive.

Execution permission

Superuser

Arguments

`-rand` *file-name*`[:`*file-name*`]...`

Specify any file to be used for random number generation. You must specify an appropriate file whose size is large enough for the random number generation.

`-des|-des3`

Specify the encryption type when encrypting the private key. If you specify this parameter, you will be requested to enter a password when creating the private key. The password must be no more than 64 characters long. When creating the Certificate Signing Request (CSR) (`hwscertutil reqgen` command) and starting the Web server, you will also be requested to enter the password.

Note that you can skip the password entry for Web server startup. To skip the password entry, specify the `SSLCertificateKeyFile` directive in which you specified the server private key file and the `SSLCertificateKeyPassword` directive in which you specified the password file to `httpsd.conf`. You can create a password file by using the `sslpasswd` command.

If `-des` is specified, the Data Encryption Standard (DES) is selected as the encryption type. If `-des3` is specified, Triple DES is selected. This parameter does not affect the encryption type used in the communication between the Web server and the Web browser.

`-out` *key-file*

Specify the file to which the Web server private key is output.

`-bits {512|1024|2048|4096}`

Specify the bit length of the Web server private key.

The following values can be specified:

- `512`
- `1024`
- `2048`
- `4096`

Default value: `1024`

Examples

To create the `httpsdkey.pem` Web server private key:

```
hwskeygen -rand file1:file2:file3:file4:file5 -out httpsdkey.pem -bits 1024
```

`file1, file2, file3, file4, file5`: Arbitrary files

Exit Status

Exit Status	Explanation
`0`	command executed successfully.
Other than `0`	error in executing the command.

Related topics

- 3.3.5 hwscertutil reqgen
- 3.3.7 sslpasswd

3.3.7 sslpasswd

The `sslpasswd` command creates a password file for the password-protected server private key.

Synopsis

```
sslpasswd server-private-key-file-name password-file-name
```

Storage location

Application Server installation directory`/httpsd/sbin`

Function

Before using the password-protected server private key, you can save the password in a file and then set the directive, so that you do not have to enter the password when the Web server starts. The procedure is described below.

1. Use the `hwskeygen` command to create a password-protected server private key.
2. Use the `sslpasswd` command to create a password file.
3. In `httpsd.conf`, set both the `SSLCertificateKeyPassword` directive that specifies the created password file and the `SSLCertificateKeyFile` directive that specifies the server private file.
4. Start or restart the Web server.

Be careful not to reveal the contents of the password file. To prohibit access from other users, set directory permissions or file permissions for both the directory that stores the server private key and the directory that stores the password file.

Execution permission

Superuser

Arguments

server-private-key-file-name

Specify the password-protected server private key.

Password-file-name

Specify the name of the file that outputs the password file.

Examples

```
sslpasswd httpsdkey.pem .keypasswd
```

Exit Status

Exit Status	Explanation
0	The command terminated normally.
Other than 0	The command terminated abnormally.

Notes

- You cannot specify the name of an existing file as the name of the password file.
- In UNIX, you cannot use the password file created by using the Windows `sslpasswd` command.
- In Windows, you cannot use the password file created by using the UNIX `sslpasswd` command.

3.4 Operation by general user

This section describes the differences between the superuser and general users, and methods to create an environment for general users to operate Web server.

Overview

For Web server, normal operation is assumed to be operation by the superuser.

When Web server is installed, various settings are configured for operation by the superuser.

Thus, when users other than the superuser (hereafter referred to as general users) operate Web server, they need to change the settings file for Web server and settings in related directories and files. For some functionality in Web server, some operations are restricted from general users.

This section describes the differences between the superuser and general users, and methods to create an environment for general users to operate Web server.

Permissions for each process

The following table lists the permissions of each process for operation by the superuser or general users.

No.	Process	Operation by the superuser	Operation by general users
1	Control process	Superuser	General user
2	rotatelogs and rotatelogs2 processes		
3	Server process	Users or groups specified in the User and Group directives	
4	CGI process		
5	gcache server		

Differences between the superuser and general users in UNIX

In UNIX, unlike general users, the superuser has system administrator permissions. The following table lists examples of the differences between the superuser and general users in UNIX.

No.	Item	Superuser	General user
1	Can stop processes that were started by users?	Yes	No
2	Can open well-known ports (ports 1023 and lower)?	Yes	No
3	Can access files that do not explicitly have read or write permissions?	Yes	No

If a general user operates Web server, because the control process in Web server operates with general user permission, the behavior in this case might differ from operation by the superuser. Therefore, if a general user operates Web server, the user needs to create an environment while considering the differences with the superuser.

Changing resource owners and groups

In UNIX, you can change resource owners and groups for content and settings files for Web server, and for files and directories accessed by Web server during operation.

At the minimum, you will need to change the resources under the installation directory (*Application Server installation directory*/httpsd).

If you want to restore resource owners and groups to the previous settings, save the owners and groups for the current resources before making changes.

The superuser can save owners and groups. The following is an example of how to do this.

Example:

For the resources under the `/opt/hitachi/APServer/httpsd` directory, create a list of owners and groups.

```
ls laR /opt/hitachi/APServer/httpsd
```

The superuser can change owners and groups. The following is an example of how to do this.

Example:

For the resources under the `/opt/hitachi/APServer/httpsd` directory, change the owner (`hwsuser`) and the group (`hwsgroup`).

```
chown R hwsuser:hwsgroup /opt/hitachi/APServer/httpsd
```

Restrictions

The commands below cannot be operated by general users. Operate these commands as the superuser.

- hwscertutil
- htpasswd
- hwskeygen
- logresolve
- sslpasswd

In operation by general users, the following directives cannot be specified. Any directive specified by general users is ignored.

- Group
- User

In operation by general users, well-known ports (ports 1023 and lower) cannot be opened. Be careful when specifying the port number in the following directives:

- Listen
- SSLCacheServerPort

4 Commands used in the Java VM

This section describes the syntax and functionality of the commands used in the Java VM.

4.1 List of commands used in the Java VM

The following table lists the commands used in the Java VM.

Command used in system operation

Command name	Classification	Summary
car_tar_gz	Core archive functionality	Groups libraries relevant to the core file, etc., and outputs them to a single archive file. The file is compressed using the gzip command.
car_tar_Z	Core archive functionality	Groups libraries relevant to the core file, etc., and outputs them to a single archive file. The file is compressed using the compress command.
eheapprof	Output of an extended thread dump with Explicit heap detailed information.	Outputs, for the java process with the specified process ID, an extended thread dump that includes Explicit heap detailed information.
java_hras	Execution of Java application programs that do not work with Application Server	Enables proprietary functionality used for troubleshooting, and starts the Java VM.
javacore	Acquisition of a core file and thread dump	Simultaneously acquires the core file and thread dumps when the command is executed.
javagc	Forced generation of GC	Forces generation of full GC at any timing, for the java process with the specified process ID.
javatrace	Collection of trace information.	Collects stack trance information from a core file.
jheapprof	Output of an extended thread dump with class-based statistics.	Outputs, for the java process with the specified process ID, an extended thread dump that includes class-based statistics.
jheapprofanalyzer	Output of a class-based statistics analysis file to a CSV file.	Outputs a class-based statistics analysis file to a CSV file.

Important note

- In a command argument, specify a value that can be specified for the command. If you specify another value, operation is not guaranteed.
- Even if a command ends normally, an error might have occurred. Check the message log and stack trace log.

4.2 Commands used in system operation

This section describes the syntax and functionality of the commands used insystem operation.

4.2.1 car_tar_gz

core archiving function

Synopsis

```
car_tar_gz [-i executable-file-name] [-f] [-s|-S] [core-file-name] >
    output-file-name
```

Storage location

Application Server installation directory/jdk/jre/bin/

Function

Collects the core file and the related libraries and outputs to a single archive file. The car_tar_gz command compresses the file and the libraries, by using the gzip command.

Arguments

`-i` *executable-file-name*

Specifies an executable file name.

`-f`

Acquires files such as the libraries with a path name.

`-s`

Reports the size (unit: bytes) of an archive file to be output.

In such cases, the archive file is not output.

`-S`

Outputs a shell script for creating an archive file.

When you specify this option, specify the shell script name in *output-file-name*.

When you execute the output shell script, the archive file is output.

core-file-name

Specify the core file name.

Default value:`core`

output-file-name

Specify a name of the archive file or shell script file.

Examples

When creating an archive file

```
car_tar_gz -f core.8326 > corefile.tar.gz
```

When checking the size of an archive file

```
car_tar_gz -s core.8326
```

When creating an archive file from a shell script

1. Creating a shell script
   ```
   car_tar_gz -S core.8326 > collect_cores.csh
   ```
2. Assigning execution permission to a shell script
   ```
   chmod +x ./collect_cores.csh
   ```
3. Creating an archive file by using the shell script
   ```
   ./collect_cores.csh > corefile.tar.gz
   ```

Exit Status

Exit Status	Explanation
0	The command is terminated normally.
1	The command is terminated abnormally.

Output messages

When the system outputs the following messages, a normal archive file is not output.

No.	Error message	Explanation
1	`usage: car_tar_gz [ -i executable-file ] [ -f ] [ -s | -S ] [ core-file ]`	Arguments of the command are invalid.
2	`car_tar_gz: inner error!`	This is an internal error of the `car_tar_gz` command.
3	`car_tar_gz: cannot create temporary name`	A file to be temporarily used cannot be created. Delete or migrate the file `car_exec??` (`??` is a number from `00` through `99`) in the current directory.
4	`car_tar_gz: ~ : file not found`	A file is not found. Check the executable file specified in the core file and the `-i` option.
5	`car_tar_gz: ~ : not supported platform`	The OS is not supported.

Notes

- You must assign the write permission to the current directory.
- In the same directory, you cannot concurrently execute the `car_tar_Z` and `car_tar_gz` commands for the same core file or different core files having the same creation time.
- The `tar` and `gzip` commands must have been installed in the system.
- If you forcibly terminate the `car_tar_gz` command during execution, temporary files or the directory created by the `car_tar_gz` command remain behind in the current directory.

  Temporarily created files: `car_exec??` (`??` is a number from `00` thorough `99`)

  A temporarily created directory: `carYYMMDDhhmm/`(`YYMMDDhhmm` is the creation year, month, date, hours and minutes of the core file).

4.2.2 car_tar_Z

core archiving

Synopsis

```
car_tar_Z [-i executable-file-name] [-f] [-s|-S] [core-file-name] >
    output-file-name
```

Storage location

Application Server installation directory/jdk/jre/bin/

Function

This command gathers a core file and its related libraries and outputs them to a single archive file. The car_tar_Z command compresses the file and the related libraries by using the compress command.

Arguments

-i *executable-file-name*

Specifies an executable file name.

-f

Acquires files such as the libraries with a path name.

-s

Reports the size (unit: bytes) of an archive file to be output.

In such cases, the archive file is not output.

-S

Outputs a shell script for creating an archive file.

When you specify this option, specify the shell script name in *output-file-name*.

When you execute the output shell script, the archive file is output.

core-file-name

Specify the core file name.

Default value:core

output-file-name

Specify a name of the archive file or shell script file.

Examples

When creating an archive file

```
car_tar_Z -f core.8326 > corefile.tar.Z
```

When checking the size of an archive file

```
car_tar_Z -s core.8326
```

When creating an archive file from a shell script

1. Creating a shell script

```
car_tar_Z -S core.8326 > collect_cores.csh
```

2. Assigning execution permission to a shell script
```
chmod +x ./collect_cores.csh
```

3. Creating an archive file by using the shell script
```
./collect_cores.csh > corefile.tar.Z
```

Exit Status

Exit Status	Explanation
0	The command is terminated normally.
1	The command is terminated abnormally.

Output messages

When the system outputs the following messages, a normal archive file is not output.

No.	Error message	Explanation
1	`usage: car_tar_Z [ -i executable-file ] [ -f ] [ -s | -S ] [ core-file ]`	Arguments of the command are invalid.
2	`car_tar_Z: inner error!`	This is an internal error of the `car_tar_Z` command.
3	`car_tar_Z: cannot create temporary name`	A file to be temporarily used cannot be created. Delete or migrate the file `car_exec??` (`??` is a number from `00` through `99`) in the current directory.
4	`car_tar_Z: ~ : file not found`	A file is not found. Check the executable file specified in the core file and the `-i` option.
5	`car_tar_Z: ~ : not supported platform`	The OS is not supported.

Notes

- You must assign the write permission to the current directory.
- In the same directory, you cannot concurrently execute the `car_tar_Z` and `car_tar_gz` commands for the same core file or different core files having the same creation time.
- The `tar` and `compress` commands must have been installed in the system.
- If you forcibly terminate the `car_tar_Z` command during execution, temporary files or the directory created by the `car_tar_Z` command remain behind in the current directory.

  Temporarily created files: `car_exec??` (`??` is a number from `00` thorough `99`)

  A temporarily created directory: `carYYMMDDhhmm/`(`YYMMDDhhmm` is the creation year, month, date, hours and minutes of the core file).

4.2.3 eheapprof

Output of extended thread dump containing the Explicit heap detailed information

Synopsis

```
eheapprof [-i|-f] [-force] [-freeratio] -p process-ID
```

Storage location

Application Server installation directory/jdk/jre/bin/

Function

This command outputs the extended thread dump containing the Explicit heap detailed information for java processes of the process ID specified in arguments. You can also output the statistical information of an object within the Explicit memory block, and the release ratio information of the Explicit memory block to an extended thread dump.

Execution permission

Execute the command as the same user as the user of the java process specified by the process ID in the arguments.

Arguments

`-i`

Displays a message confirming the execution of the output processing for an extended thread dump containing the Explicit heap detailed information. You input either `y` or `n`. If you input `y`, the extended thread dump containing the Explicit heap detailed information will output. If you input `n`, no information will be displayed and the processing will terminate. Even if you omit this option, the option is valid as long as the `-f` option is not specified.

`-f`

Disables the `-i` option. If you omit this option, the `-i` option will become valid.

`-force`

Requests the output of an extended thread dump for the java process indicated by the process ID that is specified with the `-p` option, without confirming the existence of the /tmp/hsperfdata_*user-name*/*process-ID* file created in the java process.

`-freeratio`

Enables the output of the object release ratio information of the Explicit memory block and outputs an extended thread dump.

When you execute the `eheapprof` command by specifying this option, Java VM executes the following process and acquires the object release ratio information.

- The full garbage collection
- An Explicit memory block release process

The execution of these processes lead to a risk of stopping the execution of an application for a few seconds and hence we recommend that you output the object release ratio information of the Explicit memory block during system development or when operations are stopped.

`-p` *process-ID*

Specifies the process ID of the java program that outputs the extended thread dump containing the Explicit heap detailed information.

Type: Integer

The following values can be specified:

- `0` to `4294967295`

Output format

When the output of the object release ratio information of the Explicit memory block is disabled:

```
Explicit Heap Status
--------------------
 max EH_MAX, total EH_TOTAL, used EH_USED, garbage EH_GARB (EH_PER1
 used/max, EH_PER2 used/total, EH_PER3 garbage/used), EM_NUMS spaces exist

 Explicit Memories(EM_MGR_PTR)

 "EM_NAME" eid=EID(EM_PTR)/EM_TYPE, total EM_TOTAL, used EM_USED,
 garbage EM_GARB (EM_PER1 used/total, EM_PER2 garbage/used, FL_BLOCKS
 blocks) EM_STAT

    deployed objects
    _________________Size__Instances__Class_________________
                   ISIZE       INUM  CNAME
                     ...
                  AISIZE      AINUM total
```

When the output of the object release ratio information of the Explicit memory block is enabled:

```
Explicit Heap Status
--------------------
 max EH_MAX, total EH_TOTAL, used EH_USED, garbage EH_GARB (EH_PER1
 used/max, EH_PER2 used/total, EH_PER3 garbage/used), EM_NUMS spaces exist

 Explicit Memories(EM_MGR_PTR)

 "EM_NAME" eid=EID(EM_PTR)/EM_TYPE, total EM_TOTAL, used EM_USED,
 garbage EM_GARB (EM_PER1 used/total, EM_PER2 garbage/used, FL_BLOCKS
 blocks) EM_STAT

    deployed objects
    _________________Size__Instances__FreeRatio__Class_________________
                   ISIZE       INUM     FRATIO  CNAME
                     ...
                  AISIZE      AINUM total
```

The following explains each item in the output formt.

Classification	Item to be Output	Content of the Output	Meaning
Explicit heap information	*EH_MAX*	<const>K	The maximum Explicit heap size is output in kilobytes.
	EH_TOTAL	<const>K	The secured Explicit heap size is output in kilobytes.
	EH_USED	<const>K	The used Explicit heap size is output in kilobytes.
	EH_GARB	<const>K	The internal status of the Explicit heap is output.
	EH_PER1	<decimal>%	The usage rate of the Explicit heap (*EH_USED*/*EH_MAX*) is output in percentage.
	EH_PER2	<decimal>%	The usage rate of the Explicit heap (*EH_USED*/*EH_TOTAL*) is output in percentage.
	EH_PER3	<decimal>%	The internal status of the Explicit heap is output.
	EM_NUMS	<const>	The number of enabled Explicit memory blocks is output.
	EM_MGR_PTR	<ptr>	The memory address of the internal information for Explicit heap control is output. You can use this information in troubleshooting.

Classification	Item to be Output	Content of the Output	Meaning
Explicit memory block information	*EM_NAME*	<letters>	The name of the Explicit memory block is output. If the name of the Explicit memory block contains a multi-byte character, the content of the output is not fixed. (Usually, the character is corrupted in the output.) If the Explicit memory block information is output almost at the same time of initialization of the Explicit memory block, or if the Explicit memory block was internally generated by Java VM, `NULL` might be output.
	EID	<const>	The ID of the Explicit memory block is output.
	EM_PTR	<ptr>	The memory address of the internal structure of the Explicit memory block is output. You can use this information in troubleshooting.
	EM_TYPE	`R|B|A`	The Explicit type is output. `R` indicates an Explicit memory block used inside Application Server. `B` indicates an Explicit memory block used by an application. `A` indicates an Explicit memory block specified by using the automatic placement configuration file.
	EM_TOTAL	<const>K	The secured memory size of the Explicit memory block is output in kilobytes.
	EM_USED	<const>K	The used memory size of the Explicit memory block is output in kilobytes.
	EM_GARB	<const>K	The internal status of the Explicit memory block is output in kilobytes.
	EM_PER1	<decimal>%	The usage rate of the Explicit memory block (*EM_USED*/*EM_TOTAL*) is output in percentage.
	EM_PER2	<decimal>%	The internal status of the Explicit memory block is output.
	FL_BLOCKS	<const>	`0` is always output.
	EM_STAT	`Enable|Disable`	The sub-status of the Explicit memory block is output.
Object statistics[#1]	*ISIZE*	<const>	The size of an object instantiated from a class, in the Explicit memory block, is output.
	INUM	<const>	The number of objects instantiated from a class, in the Explicit memory block, is output.
	CNAME	<letters>	The complete class name of the class indicated by *ISIZE* and *INUM* is output.
	AISIZE	<const>	The total size of all objects in the Explicit memory block is output.
	AINUM	<const>	The number of all objects in the Explicit memory block is output.
Object release ratio information[#2]	*FRATIO*	<decimal>%	The ratio of objects released by the automatic release processing of the Explicit memory block (object release ratio) is output in percentage. Object release ratio = (number of objects of the class before automatic release processing - number of objects of the class after automatic release processing) / number of objects of the class before automatic release processing x 100 At the output of the object release ratio information, − is output for the Explicit memory block that was not targeted by automatic release processing.

(Legends)

<const>: indicates a natural number.

<decimal>: indicates a positive number (to one decimal place).

<ptr>: indicates a pointer value.

<letters>: indicates an arbitrary character or character string.

#1:

The object statistics are output when the `eheapprof` was executed. In the object statistics, the size and number information about objects more than the actually created objects might be output. In this case, the excess object information is replaced to `[I` that indicates an integer-type array. This `[I` indicates an object that is not used in the Explicit memory block. An object not used in the Explicit memory block is convert to an integer-type array by the Java VM internal processing.

#2:

The object release ration information is output when the `eheapprof` command with the `-freeratio` option was executed.

Output example

When output of object statistics and the object release ratio information of the Explicit memory block is disabled:

```
Explicit Heap Status
--------------------
 max 31415926K, total 213971K, used 205369K, garbage 1234K (1.1% used/max,
 96.2% used/total, 0.0% garbage/used), 3 spaces exist

 Explicit Memories(0x12345678)

 "EJBMgrData" eid=1(0x02f25610)/R, total 154272K, used 150176K, garbage 1234K
 (97.0% used/total, 1.2% garbage/used, 0 blocks) Enable

 "VJBStored" eid=3(0x02f25910)/B, total 54272K, used 50176K, garbage 0K (90.9%
 used/total, 0.0% garbage/used, 2 blocks) Enable

 "ExplicitMemory-2" eid=2(0x02f25700)/R, total 5427K, used 5017K, garbage 0K
 (92.1% used/total, 0.0% garbage/used, 0 blocks) Enable
```

When output of object statistics and the object release ratio information of the Explicit memory block is enabled:

```
Explicit Heap Status
--------------------
 max 31415926K, total 162816K, used 150528K, garbage 10004K (0.0% used/max,
 91.1% used/total, 6.6% garbage/used), 3 spaces exist

 Explicit Memories(0x12345678)

  "EJBMgrData" eid=1(0x02f25610)/R, total 54272K, used 50176K, garbage 0K
 (91.2% used/total, 0.0% garbage/used, 0 blocks)
    deployed objects
    ________________Size__Instances__FreeRatio__Class_________________
                35234568      10648          - java.util.HashMap
                 5678900      10668          - [Ljava.util.HashMap$Entry;
                 4456788       7436          - java.util.HashMap$Entry
                 4321000        200          - java.util.WeakHashMap
                 1234568        190          - [Ljava.util.WeakHashMap$Entry;
                  454400          4          - java.util.WeakHashMap$Entry
                51380224      29146 total

  "VJBStored" eid=3(0x02f25910)/B, total 54272K, used 50176K, garbage 10004K
 (90.7% used/total, 19.9% garbage/used, 5 blocks)
    deployed objects
    ________________Size__Instances__FreeRatio__Class_________________
                35234568      10648         49 java.util.HashMap
                 5678900      10668         43 [Ljava.util.HashMap$Entry;
                 4456788       7436         50 java.util.HashMap$Entry
                 4321000        200         32 java.util.WeakHashMap
```

```
                1234568        190         45 [Ljava.util.WeakHashMap$Entry;
                 454400          4         22 java.util.WeakHashMap$Entry
               51380224      29146 total

 "ExplicitMemory-2" eid=2(0x02f25700)/B, total 54272K, used 50176K,  garbage
0K (91.1% used/total, 0.0% garbage/used, 0 blocks)
   deployed objects
   _________________Size__Instances__FreeRatio__Class_________________
               35234568      10648          - java.util.HashMap
                5678900      10668          - [Ljava.util.HashMap$Entry;
                4456788       7436          - java.util.HashMap$Entry
                4321000        200          - java.util.WeakHashMap
                1234568        190          - [Ljava.util.WeakHashMap$Entry;
                 454400          4          - java.util.WeakHashMap$Entry
               51380224      29146 total
```

Exit Status

Exit Status	Explanation
0	Terminated normally.
1	Terminated abnormally.
2	There is no response indicating that the output processing has terminated within the fixed time for the extended thread dump containing the Explicit heap detailed information.

Output messages

If the following error messages or warning messages are issued, the extended thread dump containing the Explicit heap detailed information is not output:

No.	Error message	Explanation
1	`usage: eheapprof [-f|-i] [-force] [-freeratio] -p process-id`	The argument specified in the `eheapprof` command is wrong.
2	`eheapprof: illegal option -- option`	*option* specified in the `eheapprof` command is invalid.
3	*process-ID*`: Now processing previous request, this request canceled`	The process indicated by *process-ID* specified in the argument of the `eheapprof` command outputs the current Explicit heap detailed information.
4	*process-ID*`: No such process`	The process indicated by *process-ID* specified in the argument of the `eheapprof` command is not found or the process indicated by *process-ID* specified in the `eheapprof` command is not a java process.
5	*process-ID*`: Not owner`	The user is not the owner of the process of *process-ID* specified in the argument of the `eheapprof` command.
6	`eheapprof: can't create work file at /tmp , this request canceled`	If the user does not have permission to reference or write data into the directory for temporary files, the extended thread dump containing the Explicit heap detailed information cannot be output. The output request for the extended thread dump containing the Explicit heap detailed information is cancelled.
7	eheapprof: please delete *name-of-undeleted-file* in *full-path-of-undeleted-file*	The internal files could not be deleted when the `eheapprof` command ends. Delete the un-deleted files that exist in the full path of the un-deleted files.
8	`eheapprof: unexpected error occurred:` *error-cause*	An unexpected error occurred when executing the `eheapprof` command.

No.	Error message	Explanation

		error-cause might display the following:
		• When an attempt to secure the memory for the operation fails
		`malloc systemcall fail (errno=Y)`
		• When an attempt to close the object fails
		`close systemcall fail (errno=Y)`

9	*process-ID*`: Timeout occurred. Java process not responding`	The process indicated by *process-ID* specified in the argument of the `eheapprof` command did not return a response indicating that the output processing has terminated within the fixed time for the Explicit heap detailed information.

Notes

- The eheapprof command is provided as a utility for developing programs. Do not use this command in system operations.
- The `eheapprof` command cannot be executed concurrently for the same java process. You execute the command after the Explicit heap detailed information is output to the extended thread dump by the earlier `eheapprof` command.
- You use the `/tmp/hsperfdata_`*user-name*`/`*process-ID* file, when executing the `eheapprof` command. If the applicable file do not exist, the extended thread dump containing the Explicit heap detailed information cannot be output by the `eheapprof` command. However, if the `-force` option is specified, the output of an extended thread dump is requested for the java process indicated by the process ID that is specified in the `-p` option without confirming the existence of the `/tmp/hsperfdata_`*user-name*`/`*process-ID* file.

4.2.4 java_hras

The `java_hras` subcommand runs Java application programs that do not work with Application Server.

Synopsis

```
java_hras [options] class [argument...]
java_hras [options] -jar file.jar [argument...]
```

Storage location

Application Server installation directory`/jdk/jre/bin/`

Function

The `java_hras` subcommand starts the Java VM in a state in which the stand-alone Java program execution functionality is enabled, and then executes the Java application. If the stand-alone Java program execution functionality is enabled, the proprietary functionality required for troubleshooting is also enabled.

If the `java_hras` command is executed with the value `ON` or `OFF`, which are options for the proprietary functionality required for troubleshooting, that specified value takes priority. If the same option is specified more than once, then the last-specified option is used.

The following table lists the options that are enabled when the stand-alone Java program execution functionality is used.

No.	Category	Option	Java VM default value	When the stand-alone Java program execution functionality is enabled
1	Java VM log output	HitachiVerboseGC	OFF	ON
2		HitachiOutputMilliTime	OFF	ON
3		HitachiOutOfMemoryStackTrace	OFF	ON
4		HitachiJavaClassLibTrace	OFF	ON
5	Thread dumping output	HitachiLocalsInStackTrace	OFF	ON
6		HitachiLocalsSimpleFormat	OFF	ON
7		HitachiOutOfMemoryAbort	OFF	ON
8	Memory dump	HitachiFullCore	OFF	ON

When a Java command is executed, the following are different depending on whether the stand-alone Java program execution functionality is enabled.

- Stack trace

  When a Java program is executed, stack trace information that corresponds to the main thread of the Java program is different depending on whether the stand-alone Java program execution functionality is used.
- You cannot specify the `-Xhras` option for the `java_hras` subcommand.

Other than the functionality listed above, the specification for the `java` command is the same as when the stand-alone Java program execution functionality is not used.

Arguments

Options that can be specified are the same as the `java` command. However, the `-Xhras` option cannot be specified by itself.

`options`
Specifies the command-line options.

`class`
Specifies the name of the class to be invoked.

`-jar` *`file.jar`*
Specifies the name of the JAR file that will be called.

`argument`
Specifies the arguments passed to the main function.

Output example

Example 1

When the stack trace information of the main thread is obtained by thread-dump output.

- If the stand-alone Java program execution functionality is used:

```
main #1 prio=5 os_prio=0 jid=<N/A> tid=0x000000000230a800 nid=0x1988 waiting on
condition [0x000000000282e000..0x000000000282fbc0]
   java.lang.Thread.State: TIMED_WAITING (sleeping)

stack=[0x0000000002830000..0x0000000002734000..0x0000000002731000..0x0000000002730
000]
  [user cpu time=218ms, kernel cpu time=46ms] [blocked count=0, waited count=0]
    at java.lang.Thread.sleep(Native Method)
    at Hoge.main(Hoge.java:5)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:
62)
    at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:
43)
    at java.lang.reflect.Method.invoke(Method.java:483)
    at JP.co.Hitachi.soft.jvm.tools.Java_hras.execJava(Java_hras.java:52)
    at JP.co.Hitachi.soft.jvm.tools.Java_hras.main(Java_hras.java:39)
```

- If the stand-alone Java program execution functionality is not used:

```
"main" #1 prio=5 os_prio=0 jid=<N/A> tid=0x00000000002b7800 nid=0x1fa0 waiting on
condition [0x000000000278f000..0x000000000278f9c0]
   java.lang.Thread.State: TIMED_WAITING (sleeping)

stack=[0x0000000002790000..0x0000000002694000..0x0000000002691000..0x0000000002690
000]
  [user cpu time=171ms, kernel cpu time=46ms] [blocked count=0, waited count=0]
    at java.lang.Thread.sleep(Native Method)
    at Hoge.main(Hoge.java:5)
```

Example 2

While executing the `main` method of the main class that was specified during startup, a `java.lang.Exception` exception error occurred which could not be caught in the `main` method.

- If the stand-alone Java program execution functionality is used:

```
Exception in thread "main" java.lang.reflect.InvocationTargetException
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.
java:57)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAcces
sorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:606)
    at JP.co.Hitachi.soft.jvm.tools.Java_hras.execJava(Java_hras.java:51)
    at JP.co.Hitachi.soft.jvm.tools.Java_hras.main(Java_hras.java:38)
Caused by: java.lang.Exception: java.lang.Throwable
    at Test.main(Test.java:6)
    ... 6 more
Caused by: java.lang.Throwable
    ... 7 more
```

- If the stand-alone Java program execution functionality is not used:

```
Exception in thread "main" java.lang.Exception: java.lang.Throwable
    at Test.main(Test.java:6)
Caused by: java.lang.Throwable
    ... 1 more
```

Exit Status

Same as the `java` command.

Notes

To run the `java` command, use the following methods:

- Use the *Application Server installation directory*`/jdk/jre/bin/java_hras` command.
- Use the `java` command in the specified `-Xhras` option.
- When executing the `java` command, set the `JAVA_HRAS` environment variable.

When the `java` command is executed by other methods, the command execution fails and the following message is displayed:

```
Java command cannot be used except in some special cases. When starting a Java
program, use the java_hras command, etc.

Depending on the Java program being run, it might be necessary to purchase the
other product.
```

To run the `javaw` subcommand, use the following methods:

- Use the `javaw` subcommand in the specified `-Xhras` option.
- When executing the `javaw` command, set the `JAVAW_HRAS` environment variable.

When the `javaw` command is executed by other methods, the command execution fails and the following message is displayed:

```
Javaw command cannot be used except in some special cases. When starting a
Java program, use the java_hras command, etc.

Depending on the Java program being run, it might be necessary to purchase the
other product.
```

4.2.5 javacore

Acquiring the core file and thread dump

Synopsis

```
javacore [-i|-f] [-force] -p process-ID
```

Storage location

Application Server installation directory`/jdk/jre/bin/`

Function

This command collects a core file and a thread dump simultaneously during execution.

Execution permission

Execute the command as the same user as the user of the java process specified by the process ID in the arguments.

Arguments

`-i`

Displays the messages for checking the execution of the core file and the thread dump output process. Enter `y` or `n` for the displayed messages. In such cases, when you enter `y`, the thread dump is output and when you enter `n`, the process ends without performing any operation. When omitted, this option is valid as long as the `-f` option is not specified.

`-f`

Invalidates the `-i` option. If omitted, the `-i` option is enabled.

`-force`

Outputs core for the java process with the process ID specified in the `-p` option without confirming the existence of the `/tmp/hsperfdata_`*`user-name`*`/`*`process-ID`* file created with the java process.

`-p` *`process-id`*

Specifies the process-ID of the Java program from which the core file and thread dump are to be collected.

Type: Integer

The following values can be specified:

- `0` to `4294967295`

Output format

Header

```
EEE MMM dd hh:mm:ss yyyy#

Full thread dump Java HotSpot(TM) Client VM (1.5.0_05-b05-CDK0850-build-date
 mixed mode)
invoke-command-line
...
```

#

`EEE` represents a day, *`MMM`* a month, and *`dd`* a date. *`hh`* represents hours, *`mm`* minutes, *`ss`* seconds, and *`yyyy`* represents a year.

System settings

```
System Properties
-----------------
Java Home Dir   : installation-directory-of-JDK-execution-environment
Java DLL Dir    : installation-directory-of-library-configuring-JDK
Sys Classpath   : system-class-path
User Args       :
Java-command-option-1
Java-command-option-2
...
```

Operating environment

```
Operating Environment
---------------------
Host     : host-name:IP-address
OS       : OS-version
```

```
CPU     : CPU-type, number-of-available-CPUs/number-of-CPUs-of-all-systems

Resource Limits -
         RLIMIT_CPU     :number-of-seconds-available-for-the-process
         RLIMIT_FSIZE   :maximum-file-size-(-unit-:-byte-)
         RLIMIT_DATA    :mallocable-size-(-unit-:-byte-)
         RLIMIT_STACK   :maximum-stack-size-(-unit-:-byte-)
         RLIMIT_CORE    :maximum-core-size-(-unit-:-byte-)
         RLIMIT_RSS     :process-residence-size-(-unit-:-byte-)
         RLIMIT_AS      :available-memory-for-total-processes-(-unit-:-byte-)
         RLIMIT_NOFILE  :maximum-file-descriptor-value
```

Java heap information

In the case of SerialGC:

```
Heap Status
-----------
def new generation   max max size, total capacity, used size (max usage% used/
max, total usage% used/total)
                       [bottom, commit addr, reserve addr)
  eden space capacity,  usage% used [bottom, top, reserve addr)
  from space capacity,   usage% used [bottom, top, reserve addr)
  to   space capacity,   usage% used [bottom, top, reserve addr)
tenured generation   max max size, total capacity, used size (max usage% used/
max, total usage% used/total)
                       [bottom, commit addr, reserve addr)
   the space capacity,  usage% used [bottom, top, used block, reserve addr)
Metaspace       max max size, capacity capacity words, committed committed size,
reserved reserve size, used size (max usage% used/max, total usage% used/
committed)
  class space    max max size, capacity capacity words, committed committed size,
reserved reserve size, used size (max usage% used/max, total usage% used/
committed)
                 [bottom, top, commit addr, reserve addr)
```

max size

Indicates the maximum size in kilobytes.

In case of Metaspace, when `-XX:MaxMetaspaceSize` is not specified, the maximum size is not limited, and `unlimited` is output.

capacity

Indicates the current capacity in kilobytes.

capacity words

Indicates the total size of the memory area committed by Metaspace excluding free area. This is shown in kilobytes.

committed size

Indicates the total size of the memory committed by Metaspace in kilobytes.

reserve size

Indicates the size of reserved memory in kilobytes.

size

Indicates the size of used memory in kilobytes.

max usage

Indicates the usage to the maximum capacity.

In case of Metaspace, when `-XX:MaxMetaspaceSize` is not specified, the maximum size is not limited, and `-%` is output.

total usage

Indicates the usage rate to the current capacity.

bottom

Indicates the initial address of the area.

top

This indicate the initial address of the used area.

commit addr

Indicates the last address of the committed area.

reserve addr

Indicates the last address of the reserved area.

usage

Indicates the usage rate.

used block

Indicates the initial address of the next empty block.

Note

"class space" is output only when the compressed-object pointer functionality is enabled in a 64-bit Java VM environment and "Compressed Class Space" is used.

The value of "Metaspace" is the total value of whole "Metaspace" including "class space".

In case of G1GC

```
Heap Status
-----------
 garbage-first heap   total heap capacity, used heap size [heap bottom, heap
commit addr, heap reserve addr)
  region size region size, young region young (young size), survivor region
survivors (survivor size)
 Metaspace       max max size, capacity capacity words, committed committed size,
reserved reserve size, used size (max usage% used/max, total usage% used/
committed)
  class space    max max size, capacity capacity words, committed committed size,
reserved reserve size, used size (max usage% used/max, total usage% used/
committed)
                 [bottom, top, commit addr, reserve addr)
```

Detailed format of the items to be output

Item to be Output	Content of the Output	Meaning
heap capacity	<const>K	Indicates the size of the Java heap area in kilobytes. Total size of (a) + total size of (b) + total size of (c) + total size of (d)
heap size	<const>K	Indicates the size of used Java heap area in kilobytes. Total size of (a) + total size of (b) + total size of (c)
heap bottom	<ptr64>	Indicates the initial address of the Java heap area.
heap commit addr	<ptr64>	Indicates the last address of the committed Java heap area.
heap reserve addr	<ptr64>	Indicates the last address of the reserved Java heap area.
region size	<const>K	Indicates the size of one region in kilobytes.
young region	<const>	Indicates the number of used regions in the New area. Number of regions of (a) + number of regions of (b)

Item to be Output	Content of the Output	Meaning
young size	<const>K	Indicates the size of the used New area in kilobytes. (number of regions of (a) + number of regions of (b)) * "region_size"
survivor region	<const>	Indicates the number of used regions in the Survivor area. Number of regions of (b)
survivor size	<const>K	Indicates the size of the used Survivor area in kilobytes. Number of regions of (b) * "region_size"

The items output for "Metaspace" and "class space" are the same as that in case of Serial GC. See "In case of Serial GC".

(Legends)

(a): Eden area

(b): Survivor area

(c): Tenured area

(d): Free area

Internal memory map information for Java VM

```
JVM Internal Memory Map
-----------------------
memory-secure-function:address = start-address - end-address (size:size)
```

Note:

Memory-secure-function: Either `mmap()` or `malloc()` is output.

start-address: Start address of the memory area is output as hexadecimal.

end-address: End address of the memory area is output as hexadecimal.

size: Size of the secured memory area is output (Units: Bytes).

Internal memory size information for Java VM

```
JVM Internal Memory Status
--------------------------
  Heap Size      :secured-memory-size#
  Alloc Size     :used-memory-size#
  Free Size      :free-memory-size#
```

#

Unit: Bytes

Application environment

```
Application Environment
-----------------------
Signal Handlers -#1
  SIGHUP         :signal-handler-information
  SIGINT         :signal-handler-information
...
  SIGSOUND       :signal-handler-information
  SIGSAK         :signal-handler-information

Signal Handlers -
signal-type: [signal-handler-address], sa_mask[0]= signal-mask,
 sa_flags= specific-flag
...

Environment Variables -#2
environment-variable=value
```

```
...
Current Directory -
  Application Server installation directory/CC/server/...
```

#1

The following information is displayed:

- If the signal handler is installed, its address is displayed.

- In the case of `SIG_DFL`, `default` is displayed.

- In the case of `SIG_IGN`, `ignored` is displayed.

#2

The following information is displayed:

- The signal name defined in `/usr/include/sys/signal.h` is displayed in signal type.

- The address of the signal handler is output as hexadecimal in the "signal handler address". The address may be displayed in the form of "Library name + Offset."

- The `sa_mask` field value of the structure fetched by `sigaction()` is output as hexadecimal in the "signal mask".

- The `sa_flags` field value of the structure fetched by `sigaction()` is output as hexadecimal in the "specific flag".

Library information (in Linux)

```
Loaded Libraries
----------------
Dynamic libraries :
start-address-end-address   command
start-address-end-address   library
...
```

Thread information

```
-----------
"thread-name" daemon prio= priority jid=hash-value tid= thread-ID nid= nativeID
 status [start-address...end-address]
java.lang.Thread.State: current-status-of-thread#
 stack=
[stack-start-address..YellowPage-address..RedPage-address..stack-end-address]
 [user cpu time=user-time ms, kernel cpu time=kernel-time ms]
 [blocked count=block-count, waited count=standby-count]
 at class-name.method-name(method-information)
...
```

#

Information of *current-status-of-thread* is output only if JDK 6 is the base and the version is 08-10 or later.

The output contents are as follows:

thread-name

The thread name specified in the constructor of the Thread class is output.

daemon: In the case of the daemon thread, thread name is output as "`daemon`".

priority: The priority set in the Thread#setPriority is output.

hash-value: The value same as the one obtained by invoking System.identityHashCode() is output as 8-digit hexadecimal.

thread-ID: Memory address of thread object.

nativeID: Thread ID of the OS level.

status: Thread state.

runnable	Running or executable thread in `Object.wait()`, waiting for monitor entry, or waiting on
condition	Thread waiting for monitor lock
sleeping	Thread in the suspended state

start-address: The top-level stack address of the Java frame is output as hexadecimal.

end-address: The top-level stack address with JavaLock is output as hexadecimal.

*user-time*The user time since the thread is started is output in milliseconds.

*kernel-time*The kernel time since the thread is started is output in milliseconds.

block-count: The number of times the process is blocked since the thread is started is output.

standby- count: The number of times the process is pending since the thread is started is output.

current-status-of-thread

The message indicating the current status of thread is output. The contents of the message correspond to the java.lang.Thread.State enumerated type.

stack-start- address

The stack start address is output in hexadecimals.

YellowPage-address

The first address of the stack Yellow guard page is output in hexadecimals.

RedPage-address

The first address of the stack Red guard page is output in hexadecimals.

stack-end-address

The stack end address is output in hexadecimals.

class-name

The class name is output.

method-name

The method name is output.

method-information

The following method information is output:

Native Method	The information is output in the case of the native method.
File name: Line number	The information is output when the Java method is compiled with the line number.
Unknown Source	The information is output when the Java method is compiled without the line number.

Java monitor dump

```
Java monitor
------------
lock-object@hash-code owner-information
 standby-state:standby-thread-number
 standby-thread-information
```

The output contents are explained below:

lock-object

The class name of the object to be locked is output.

hash-code

The hash code to be obtained with Object.hashCode is output.

owner-information

owner "*thread-name*"*thread-ID*: This is output when the monitor has an owner.

`owner "`*thread-name*`"`*thread-ID*: When the monitor has an owner, `owner "`*thread-name*`"`*thread-ID* is output.

`no owner`: When the monitor does not have an owner, "`no owner`" is output.

standby-state

`... waiting to enter`: This state is displayed when the monitor is waiting for method execution.

`... waiting to be notified`: This state is displayed when the monitor is waiting for notification.

standby-thread-count

The thread count is output.

standby-thread-information

The information is output in the form of "*thread-name*" *thread-ID*.

Information on the number of JNI global references

```
JNI Information
---------------
JNI global references: JNI-global-reference-count
```

The output contents are explained below:

JNI global reference count

The number of global references maintained by Java VM are output.

Note:

Since the JNI global reference is reused even in Java VM, the numeric value does not reduce even after issuing the `DeleteGlobalRef` function supported by JNI to delete the JNI global reference. Even if the `NewGlobalRef` function is issued to create a new JNI global reference, the numeric value does not increase if the JNI global reference reused by Java VM is allocated.

Explicit heap information and Explicit memory block information

```
Explicit Heap Status
--------------------
 max EH_MAX, total EH_TOTAL, used EH_USED, garbage EH_GARB (EH_PER1
 used/max, EH_PER2 used/total, EH_PER3 garbage/used), EM_NUMS spaces exist

 Explicit Memories(EM_MGR_PTR)

  "EM_NAME" eid=EID(EM_PTR)/EM_TYPE, total EM_TOTAL, used EM_USED,
 garbage EM_GARB (EM_PER1 used/total, EM_PER2 garbage/used, FL_BLOCKS
 blocks) EM_STAT
...
```

Note the following:

- A blank line exists between the Explicit heap information and Explicit memory block information.
- The output order for the Explicit memory block information (which Explicit memory block will output) is not defined.
- Two one-byte character spaces exist before *EM_NAME*.
- A blank line exist as the last line. As a result, a blank line exist between the output of each Explicit memory block.

Footer

```
Full thread dump completed.   EEE MMM dd hh:mm:ss yyyy#
```

#

EEE represents a day, *MMM* a month, and *dd* a date. *hh* represents hours, *mm* minutes, *ss* seconds, and *yyyy* represents a year.

The following compares standard thread dump information with extended thread dump information:

Output information	Standard thread dump	Extended thread dump
Header	N	Y
System settings	N	Y
Operation environment	N	Y
Memory information (in Windows only)	N	Y
Java heap information	N	Y
Internal memory map information for Java VM	N	Y
Internal memory size information for Java VM	N	Y
Application environment	N	Y
Library information	N	Y
Thread information	Y	Y#1
Java monitor dump	N	Y
Footer	N	Y
Thread dump output destination	Standard output	Standard output#2 Java VM log fi

Legend:

Y: Information is output.

N: Information is not output.

#1

Information such as the start and end address of stack is output.

#2

The information is output when the `-XX:+HitachiThreadDumpToStdout` option is specified.

Examples

1. Execution of the `javacore` command, with the `-f` option omitted:
   ```
   javacore -p 8326
   ```
2. A message for confirming the execution of the core file and the thread dump output process is displayed:
   ```
   send SIGQUIT to 8326:?(y/n)
   ```
3. Enter `y` to collect the core file and thread dump (otherwise enter `n`):
   ```
   send SIGQUIT to 8326:?(y/n)y
   ```
4. When the core file and thread dump are collected, the Java program being executed outputs the following message:

```
Now generating core file (javacore8662.030806215140.core)...
done
(thread-dump-output)
Writing Java core  to javacore8662.030806215140.txt... OK
```

5. The Java program being executed creates the following files under the current directory and continues the program:
   - Core file
     javacore*process-id*.*time*.core
   - Thread dump
     javacore*process-id*.*time*.txt

Exit Status

Exit Status	Explanation
0	The command terminated normally.
1	The command terminated abnormally.
2	The core generation process did not return a response within the specified timeout time.

Output messages

If any of the error messages described below or a warning message is issued, the core file and thread dump have not been collected.

No.	Error message	Explanation
1	usage: javacore [-f|-i] [-force] -p process-id	A command argument is invalid.
2	javacore: can't create work file at /tmp, this request canceled	The user does not have permission to reference or write data into /tmp.
3	javacore: illegal option --*option*	The indicated option specified in the command arguments is invalid.
4	javacore: unexpected error occurred:*error-cause*	An unexpected error occurred during command execution.
5	javacore: please delete *name-of-undeleted-file* in *full-path-of-undeleted-file*	When the command terminated, a file created by the internal process of the javacore command could not be deleted. Delete the non-deleted file in the *full-path-of-the-non-deleted-file*.
6	*process-id*: No such process	No process is found that matches the process indicated by process-id specified in the argument of the javacore command or the process indicated by process-id specified in the javacore command is not the java process.
7	*process-id*: Not owner	The executing user is not the owner of the process indicated by process-id specified in the command.
8	*process-id*: Now processing previous request, this request canceled	The process indicated by process-id specified in the command is currently generating a core.
9	*process-id*: Timeout occurred. Java process not responding.	The process indicated by process-id specified in the argument of the command does not return a response indicating that the core output processing terminated within the fixed time.

Notes

- The `javacore` command sends the `SIGQUIT` signal to the specified process. If a program other than a Java program is specified by mistake, other programs may stop.
- The `javacore` command cannot be executed concurrently for the same Java process. You execute the command after the core output processing by the previous `javacore` command has terminated.
- If a Java VM process has not responded to a thread dump collection request and has gone onto no-response status, you cannot use the `javacore` command to collect the core file. In such a case, execute the `kill -6` command to terminate the Java VM process forcibly, then collect the core file.
- In Linux, generate core with the `gcore` command of gdb. If gdb is not installed, the following error message will be output as the standard output:
  `Error occurred in generating core file, gdb not found.`
  Furthermore, if the installed gdb is from an old version, the following message will be output as the standard output:
  `Error occurred in generating core file, gdb version 5.2 or later needed.`
- The `javacore` command uses the `/tmp/hsperfdata_`*`user-name`*`/`*`process-ID`* file at runtime. If the applicable file does not exist, the core file cannot be output by the `javacore` command.
  However, if the `-force` option is specified, core is output for the Java process indicated by the process ID that is specified in the `-p` option without confirming the existence of the `/tmp/hsperfdata_`*`user-name`*`/`*`process-ID`* file.

4.2.6 javagc

forced execution of garbage collection

Synopsis

```
javagc [-i|-f] [-v] [-s] [-force] [-ehgc] -p process-ID
```

Storage location

`Application Server installation directory``/jdk/jre/bin/`

Function

This command executes full garbage collection at a desired time for the Java process whose process ID is *`process-ID`*, such as when a memory leak or system error has occurred, or when application debugging is to be performed. Also, executes the full garbage collection and the Explicit memory block release at any time, for the Java process with the specified *`process- ID`*.

The `SIGQUIT` signal is used for communicating with a Java process. When the command is executed, requests user to check the process contents of the command. When the response is `n` (do not generate, execute, or send), the command does not execute the process contents of the command (the return value is `1`). You can omit this configuration step by specifying the `-f` option.

If a copy garbage collection or a full garbage collection that occurs because of normal causes is running in the java process indicated by *`process-ID`*, you wait for the garbage collection to end, and then execute the command.

Execution permission

Execute the command as the same user as the user of the java process specified by the process ID in the arguments.

Arguments

`-i`

Does not ask the user to check the process contents of the following command:

- Whether to execute the Explicit memory block release for the process given in *process-ID*
- Whether to send the `SIGQUIT` signal for generating the garbage collection

A specification of the `-f` option before the `-i` option is ignored.

`-f`

Does not ask the user to check the process contents of the following command:

- Whether to execute the Explicit memory block release
- Whether to send the `SIGQUIT` signal for generating the garbage collection

A specification of the `-i` option before the `-f` option is ignored.

`-v`

If the `-XX:+HitachiVerboseGC` option is not specified, creates a Java VM log file according to the Java VM log file creation rules and outputs the extended verbosegc information.

In such cases, output the extended verbosegc information of the contents, also with the effect of the following option values:

- `-XX:+HitachiVerboseGCPrintDate`
- `-XX:+HitachiVerboseGCPrintCause`
- `-XX:+HitachiVerboseGCCpuTime`
- `-XX:+HitachiCommaVerboseGC`

`-s`

Outputs the extended verbosegc information to the standard output.

In such cases, `-s` outputs the extended verbosegc information of the contents in which the following option values are also reflected:

- `-XX:+HitachiVerboseGCPrintDate`
- `-XX:+HitachiVerboseGCPrintCause`
- `-XX:+HitachiVerboseGCCpuTime`
- `-XX:+HitachiCommaVerboseGC`

`-force`

Executes the garbage collection or the Explicit memory block release for the java process indicated by the process ID that is specified in the `-p` option, without confirming the existence of the `/tmp/hsperfdata_`*user-name*`/`*process-ID* file created in the java process.

`-ehgc`

Executes the full garbage collection and the Explicit memory block release for the Java process with the specified *process-ID*.

You can explicitly release the Explicit memory blocks, which were not released by release processing when the automatic release functionality is enabled.

When you execute the `javagc` command, Java VM executes a full garbage collection and outputs the `EMJavaGC` command as the cause of the garbage collection in extended the verbosegc information. After that, the following Explicit memory blocks are released:

- Explicit memory blocks that are reserved by explicit release reservation, when automatic release functionality of the Explicit Memory Management functionality is enabled
- Explicit memory blocks generated by the explicit management heap automatic placement configuration file or Java VM
- Explicit memory blocks that were not released in the previous release processing

The release processing is not executed in the following cases:

- When you try to release Explicit memory blocks exceeding the maximum limit
  This refers to the case when the number of existing Explicit memory blocks is `1,048,575`.
- When the Explicit Memory Management functionality is `OFF`
  This refers to the case when `-XX:-HitachiUseExplicitMemory` option is specified.

In this case, although the constructor is successfully executed, memory blocks are handled as invalid Explicit memory blocks (ExplicitMemory instances).

With release processing of Explicit memory blocks by using the javagc command, a full garbage collection is executed. Hence, it is not appropriate for the processing related to running applications. We recommend that you execute release processing when the application is not running, such as at the time of undeploying and at night time.

`-p` *`process-id`*

Specifies a process ID for which a full garbage collection or the Explicit memory block release is to be executed.

Type: Integer

The following values can be specified:

- `0` to `4294967295`

Output format (In the case of SerialGC)

If the option `v` or `s` is specified, the Java VM log is output."

```
[id] <date> (Skip Full:full_count, Copy:copy_count) [gc_kind gc_info, gc_time
secs][Eden: eden_info][Survivor: survivor_info][Tenured: tenured_info]
[Metaspace: Metaspace_info][class space: class_space_info] [cause:cause_info]
[User: user_cpu secs][Sys: system_cpu secs][IM: jvm_alloc_size,
mmap_total_size, malloc_total_size][TC: thread_count][DOE: doe_alloc_size,
called_count][CCI: cc_used_sizeK, cc_max_sizeK, cc_infoK]
```

The following is a description of the output contents:

id

Identifier of the JavaVM log file.

date

Indicates the date and time for starting garbage collection.

When the `-XX:-HitachiVerboseGCPrintDate` option is specified, the date is not output.

full_count

Indicates the number of times the output of full GC information is skipped.

This is output when the `-XX:HitachiVerboseGCIntervalTime` option is specified.

copy_count

Indicates the number of times the output of copy GC information is skipped.

This is output when the `-XX:HitachiVerboseGCIntervalTime` option is specified.

gc_kind

Indicates the garbage collection type. "`FullGC`" or "`GC`" is output.

gc_info

Indicates the garbage collection information. The information is output in the following format:

area-length-before-the-garbage-collection -> area-length-after-the-garbage-collection (*area-size*)

gc_time

Indicates the elapsed time for garbage collection.

Eden

Indicates the type of eden. "`DefNew::Eden`" is output.

eden_info

Indicates the Eden information. The information is output in the following format:

area-length-before-the-garbage-collection -> area-length-after-the-garbage-collection (*area-size*)

Survivor

Indicates the survivor type. "`DefNew:: Survivor`" is output.

survivor_info

Indicates the Survivor information. The information is output in the following format:

area-length-before-the-garbage-collection -> area-length-after-the-garbage-collection (*area-size*)

Tenured

Indicates the type of Tenured. "`Tenured`" is output.

tenured_info

Indicates the Tenured information. The information is output in the following format:

area-length-before-the-garbage-collection -> area-length-after-the-garbage-collection (*area-size*)

Metaspace_info

Indicates memory information of the Metaspace area. This is output in the following format, in kilobytes:

<const1>K(<const2>K, <const3>K)-><const4>K(<const5>K, <const6>K)]

- <const1>: The size of the used Metaspace area before GC
- <const2>: The size of the capacity of the Metaspace area before GC
- <const3>: The size of the committed Metaspace area before GC
- <const4>: The size of the used Metaspace area after GC
- <const5>: The size of the capacity of the Metaspace area after GC
- <const6>: The size of the committed Metaspace area after GC

class_space_info

Indicates memory information of the "class space" area. This is output in the format below, in kilobytes.

If the compressed-object pointer functionality is disabled, this information is not output.

<const1>K(<const2>K, <const3>K)-><const4>K(<const5>K, <const6>K)]

- <const1>: The size of the used "class space" area before GC
- <const2>: The size of the capacity of the "class space" area before GC
- <const3>: The size of the committed "class space" area before GC
- <const4>: The size of the used "class space" area after GC
- <const5>: The size of the capacity of the "class space" area after GC
- <const6>: The size of the committed "class space" area after GC

cause_info

Indicates the cause of garbage collection.

When the `-XX:-HitachiVerboseGCPrintCause` option is specified, this information is not output.

user_cpu

Indicates the CPU time that the garbage collection thread has consumed in the user mode. The unit is in seconds.

If an attempt to obtain the CPU time fails, `unknown` is displayed as in the case of [`User: unknown`].

When the `-XX:-HitachiVerboseGCCpuTime` option is specified, the information is not output.

system_cpu

Indicates the CPU time that the garbage collection thread has consumed in the kernel mode. The unit is in seconds.

If an attempt to obtain the CPU time, `unknown` is displayed as in the case of [`Sys: unknown`].

When the `-XX:-HitachiVerboseGCCpuTime` option is specified, the information is not output.

jvm_alloc_size

Specifies the size of the area currently in use, from the areas being managed in Java VM (size of the area currently in use, from the total size of *`mmap_total_size`* and *`malloc_total_size`*).

When the `-XX:HitachiVerboseGCPrintJVMInternalMemory` option is specified, this information is not output.

mmap_total_size

Specifies the total C heap size allocated for mmap (VirtualAlloc in Windows), from the areas being managed in Java VM.

When the `-XX:-HitachiVerboseGCPrintJVMInternalMemory` option is specified, this information is not output.

malloc_total_size

Specifies the total C heap size allocated for `malloc`, from the areas being managed in Java VM.

When the `-XX:-HitachiVerboseGCPrintJVMInternalMemory` option is specified, this information is not output.

thread_count

Specifies the number of Java threads.

When the `-XX:-HitachiVerboseGCPrintThreadCount` option is specified, this information is not output.

doe_alloc_size

Specifies the cumulative heap size allocated by invoking the `java.io.File.deleteOnExit()method`.

When the `-XX:-HitachiVerboseGCPrintDeleteOnExit` option is specified, this information is not output.

called_count

Specifies the invocation count of the `java.io.File.deleteOnExit()` method.

When the `-XX:-HitachiVerboseGCPrintDeleteOnExit` option is specified, this information is not output.

cc_used_size

Specifies the size of the code cache area used when the garbage collection occurs. The unit is kilobyte.

When the `-XX:-PrintCodeCacheInfo` option is specified, this information is not output.

cc_max_size

Specifies the maximum size of the code cache area. The unit is kilobyte.

When the `-XX:-PrintCodeCacheInfo` option is specified, this information is not output.

cc_info

Specifies the maintenance information.

When the `-XX:-PrintCodeCacheInfo` option is specified, this information is not output.

Output format (In the case of G1GC)

If the option `v` or `s` is specified, the Java VM log is output."

In G1GC, the log relating to GC (VG1 log hereafter) and the log relating to Concurrent Marking (CM) (VCM log hereafter) are output to the Java VM log file. CM and applications are processed in parallel, so one CM log is output into multiple lines.

The following shows the details of the output formats of the VG1 log and VCM log:

VG1 log

`[`*id*`]<`*date*`>[`*gc_kind gc_info*`, `*gc_time* `secs][Status:`*gc_status*`][G1GC::Eden:` *eden_info*`][G1GC::Survivor:` *survivor_info*`][G1GC::Tenured:` *tenured_info*`] [G1GC::Humongous:` *humongous_info*`][G1GC::Free:` *free_info*`][Metaspace:` *Metaspace_info*`][class space:` *class_space_info*`] [cause:`*cause_info*`][RegionSize:` *region_size*`K][Target:` *target_time* `secs][Predicted:` *predicted_time* `secs] [TargetTenured:` *target_size*`K][Reclaimable:` *reclaimable_info*`][User:` *user_cpu* `secs][Sys:` *system_cpu* `secs][IM:` *jvm_alloc_size*`K, `*mmap_total_size*`K, `*malloc_total_size*`K][TC:` *thread_count*`][DOE:` *doe_alloc_size*`K, `*called_count*`][CCI:` *cc_used_size*`K, `*cc_max_size*`K, `*cc_info*`K]`

Note that there are no line break and space between items.

Item to be Output	Content of the Output	Meaning
id	VG1	Indicates the identifier of the Java VM log file. `[VG1]` is output in the GC log of G1GC.
date	<letters>	Indicates the date and time when GC or CM was started. This item is not output if `-XX:-HitachiVerboseGCPrintDate` is specified.
gc_kind	Full GC| Mixed GC|Young GC |Young GC(initial-mark)|CM Remark|CM Cleanup	Indicates the type of GC or CM.
gc_info	<const1>K/<const2>K(<const3>K)-><const4>K/<const5>K(<const6>K)	Indicates memory information of the Java heap area. <const1>: The size of the used Java heap area before GC <const2>: The size of the used Java heap area before GC (region equivalent)[#1] <const3>: The size of the Java heap area before GC (region equivalent) <const4>: The size of the used Java heap area after GC <const5>: The size of the used Java heap area after GC (region equivalent) <const6>: The size of the Java heap area after GC (region equivalent) These are output in kilobytes.

Item to be Output	Content of the Output	Meaning
gc_time	*<time>*	Indicates the time the application was stopped by GC. This is output in seconds.
gc_status	-|to exhausted	Indicates the GC status. • If an overflow occurs in the To area, `to exhausted` is output. • For statuses other than above, - is output. The following shows the statuses output for each GC. If "gc_kind" is "Young GC", "Young GC (initial-mark)" or "Mixed GC": - or `to exhausted` is output. If "gc_kind" is other than above: - is output.
eden_info	<const1>K(<const2>K)-><const3>K(<const4>K)	Indicates memory information of the Eden area. <const1>: The size of the used Eden area before GC (region equivalent) <const2>: The possible maximum size of the Eden area before GC (region equivalent)#2 <const3>: The size of the used Eden area after GC (region equivalent) <const4>: The possible maximum size of the Eden area after GC (region equivalent)#2 These are output in kilobytes.
survivor_info	<const1>K-><const2>K	Indicates memory information of the Survivor area. <const1>: The size of the used Survivor area before GC (region equivalent) <const2>: The size of the used Survivor area after GC (region equivalent) These are output in kilobytes.
tenured_info	<const1>K-><const2>K	Indicates memory information of the Tenured area. <const1>: The size of the used Tenured area before GC (region equivalent) <const2>: The size of the used Tenured area after GC (region equivalent) These are output in kilobytes.
humongous_info	<const1>K-><const2>K	Indicates memory information of the Humongous area. <const1>: The size of the used Humongous area before GC (region equivalent) <const2>: The size of the used Humongous area after GC (region equivalent) These are output in kilobytes.
free_info	<const1>K-><const2>K	Indicates memory information of the Free area. <const1>: The size of the used Free area before GC (region equivalent) <const2>: The size of the used Free area after GC (region equivalent) These are output in kilobytes.
Metaspace_info	<const1>K(<const2>K, <const3>K)-><const4>K(<const5>K, <const6>K)]	Indicates memory information of the Metaspace area. • <const1>: The size of the used Metaspace area before GC • <const2>: The size of the capacity of the Metaspace area before GC • <const3>: The size of the committed Metaspace area before GC • <const4>: The size of the used Metaspace area after GC • <const5>: The size of the capacity of the Metaspace area after GC • <const6>: The size of the committed Metaspace area after GC These are output in kilobytes.
class_space_info	<const1>K(<const2>K, <const3>K)-><const4>K(<const5>K, <const6>K)]	Indicates memory information of the "class space" area. This item is not output if the compressed-object pointer functionality is disabled. • <const1>: The size of the used "class space" area before GC • <const2>: The size of the capacity of the "class space" area before GC • <const3>: The size of the committed "class space" area before GC • <const4>: The size of the used "class space" area after GC

Item to be Output	Content of the Output	Meaning
		• <const5>: The size of the capacity of the "class space" area after GC • <const6>: The size of the committed "class space" area after GC These are output in kilobytes.
cause_info	<letters>	Indicates the detailed cause of GC. This item is not output if `-XX:-HitachiVerboseGCPrintCause` is specified.
region_size	<const>	Indicates the size of one region. This is output in kilobytes.
target_time	<time>	Indicates the target time during which an application is stopped by GC. This is output in seconds.
predicted_time	<time>	Indicates the time during which an application is stopped by GC, as predicted by Java VM. This is output in seconds. If the GC type is `Full GC`, `CM Remark`, or `CM Cleanup`, the time is not predicted and 0 is output.
target_size	<const>	Indicates the size of the Tenured area targeted for GC by Mixed GC. This is output in kilobytes. If the GC type is other than `Mixed GC`, 0 is output.
reclaimable_info	<const1>K(<const2>.<digit><digit> %)	Indicates the predicted collection size information. <const1>: The predicted collection size This is output in kilobytes. <const2>.<digit><digit>: The predicted collection rate (to two decimal places) The predicted collection size information is output only for Young GC or Mixed GC immediately after the end of CM. In other cases, the size is not predicted and 0 is output.
user_cpu	<time>	Indicates the total CPU time spent in the user mode by all the GC threads. This is output in seconds. This item is not output if `-XX:-HitachiVerboseGCCpuTime` is specified. If collection of the CPU time fails, `unknown` is displayed as follows: [User: unknown]
system_cpu	<time>	Indicates the total CPU time spent in the kernel mode by the all GC thread. This is output in seconds. This item is not output if `-XX:-HitachiVerboseGCCpuTime` is specified. If collection of the CPU time fails, `unknown` is displayed as follows: [Sys: unknown]

#1: "Region equivalent" is a value indicated by multiples of the size of one region, calculated by rounding up the area size by the size of one region.

#2: The possible maximum size of the Eden area is: (size of New area) - (size of used Survivor area).

The VCM log

`[`*id*`]<`*date*`> [`*cm_event*`][User: `*user_cpu*` secs][Sys: `*sys_cpu*` secs]`

#No line break. #No space between items.

Items to be Output	Content of the Output	Meaning
id	VCM	Indicates the identifier of the Java VM log file. `[VCM]` is output in the CM log of G1GC.
date	<letters>	Indicates the date and time when CM was started. This item is not output if `-XX:-HitachiVerboseGCPrintDate` is specified.
cm_event	Concurrent Root Region Scan Start|Concurrent Root Region Scan End|Concurrent Mark Start|Concurrent Mark End|Concurrent Mark Stop|Concurrent Cleanup Start|Concurrent Cleanup End	Indicates the CM status. `Concurrent Root Region Scan Start`: Concurrent Root Region Scan started. `Concurrent Root Region Scan End`: Concurrent Root Region Scan ended. `Concurrent Mark Start`: Concurrent Mark started. `Concurrent Mark End`: Concurrent Mark ended. `Concurrent Mark Stop`: Concurrent Mark stopped. `Concurrent Cleanup Start`: Concurrent Cleanup started. `Concurrent Cleanup End`: Concurrent Cleanup ended.
usr_cpu	<time>	Indicates the total CPU time spent in the user mode by the all CM thread. This is output in seconds. This item is not output if `-XX:-HitachiVerboseGCCpuTime` is specified. If collection of the CPU time fails, `unknown` is displayed as follows: [User: unknown] If the CM status is "Start", 0 is output.
sys_cpu	<time>	Indicates the total CPU time spent in the kernel mode by the all CM thread. This is output in seconds. This item is output if `-XX:-HitachiVerboseGCCpuTime` is specified. If collection of the CPU time fails, `unknown` is displayed as follows: [Sys: unknown] If the CM status is "Start", 0 is output.

Input examples

1. Execution of the `javagc` command, with the `-i` option specified:
   ```
   javagc -i -v -p 8326
   ```
2. A process ID confirmation message is displayed:
   ```
   send SIGQUIT to 8326:?(y/n)
   ```
3. Enter `y` to send the `SIGQUIT` signal or `n` to not send the signal:
   ```
   send SIGQUIT to 8326:?(y/n)y
   ```

Output example

The VG1 log:

```
[VG1]<Thu Oct 02 10:38:56.193 2014>[Full GC 753K/2048K(8192K)->678K/
1024K(8192K), 0.0097901 secs][Status:-][G1GC::Eden: 1024K(2048K)->0K(2048K)]
[G1GC::Survivor: 0K->0K][G1GC::Tenured: 1024K->1024K][G1GC::Humongous: 0K->0K]
[G1GC::Free: 6144K->7168K] [Metaspace: 3634K(4492K, 4492K)->3634K(4492K,
4492K)][class space: 356K(388K, 388K)->356K(388K, 388K)] [cause:System.gc]
[RegionSize: 1024K][Target: 0.2000000 secs][Predicted: 0.0000000 secs]
[TargetTenured: 0K][Reclaimable: 0K(0.00%)][User: 0.0000000 secs][Sys:
```

```
0.0000000 secs][IM: 20459K, 21920K, 0K][TC: 35][DOE: 0K, 0][CCI: 1172K,
245760K, 2496K]
```

The VCM log:

```
[VCM]<Wed Jul 24 11:45:20 2013>[Concurrent Root Region Scan Start][User:
0.0000000 secs][Sys: 0.0000000 secs] [VCM]<Wed Jul 24 11:45:20
2013>[Concurrent Root Region Scan End][User: 0.0126134 secs][Sys: 0.0146961
secs] [VCM]<Wed Jul 24 11:45:20 2013>[Concurrent Mark Start][User: 0.0000000
secs][Sys: 0.0000000 secs] [VCM]<Wed Jul 24 11:45:34 2013>[Concurrent Mark
End][User: 0.0156250 secs][Sys: 0.2495800 secs]
```

Exit Status

Exit Status	Explanation
0	The command terminated normally.
1	The command terminated abnormally.
2	A response indicating that garbage collection was completed was not received within the specified amount of time.

Output messages

If any of the following error or warning messages is output, an extended thread dump with detailed information of the Explicit heap is not output:

No.	Error message	Explanation
1	`javagc [-f|-i][-v][-s] [-force] [-ehgc] -p process-id`	An argument specified in the `javagc` command is invalid.
2	`javagc: illegal option--`*`option`*	The indicated option (*`option`*) specified in the javagc command arguments is invalid.
3	*`process-id`*`: No such process`	The process indicated by *`process-id`* specified in the argument of the `javagc` command is not found or the process indicated by *`process-id`* specified in the javagc command is not the java process.
4	*`process-id`*`: Not owner`	The executing user is not the owner of the process indicated by *`process-id`* specified in the javagc command.
5	*`process-id`*`: Now processing previous request, this request canceled`	The process indicated by *`process-id`* specified in the argument of the `javagc` command is currently executing the garbage collection using the previous `javagc` command. The `javagc` command cancels the garbage collection execution request.
6	`javagc: can't create work file at /tmp, this request canceled`	The user does not have permission to reference or write data into `/tmp`, so a garbage collection request file cannot be created. This garbage collection execution request is cancelled.
7	`javagc: unexpected error occurred:`*`error-cause`*	An unexpected error occurred during `javagc` command execution. The following are examples of causes that may be displayed in *`error-cause`*: • When work memory could not be allocated: `malloc systemcall fail (errno=Y)` • When an object could not be closed: `close systemcall fail (errno=Y)`

No.	Error message	Explanation
8	*process-id*: Timeout occurred. Java process not responding.	The process indicated by *process-id* specified in the argument of the javagc command does not return a response indicating that the garbage collection is completed within the fixed time.
9	javagc : please delete *name-of-undeleted-file* in *full-path-of-undeleted-file*	When the javagc command terminated, it could not delete an internal file. Delete the indicated file on the indicated full path.
10	*process-id*: Failed to retry GC. Java process is GC locked.	The execution of GC by the process *process-ID* specified in an argument of the javagc command is suppressed, so GC could not be executed.

Notes

- The javagc command cannot be executed concurrently for the same java process. You execute the command when the previous javagc command terminates the garbage collection. After a previous garbage collection process is terminated, the JavaGC command is displayed as a garbage collection cause of the extended verbosegc functionality that is output to the Java VM log file.
- The javagc command sends the SIGQUIT signal to the specified process ID. If you specify a process ID of a non-java process by mistake, the other programs might stop.
- The javagc command uses the /tmp/hsperfdata_*user-name*/*process-ID* file at runtime. If the applicable file does not exist, the garbage collection cannot be requested using the javagc command. However, if the -force option is specified, a garbage collection occurs for the java process that is indicated by the process ID specified in the -p option, without confirming the existence of the /tmp/hsperfdata_*user-name*/*process-ID* file.

4.2.7 javatrace

trace information collection

Synopsis

```
javatrace core-file-name executable-file-name [output-file-name]
    [-l library-file-name ...]
```

Storage location

Application Server installation directory/jdk/jre/bin/

Function

This command collects stack trace information from a core file.

When a Java VM terminates abnormally and generates a core file, this command is executed to collect the information that is needed to identify the cause of the abnormal termination. You can use this command for conducting a detailed investigation of the cause of a process failure.

Arguments

core-file-name

Specifies the name of a core file.

executable-file-name

Specifies the name of the executable file that generated the core file.

output-file-name

Specifies a name for the output file.

If omitted, the trace information is output to `javatrace.log` under the current directory.

`-l` *library-file-name*

Specifies the library file to be used.

You do not need to specify this argument if you execute `javatrace` at the machine on which the core file was generated.

Output example

When a Java VM terminates abnormally and generates a core file, the message shown below is issued. Execute the `javatrace` command character string within this message.

Note that this message is also output to the error report file (`hs_err_pid`*process-id*`.log`), which is generated during abnormal termination.

```
        :
# You can get further information from javatrace.log file generated
# by using javatrace command.
# usage: javatrace core-file-name loadmodule-name [out-file-name]
# [-l(library-name)...]
# Please use javatrace command as follows and submit a bug report
# to Hitachi with javatrace.log file:
# [Application Server installation directory/jdk/jre/bin/javatrace core
# Application Server installation directory/jdk/jre/bin/java]
#
```

When the name of the core file is `core`:

```
Application Server installation directory/jdk/jre/bin/javatrace core
Application Server installation directory/jdk/jre/bin/java
```

Depending on the OS, the name of the actual core file that is output may be `core.`*process-id*. If this is the case, specify the name of the actual core file that is output in the `javatrace` command argument.

When the name of the core file is `core.`*process-id*:

```
Application Server installation directory/jdk/jre/bin/javatracecore.8326
Application Server installation directory/jdk/jre/bin/java
```

Exit Status

Exit Status	Explanation
0	The command terminated normally.
1	The command terminated abnormally.

Output messages

If any of the following error or warning messages is output, stack information is not output.

No.	Error message	Explanation
1	`usage : javatrace core-filename loadmodule-name [out-filename] [-llibrary-name...]`	An argument specified in the `javatrace` command is invalid.
2	`javatrace: Cannot open` *`file-name`* `:No such file or directory`	The file (*`file-name`*), other than the one specified in *`output-file-name`*, could not be found.
3	`javatrace: Cannot open` *`file-name`* `: Permission denied`	Loading of a file (*`file-name`*) is not permitted.
4	`javatrace : Cannot create` *`file-name`* `: Already exist`	The file (*`file-name`*) specified in *`output-file-name`* already exists.
5	`* unknown core type(`*`XXXXXXX`*`) ignored.`	The file specified in *`core-file-name`* is not a core file.
6	`* ERROR :` *`file-name`*`, unknown magic(0x`*`XXXX`*`)`	The file specified in *`executable-file-name`* is not an executable file.
7	`WARNING : core file may not match loadmodule (core file from '`*`loadmodule-name`*`')`	The file specified in *`executable-file-name`* does not match the executable file (*`loadmodule-name`*) extracted from the core file specified in *`core-file-name`*.
8	`javatrace : Bad argument :` *`argument-name`* `is directory`	A specified file (*`argument-name`*) is actually a directory.
9	`* ERROR :` *`library-name`* `open : No such file or directory`	A shared library (*`library-name`*) included in the executable file could not be found. If the shared library was included using a relative path, you must use the `-l` option to explicitly specify it.
10	`javatrace : illegal option --` *`x`*	An invalid option character (*x*) is specified.
11	`javatrace : Cannot create` *`file-name`* `: Permission denied`	Writing of data into the file (*`file-name`*) specified in output-file-name is not permitted.

Notes

- You execute the `javatrace` command on a machine on which the core file is generated.
- In the message output to the standard output and the error report file, when JavaVM terminates abnormally, the second argument of the `javatrace` command is not always the load module name of the absolute path, as given below:

```
        :
# You can get further information from javatrace.log file generated
# by using javatrace command.
# usage: javatrace core-file-name loadmodule-name [out-file-name]
# [-l(library-name)...]
# Please use javatrace command as follows and submit a bug report
# to Hitachi with javatrace.log file:
# [Application Server installation directory/jdk/jre/bin/javatrace core
# Application Server installation directory/jdk/jre/bin/java]
#
```

  In such cases, first convert the load module name specified in the second argument of the `javatrace` command to the absolute path, and then execute the command.

4.2.8 jheapprof

Output of extended thread dump containing class-wise statistical information

Synopsis

```
jheapprof [-i|-f] [-class class-name] [-staticroot|-nostaticroot]
    [-explicit|-noexplicit] [-fullgc|-copygc|-nogc]
    [-garbage|-nogarbage] [-rootobjectinfo|-norootobjectinfo]
    [-rootobjectinfost value] [-force] -p process-ID
```

Storage location

Application Server installation directory/jdk/jre/bin/

Function

This command outputs an extended thread dump, including statistics by class, about the java process with the process ID specified in the argument.

You can output the size of all instances under the members of each class instance, into an extended thread dump as statistics by class. By outputting statistics multiple times, you can check the change in a Java object size or other information. You can use this information to detect memory leakage.

Execution permission

Execute the command as the same user as the user of the java process specified by the process ID in the arguments.

Arguments

`-i`

Displays the message confirming the execution of the output processing for an extended thread dump containing the class-wise statistical information. You input either `y` or `n`. If you input `y`, the extended thread dump containing the class-wise statistical information is output. If you input `n`, no information is output and the processing terminates. Even if you omit this option, the option is valid as long as the `-f` option is not specified.

`-f`

Disables the `-i` option. If you omit this option, the `-i` option becomes valid.

`-class` *class-name*

Creates a list of the classes that include the specified class as a member, and outputs the list within the thread dump.

`-staticroot`

Enables the reference related information output functionality that is based on the static field and outputs the reference related information based on the static field. If you omit this option, the `-nostaticroot` option is enabled.

The specification of the `-class` option is a prerequisite for this option. When you do not specify the `-class` option, this option is disabled.

If this option is concurrently specified with the `-nostaticroot` option, the option specified at the end is enabled.

`-nostaticroot`

Disables the reference related information output functionality based on the static field.

If this option is concurrently specified with the `-staticroot` option, the option specified at the end is enabled.

-explicit

The Explicit heap is included in the statistics of the instance statistics function. If omitted, this option is enabled unless the -noexplicit option is specified.

Note that if this option and the -noexplicit option are specified concurrently, the option specified last is enabled.

-noexplicit

The Explicit heap is not included in the statistics of the instance statistics function. If it is omitted , this option is enabled as long as -copygc and -nogc option are not specified. If omitted, the -explicit option is enabled.

Note that if this option and the -explicit option are specified concurrently, the option specified last is enabled.

-fullgc

Specifies a full garbage collection for the garbage collection that is to be executed before the statistics is collected. If omitted, this option is enabled unless the -copygc option and -nogc option is specified.

Note that if you concurrently specify this option with the -copygc option or the -nogc option, the option that is specified last is applied.

-copygc

Specifies a copy garbage collection for the garbage collection that is to be executed before the statistics is collected. If this option and the -nogc option are omitted, the -fullgc option is enabled.

Note that if you concurrently specify this option with the -fullgc option or the -nogc option, the option that is specified last is applied.

-nogc

The garbage collection will not be executed before the statistics is collected. If this option and the -copygc option are omitted, the-fullgc option is enabled.

Note that if you concurrently specify this option with the -fullgc option or the -copygc option, the option that is specified last is applied.

-garbage

Enables the unused object statistic function in the Tenured area and outputs the class-wise statistical information for the statistic information of the unused objects to the Tenured area. The instance statistic function and the STATIC member statistic function are disabled. If omitted, the -nogarbage option is enabled.

For the Select pre-statistic garbage collection function, the -fullgc option and the -copygc option are disabled, and the -nogc option is enabled. Consequently, the garbage collection is not executed before the statistics process.

Note that if you specify this option concurrently with the-nogarbage option, the option that is specified last will be applied.

-nogarbage

The unused object statistic function in the Tenured area is disabled. Therefore, the class-wise statistical information is not output for the statistic information of the unused objects to the Tenured area. If omitted, this option is enabled unless you specify the -garbage option.

Note that if you specify this option concurrently with the -garbage option, the option that is specified last will be applied.

-rootobjectinfo

The output functionality of the base object list for increasing the Tenured area is enabled and outputs the base object for increasing the Tenured area.

For this option, you must enable the -garbage option. Also, when you enable the -nogarbage option, this option is disabled. If omitted, this option is enabled unless you specify the -norootobjectinfo option.

Note that if you specify this option concurrently with the -norootobjectinfo option, the option specified last will be applied.

`-norootobjectinfo`

The base object list output functionality for increasing the Tenured area is disabled. Therefore, the base object list for increasing the Tenured area is not output. If you omit the output of the base object list, the `-rootobjectinfo` option is enabled.

Note that if you specify this option concurrently with the `-rootobjectinfo` option, the option specified last will be applied.

`-rootobjectinfost` *value*

Controls the volume of the information of the basic object list for increasing the Tenured area. The class information, with the total instance size greater than the specified *value*, is output to the basic object list for increasing the tenured area.

For this option, you must enable the `-rootobjectinfo` option. Therefore, when you enable the `-norootobjectinfo` option, this option is disabled. You can specify a whole number in *value*. When you specify any number other than a whole number or a character string, the process ends with an error message indicating that there is an error in specifying the argument.

Default value: `0`

`-force`

Requests the output of the extended thread dump for the java process indicated by the process ID specified in the `-p` option, without confirming the existence of the `/tmp/hsperfdata_`*user-name*`/`*process-ID* file created in the java process.

`-p` *process-id*

Specifies the process ID of the Java program for which the class-wise statistical information is to be output.

Type: Integer

The following values can be specified:

- `0` to `4294967295`

Output format

Use the `jheapprof` commands to output the class-wise statistical information to the extended thread dump. Specify the Java process, where you want to output the class-wise statistical information, and the class, where you want to output the reference-related information, and then execute the `jheapprof` commands.

You can specify the following when executing the `jheapprof` command:

- Specify whether to output the information of an Explicit heap as class-wise statistical information.
- Specify whether to execute the garbage collection before acquiring the class-wise statistical information.
- The following shows an example of executing the `jheapprof` command:

  Here, the class-wise statistical information of Java process with process ID `2463` is output.

  1. In the `-p` option, specify the process ID of the Java process where you want to output the class-wise statistical information, and then execute the `jheapprof` command.

     ```
     % jheapprof -p 2463
     ```

     When the `-f` option is being omitted in the `jheapprof` command, the following confirmation message is displayed:

     The confirmation message of process ID is displayed in the following format:

     ```
     send SIGQUIT to 2463: ? (y/n)
     ```

  2. Enter `y`.

An extended thread dump with class-wise statistics is output. The following message is output in the running java program:

```
Writing Java core to javacore2463.030806215140.txt... OK
```

The running java program creates an extended thread dump with class-wise statistics (`javacore.`*`process ID`*`.`*`date-time`*`.txt`) in the current directory and continues the program.

- When the information of an Explicit heap is output to the class-wise statistical information

  If the following conditions are satisfied, you can output the information of an Explicit heap to the class-wise statistical information:

  - `-XX:+HitachiUseExplicitMemory` is specified in the JavaVM start option.
  - Explicit heap is used for implementing the application, or setting the execution environment (J2EE server).

  Specify the `-explicit` option in the `jheapprof` command, and then execute the command to output the information of an Explicit heap to the class-wise statistical information.

- When specifying whether to execute the garbage collection

  You can select whether to execute the garbage collection before the class-wise statistical information is output. This functionality is called as pre-statistical garbage collection selection functionality. Specify any of the following options in the `jheapprof` command, if you want to execute the garbage collection before the class-wise statistical information is output:

  - `-fullgc`

    Executes the full garbage collection, and then outputs the class-wise statistical information.
  - `-copygc`

    Executes the copy garbage collection, and then outputs the class-wise statistical information.
  - `-nogc`

    Outputs the class-wise statistical information without executing the garbage collection.

Examples

1. Execution of the `jheapprof` command, with the `-f` option omitted:
   ```
   % jheapprof -p 2463
   ```
2. The process ID confirmation message is displayed:
   ```
   send SIGQUIT to 2463: ? (y/n)
   ```
3. Enter `y` to send the `SIGQUIT` signal or enter `n` to not send the signal:
   ```
   send SIGQUIT to 2463: ? (y/n)y
   ```
4. When the extended thread dump containing the class-wise statistical information is output, the running java program issues the following message:
   ```
   Writing Java core to javacore2463.030806215140.txt... OK
   ```
5. The running java program creates an extended thread dump containing the class-wise statistical information (`javacore`*`process-id`*`.`*`date`*`.txt`) in the current directory and continues.

Exit Status

Exit Status	Explanation
0	The command terminated normally.
1	The command terminated abnormally.

Exit Status	Explanation
2	There is no response indicating that the output processing has terminated within the fixed time for the class-wise statistical information.

Output messages

If any of the following error or warning messages is output, the extended thread dump with statistics by class is not output.

No.	Error message	Explanation
1	`usage: jheapprof [-f|-i] [-class classname] [-staticroot|-nostaticroot] [-explicit|-noexplicit] [-fullgc|-copygc|-nogc] [-garbage|-nogarbage] [-rootobjectinfo|-norootobjectinfo] [-rootobjectinfost size] [-force] -p process-id`	An argument specified for the `jheapprof` command is invalid.
2	`jheapprof: illegal option -- option`	The indicated option (*option*) specified in the `jheapprof` command is invalid.
3	*process-id*`: Now processing previous request, this request canceled`	The process indicated by *process-id* specified in the argument of the `jheapprof` command is currently engaged in the output of the class-wise statistical information.
4	*process-id*`: No such process`	The process indicated by *process-id* specified in the argument of the `jheapprof` command is not found or the process indicated by *process-id* specified in the `jheapprof` command is not the java process.
5	*process-id*`: Not owner`	The user is not the owner of the process indicated by *process-id* specified in the argument of the `jheapprof` command.
6	`jheapprof: can't create work file at /tmp , this request canceled`	If the user does not have permission to reference or write data into the directory for temporary files, the extended thread dump containing the class-wise statistical information cannot be output. The output request for the extended thread dump containing the class-wise statistical information is cancelled.
7	`jheapprof: please delete` *name-of-undeleted-file* `in` *full-path-of-undeleted-file*	When the `jheapprof` command terminated, it could not delete an internal file. Delete the indicated file on the indicated full path.

8	`jheapprof: unexpected error occurred:` *error-cause*	An unexpected error occurred during `jheapprof` command execution.
		The following are examples of causes that may be displayed in *error-cause*:
		• When work memory could not be allocated:
		`malloc systemcall fail (errno=Y)`
		• When an object could not be closed:
		`close systemcall fail (errno=Y)`

9	*process-id*`: Timeout occurred. Java process not responding`	The process indicated by *process-id* specified in the argument of the `jheapprof` command did not return a response indicating that the output processing terminated within the fixed time for the class-wise statistical information.
10	*process-id*`: Failed to retry GC. Java process is GC locked.`	The execution of GC by the process *process-ID* specified in the argument of the `jheapprof` command is suppressed, so GC could not be executed.

Notes

- The jheapprof command is provided as a utility for program development. You do not use this command in system operations.
- The jheapprof command cannot be executed concurrently for the same java process. Execute the command after the class-wise statistical information is output to the extended thread dump by a previous jheapprof command.
- The jheapprof command sends the SIGQUIT signal to the specified process. If you specify a program other than the java program by mistake, the other program might stop.
- The jheapprof command uses the /tmp/hsperfdata_*user-name*/*process-ID* file at runtime. If the applicable file does not exist, the statistical information by the class is not output from the jheapprof command.
- To include the Explicit heap in the statistics of the instance statistical information, you specify -XX:+HitachiUseExplicitMemory, and then specify the -explicit option using the jheapprof command for the java process in which the explicitly managed heap functionality is enabled.

  Note that if -XX:-HitachiUseExplicitMemory is specified and the explicitly managed heap functionality is disabled, even if you specify -explicit, the instance in the Explicit heap will not be included in the statistics of the instance statistical information.

4.2.9 jheapprofanalyzer

CSV output of class-wise statistical information analysis file

Synopsis

```
jheapprofanalyzer [-J option-name] [file-name]
```

Storage location

Application Server installation directory/jdk/jre/bin/

Function

This command uses multiple extended thread dump files (with statistics by class) as input, and outputs the total size of instances by class and number of instances by class in chronological order. The output file is a CSV file.

When you use the statistics-by-class analysis function, you can output only statistics of instances whose total sizes are large, and check the memory usage only for these instances. To output only instances whose total sizes are large, execute the jheapprofanalyzer command and specify a threshold for -DJP.co.Hitachi.soft.jvm.tools.jheapprofanalyzer.threshold.

Files

Extended thread dump file with statistics by class (javacore*process-ID*.*date-and-time*.txt)

Arguments

-J *option-name*

You can specify the following options in *option-name*. Also, when you specify options excluding the following options, the operation is not guaranteed:

- `-Xms`

  Specify the initial size of the memory allocation pool in number of bytes.

- `-Xmx`

  Specify the maximum size of memory allocation pool in number of bytes.

- `-DJP.co.Hitachi.soft.jvm.tools.jheapprofanalyzer.threshold=`*num*

  num: Specifies the threshold value of the instance total size. The range is from `0` to `263-1` (`Long.MAX_VALUE`). Only the classes where the instance total size is more than *num* are output.

  Default value: `1024`

File name

You can specify the extended thread dump file containing the class-wise statistical information. There are no special rules for specifying file names. Moreover, you can specify the files in any order without any limitations on the file count.

Output format

The following explains the input file, output file, and output format of the statistics-by-class analysis function.

Input files

The extended thread dump files in which the class-wise statistical information is output are used as the input files in the class-wise statistical information analysis functionality.

Output files

The files output in the class-wise statistical information analysis functionality include two types of files, such as files that output the total instance size of each class and files that output the number of instances of each class. The output files are created in the current directory with the following names.

Type of output file	Example of output file name
Instance total size file	`JheapprofAnalyzer_size_`*nnn*`.csv`
Instance count file	`JheapprofAnalyzer_num_`*nnn*`.csv`

Legend:

nnn: The file segmentation number is output. The segmentation number is in the range of `001` to `999`.

The output file is segmented when the number of columns exceeds 201. When the number of files exceeds 999, the count returns to `001` and files are re-written.

When the number of columns at which segmentation is performed exceeds 201 (1 column for class name + 200 columns for value), the output format is the same for the segmented file as well.

Output format

The following table shows the output format of the files output in the class-wise statistical information analysis file.

Note that the output format of the CSV file in which the total instance size and number of instances are output is also the same.

The first column is the class name. The maximum number of input file names is 200 (columns).

class name,	input file name,	input file name,	...	input file name,
class name,	*value*-1-1,	*value*-1-2,	...	*value*-1-xxx
:	:	:	...	:
class name,	*value*-y-1,	*value*-y-2,	...	*value*-y-xxx

Legend:

input file name: Indicates the statistics by class specified as the processing target.

class name: Indicates the class name output in the input file.

value: Indicates the total instance size or number of instances.

Demarcate a class name and value, and a value from another value with a comma. End a line with a value (including a blank).

Class names are output in a random order. Based on the date displayed as the value in the first row of an input file, the input files are arranged side by side starting from the file with the oldest date. If input files with the same date exist, they are connected randomly and arranged side by side.

Reference note

If you execute the class-wise statistical information analysis functionality more than once, classes might be removed or added during processing. Also, `0` will be output as the value when the corresponding class does not exist. The following figure shows the class information.

First class-wise statistical information (A.txt)	Second class-wise statistical information (B.txt)	Third class-wise statistical information (C.txt)

`ClassA 100`
`ClassB 100`
	`ClassA 100`
	`ClassB 30`
	`ClassC 50`
	`ClassD 0`
		`ClassA 100`
		
		`ClassC 50`

For the class information above, if `0` is specified for the threshold in `-DJP.co.Hitachi.soft.jvm.tools.jheapprofanalyzer.threshold`, the following result is output:

```
class name,A.txt,B.txt,C.txt
ClassA, 100,100, 100
ClassB, 100, 30, 0
ClassC, 0, 50, 50
ClassD, 0, 0, 0
```

The maximum value of total instance size is `0` to $2^{63}-1$, and the maximum value of the number of instances is `0` to $2^{31}-1$. If the same class name exists in an input file, the total instance size is added. The number of instances is also added. If the respective maximum values are exceeded due to adding up, the specified maximum value is output. Note that if the corresponding class information does not exist in all input files of a class or if the threshold value is not reached, the information of that class is not output.

Examples

```
jheapprofanalyzer -J-Xms1024m -J-Xmx1024m -J-
DJP.co.Hitachi.soft.jvm.tools.jheapprofanayzer.threshold=5000
javacore22356.080523161703.txt javacore22356.080523161711.txt
```

Exit Status

Exit Status	Explanation
`0`	The command is terminated normally.
`1`	The command is terminated abnormally.

Output message

If any of the following error or warning message is output, the statistics-by-class analysis file is not output. If other error messages are output, the default exception processing is performed.

No.	Error message	Explanation	Operation after output
1	`usage: jheapprofanalyzer [options] file...where options include:-J-Xms<size> set initial Java heap size-J-Xmx<size> set maximum Java heap size-J-DJP.co.Hitachi.soft.jvm.tools.jheapprofanalyzer.threshold=<num> set instance total size threshold`	An argument specified for the `JheapprofAnalyzer` class is invalid.	(a)
2	`JheapprofAnalyzer: Illegal property value`*num*`. Default is assumed.`	A value other than an integer is specified in *num* of `JP.co.Hitachi.soft.jvm.tools.jheapprofanalyzer.threshold` or the *num* is out of range.	(b)
3	`JheapprofAnalyzer: can't open input file`*file-name*	*file-name* does not exist in the directory or the file does not open due to some other factors.	(c)
4	`JheapprofAnalyzer: can't read input file`*file-name*	Failed in loading *file-name*	(c)
5	`JheapprofAnalyzer: Illegal input file format`*file-name*	*file-name* is not an extended thread dump file containing the class-wise statistical information.	(c)
6	`JheapprofAnalyzer: can't open output file`*file-name*	Unable to open the output file. The following status can be considered as the reasons for error: • The output file is changed to a directory. • There is no output file. • Unable to open the output file due to some other reasons.	(a)
7	`JheapprofAnalyzer: can't write output file`*file-name*	Failed in writing *file-name*.	(a)

Legend:

(a): The process ends abnormally.

(b): Continues executing the process assuming the default value.

(c): Continues the process and checks the error of all the specified input files.

Notes

In the class-wise statistical information analysis functionality, the file is opened only when the date is acquired and when data is loaded. Therefore, the result is not guaranteed for updating and deleting the input data, when the command is being executed.

5 Commands used in the PRF

This chapter describes the syntax and functionality of the commands used in the PRF.

5.1 List of commands to be used in the PRF

The following table shows a list of commands to be used in the PRF.

Commands used for start and stop PRF daemon

Command Name	Classification	Summary
cprfgetpid	Acquire a PRF daemon process ID	This command acquires the process ID of the specified PRF daemon.
cprfstart	Start PRF daemon	This command starts the PRF daemon (`cprfd`).
cprfstop	Stop PRF daemon	This command stops the PRF daemon (`cprfd`).

Commands used for output the trace information

Command Name	Classification	Summary
cprfed	Edit and output performance analysis trace information	This command inputs PRF trace information and edits and outputs performance analysis trace information.
cprfflush	Forcibly output buffer contents to a file	This command requests the PRF daemon to output the trace information stored in the buffer.
cprflevel	Display or change the PRF trace collection level	This command displays or changes the PRF trace collection level.

Important note

- Specify the value which can be specified by each command as the argument of a command. Operation is not guaranteed when other values are specified.
- Even if a command is executed normally, errors may have occurred. Check the message log and stack trace log.

5.2 Commands used for start and stop PRF daemon

This section describes the syntax and functionality of the commands used for start and stop PRF daemon.

5.2.1 cprfgetpid

acquire a PRF daemon process ID

Synopsis

```
cprfgetpid [-PRFID prf-identifier] [-h]
```

Storage location

Application Server installation directory/common/PRF/bin/

Function

This command acquires the process ID of the specified PRF daemon. It also displays whether or not the PRF daemon has started.

Execution permission

A user who has the following permission:

- Either the same user for the processes that obtain PRF or the superuser
- In an environment that uses the same PRFSPOOL environment variable, the superuser and other users coexist and the PRF commands cannot be executed. The PRF commands must always be executed by the same user.

PreCondition

- The same PRFSPOOL environment variable as the one assumed by the request destination PRF daemon must be set.
- If the PRF daemon has not been started at all or if this command is executed after the PRF daemon terminated normally, an error results.

Environment variable

- PATH

  Add *Application Server installation directory*/common/PRF/bin and /bin.
- LD_LIBRARY_PATH(Linux)

  Specify *Application Server installation directory*/common/PRF/lib.
- PRFSPOOL

  Specifies the execution environment directory of the performance tracer (PRF daemon).

  The PRFSPOOL environment variable is set by the domain administration server when the domain administration server starts a JAVA EE server or a cluster. If a PRF association whose association source is the Java EE server exists, the PRFSPOOL environment variable is set in the following format based on the PRF name of the association destination:

"*Java-EE-Server-log-output-destination-directory-for-the-node-in-which-the-JAVA-EE-server-is-built*/nodes/*node-name*/*PRF-name*"

- TZ(UNIX)

  Specifies the time zone.

Arguments

-PRFID *prf-identifier*

For the PRF identifier, specify the PRF server name that is given when the PRF is created by using the create-prf subcommand of asadmin.

Specifies a PRF daemon identifier, as 1 to 31 alphanumeric characters and the underscore (_). Do not specify a character string that begins with TSC or tsc or with CTM or ctm.

Default value: PRF_ID

-h

Displays the command's usage.

Input/output examples

- Input example

  cprfgetpid

- Output example

  The result is output to the standard output.

  1700

Exit Status

Exit Status	Explanation
0	The command terminated normally. Processing might not have started.
1	Processing has not started.
2	The PRFSPOOL environment variable has not been set up.
Other than the above	An error occurred during command processing. Take the corrective action indicated in the output message, and then re-execute the command. The error message is output to the standard error output and a log file.

5.2.2 cprfstart

The cprfstart command starts the PRF daemon.

Synopsis

```
cprfstart [-h] [-PRFID prf-identifier] [-PrfTraceLevel
    prf-trace-collection-level[,[prf-trace-collection-level]...]]
    [-PrfTraceCount prf-trace-file-count]
    [-PrfTraceFileSize prf-trace-file-size]
    [-PrfTraceBufferSize prf-trace-file-size]
    [-PrfRemakeBuffer] [-PrfNoBackUp [0|1]] [-PrfConsole {0|1}]
    [-PrfLogShiftTime PRF-log-shift-time]
```

```
    [-PrfLogFileSize PRF-log-file-size] [-PrfLogFileCount Maximum-number-of-PRF-log-
files]
```

Storage location

Application Server installation directory/common/PRF/bin/

Function

This command starts the PRF daemon (cprfd). You must start the PRF daemon before starting other processes.

Once the PRF daemon starts, you must change the current directory to the following:

prf-trace-output-directory$PRFSPOOL/utt/prf/*prf-identifier*/

Execution permission

A user who has the following permission:

- Either the same user for the processes that obtain PRF or the superuser
- In an environment that uses the same PRFSPOOL environment variable, the superuser and other users coexist and the PRF commands cannot be executed. The PRF commands must always be executed by the same user.

Precondition

- You must set the PRFSPOOL environment variable. If it is not set, the command returns an error.
- In the default settings after the OS is installed, the local host name is set for the IP address 127.0.0.1 in the /etc/hosts file. When a PRF is used, the local host name must be set for the IP address assigned to the network interface. For this reason, in the /etc/hosts file, set the correct IP address that is assigned to the network interface for the local host.

Environment variable

- PATH

  Add *Application Server installation directory*/common/PRF/bin and /bin.
- LD_LIBRARY_PATH(Linux)

  Specify *Application Server installation directory*/common/PRF/lib.
- PRFSPOOL

  Specifies the execution environment directory of the performance tracer (PRF daemon).

  The PRFSPOOL environment variable is set by the domain administration server when the domain administration server starts a JAVA EE server or a cluster. If a PRF association whose association source is the Java EE server exists, the PRFSPOOL environment variable is set in the following format based on the PRF name of the association destination:

  "*Java-EE-Server-log-output-destination-directory-for-the-node-in-which-the-JAVA-EE-server-is-built*/nodes/*node-name*/*PRF-name*"
- TZ(UNIX)

  Specifies the time zone.

Arguments

`-h`

Displays the command's usage.

`-PRFID` *`prf-identifier`*

For the PRF identifier, specify the PRF server name that is given when the PRF is created by using the `create-prf` subcommand of `asadmin`.

Specifies a PRF daemon identifier, as 1 to 31 alphanumeric characters and the underscore (_). Do not specify a character string that begins with `TSC` or `tsc` or with `CTM` or `ctm`.

Default value: `PRF_ID`

`-PrfTraceLevel` *`prf-trace-collection-level`* `[,[`*`prf-trace-collection-level`*`]...]`

When `-PrfTraceLevel` is omitted, the information is output in Standard-level.

Specifies PRF trace collection levels, each as a 4-byte hexadecimal number (8-digit value). You can add `0x` at the beginning of the 4-byte hexadecimal number, however, `0x` will be ignored.

You can specify multiple trace collection level values, corresponding to the index numbers (left to right), delimited by a comma (`,`). If a level value specification is to be omitted, do not specify any value. To omit specification of level values for all index numbers beginning after a specific index number, omit all the subsequent comma delimiters. To set the PRF trace collection level for index No. 1 to (1) and the PRF trace collection level for index No. 2 to (2), specify the arguments as (1),(2),....

The table below shows examples of PRF trace collection level specifications.

PRF trace collection level specification example	Option specification
Specify a PRF trace collection level for index No. 1	`-PrfTraceLevel 0x44445555`
Specify PRF trace collection levels for index Nos. 1 and 2	`-PrfTraceLevel 0x44445555,0x55554444`
Specify a PRF trace collection level for index No. 2	`-PrfTraceLevel ,0x55554444`

Assignment of PRF trace collection levels at each functionality layer is explained here.

A PRF trace collection level is specified as an eight-digit hexadecimal number for each index. Each digit is allocated two functionality layers. When a hexadecimal number is expressed as a decimal number, the functionality layer allocated to the upper 2 bits is called the upper layer, and the functionality layer allocated to the lower 2 bits is called the lower layer.

The table below shows the functionality layers that are allocated to the upper and lower layers for each digit.

Index	Digits	Upper layer	Lower layer
Index 1	Digit 1	(None)	Java EE Server
	Digit 2	(None)	Java VM
	Digit 5	(None)	uCosminexus TP1 Connector, TP1/Client/J

To specify a PRF trace collection level, decide whether to specify Standard, Detail, or Maintenance for the PRF trace collection level of the upper and lower layers for each digit, and specify it as a hexadecimal number. The table below shows the correspondence between combinations of PRF trace collection levels for the upper and lower layers with specification values.

Note that because the Maintenance level is used for collecting maintenance information when an error occurs, it should not be specified during normal operation.

Upper layer	Lower layer	Specification value
Standard	Standard	0

Upper layer	Lower layer	Specification value
Standard	Detail	1
Standard	Maintenance	2
Detail	Standard	4
Detail	Detail	5
Detail	Maintenance	6
Maintenance	Standard	8
Maintenance	Detail	9
Maintenance	Maintenance	a

Specification examples are shown below.

Index	Specification example	Explanation
Index 1	00000000	Acquires Standard-level trace information from each functionality layer of index 1.
	10000000	Acquires Detail-level trace information from Java EE Server layer only, and acquires Standard-level trace information from other functionality layers.
	11000000	Acquires Detail-level trace information from both Java EE Server and the Java VM layers.

`-PrfTraceCount` *`prf-trace-file-count`*

Specifies the number of trace files, as a value in the range `3` to `256`.

The PRF trace files are backed up during a normal stop and during a restart following a forced stop. If the total PRF trace file size is large, it might take a considerable amount of time to start and stop the PRF daemon because of the backup processing. Consequently, when Management Server is used to monitor processes, a timeout might occur if the monitoring time is set to the default value. If a timeout occurs, increase the monitoring time.

Default value: `4`

`-PrfTraceFileSize` *`prf-trace-file-size`*

Specifies the maximum size (in kilobytes) of each PRF trace file, as a value in the range `1024` to `1048576`. To set 1 megabyte as the file size, specify `1024`. The actual file size may be slightly more or less than this value. The maximum difference will be 32 KB less than the value specified in `-PrfTraceBufferSize`.

If you specify `0` for the `-PrfNoBackUp` option, the PRF trace files are backed up during a normal stop, and during a restart following a forced stop.

If the total PRF trace file size is large, it might take a considerable amount of time to start and stop the PRF daemon because of the backup processing.

Therefore, if you specify `0` for the `-PrfNoBackUp` option to monitor processes by the domain administration server, a timeout might occur if the monitoring time is set to the default value.

If a timeout occurs, increase the monitoring time.

Default value: `8192`

If you specify a value that is smaller than the default value of `8192`, you must also specify a value smaller than the default value in `-PrfTraceBufferSize`.

`-PrfTraceBufferSize` *`prf-trace-buffer-size`*

Specifies the size (in kilobyte) of the buffer to be allocated in the shared memory, as a value in the range `512` to `102400`. In this you cannot specify a value greater than the value specified in `-PrfTraceFileSize` with this option.

If the buffer area is insufficient, the KFCT26999-W message is displayed and the PRF trace might be omitted. Therefore, extend the buffer area until the message is no longer displayed.

Default value: 8192

-PrfRemakeBuffer

Re-creates the shared memory before restarting the PRF daemon. If the shared memory cannot be re-created, an error occurs.

-PrfNoBackUp [0|1]

Specifies whether to take a back up of the PRF trace file when the PRF daemon starts and stops.

If 0 is specified in the value, take the back up of the PRF trace file. Because the PRF trace file is copied, it might take time to start and stop the PRF daemon. After failure occurs, if the PRF trace file cannot be extracted until the PRF trace is wrapped, specify 0 in the value.

Note that when Management Server is used, the snapshot log containing PRF trace files is collected automatically after failure.

If 1 is specified, do not take a back up of the PRF trace file. If a value is not specified, the operation is same as when 1 is specified.

The following is the relation between the timings for taking backup of the PRF trace file, availability of backup, and the -PRFNoBackUp option:

Execution Command	End condition of PRF daemon which has started previously	Backup of PRF trace file	
		-PrfNoBackUp 0	**-PrfNoBackUp 1**
cprfstart	Normal end	No	No
	Forceful end or abnormal end	Yes	No
cprfstop	Not dependent on end condition	Yes	No
cprfstop -Force	Not dependent on end condition	No	No

Default value: 1

-PrfConsole {0|1}

Specify whether to display the DOS prompt of the PRF daemon when the cprfstart command is executed from a Windows desktop. This option is available for the Windows OS only.

The following values can be specified:

- 1

  Displays the DOS prompt.
- 0

  Does not display the DOS prompt.

Default value: 0

-PrfLogShiftTime *PRF-log-shift-time*

If you want to shift the log file to be output by the PRF according to time, specify the shift time. Logs are shifted when logs are output to the log file. Therefore, logs are shifted only if logs are output.

The following values can be specified:

- 000000 to 235959

Default value: 000000

-PrfLogFileSize *PRF-log-file-size*

Specify, in megabytes, the maximum size of the log file to be output by the PRF.

The following values can be specified:

- 1 to 100

Default value: 10

-PrfLogFileCount *Maximum-number-of-PRF-log-files*

Specify the maximum number of log files that the PRF can output.

The following values can be specified:

- 1 to 32

Default value: 8

Exit Status

Exit Status	Explanation
0	The PRF daemon started normally.
1	Startup of the PRF daemon failed. The PRFSPOOL environment variable has not been set up.
Other than the above	An error occurred during command processing. Take the corrective action indicated in the output message, and then re-execute the command. The error message is output to the standard error output and a log file.

Notes

- If you have to restart the PRF daemon, specify the same values in -PrfTraceCount and -PrfTraceFileSize as were set previously. Otherwise, when the PRF daemon is restarted without these options specified, the values that were being used during the previous session will not be inherited for the PRF trace file count and the PRF trace file size, so the default values will be used.
- If the PRF daemon terminated abnormally and is then restarted, the trace collection level that was in effect when the PRF daemon terminated is inherited. Consequently, if the abnormal termination occurred after the trace collection level had been changed, a value that is different (changed value) from the trace collection level specified during the restart (the same value as was used during the previous session) is set. An example follows:

  1. PRF daemon startup: cprfstart -PrfTraceLevel 0x00000001
  2. Trace collection level change (0x00000001 to 0x0000000f)
  3. PRF daemon abnormal termination
  4. PRF daemon restart: cprfstart -PrfTraceLevel 0x00000001

  In this case, the trace collection level following the restart is 0x000000f.
- When the PRF daemon is restarted, the buffer that was used during the previous session is reused and the trace collection level is inherited. If the buffer size has been changed, the previous buffer is deleted and a new one is re-created. If another process is accessing the buffer, this re-creation process fails.

  $PRFSPOOL/utt/prf/*prf-identifier*/spool/save/
- If the PRF daemon terminates abnormally, re-start only the PRF daemon.
- The trace, collected from the period of abnormal termination of the PRF daemon until the PRF daemon re-starts, is destroyed.
- If a big value is specified in -PrfTraceCount or -PrfTraceFileSize, the re-starting of the PRF daemon after the abnormal termination might take time.

5.2.3 cprfstop

The cprfstop command stops the PRF daemon.

Synopsis

```
cprfstop [-PRFID PRF-identifier] [-Force] [-h]
```

Storage location

Application Server installation directory/common/PRF/bin/

Function

This command stops the PRF daemon (cprfd). You should stop the PRF daemon only after all other processes have been stopped.

Execution permission

A user who has the following permission:

- Either the same user for the processes that obtain PRF or the superuser
- In an environment that uses the same PRFSPOOL environment variable, the superuser and other users coexist and the PRF commands cannot be executed. The PRF commands must always be executed by the same user.

Precondition

The same PRFSPOOL environment variable as the one assumed by the request destination PRF daemon must be set. Otherwise, the command returns an error.

Environment variable

- PATH

  Add *Application Server installation directory*/common/PRF/bin and /bin.

- LD_LIBRARY_PATH(Linux)

  Specify *Application Server installation directory*/common/PRF/lib.

- PRFSPOOL

  Specifies the execution environment directory of the performance tracer (PRF daemon).

  The PRFSPOOL environment variable is set by the domain administration server when the domain administration server starts a JAVA EE server or a cluster. If a PRF association whose association source is the Java EE server exists, the PRFSPOOL environment variable is set in the following format based on the PRF name of the association destination:

  "*Java-EE-Server-log-output-destination-directory-for-the-node-in-which-the-JAVA-EE-server-is-built*/nodes/*node-name*/*PRF-name*"

- TZ(UNIX)

  Specifies the time zone.

Arguments

`-PRFID` *`prf-identifier`*

For the PRF identifier, specify the PRF server name that is given when the PRF is created by using the `create-prf` subcommand of `asadmin`.

Specifies a PRF daemon identifier, as 1 to 31 alphanumeric characters and the underscore (_). Do not specify a character string that begins with `TSC` or `tsc` or with `CTM` or `ctm`.

Default value: `PRF_ID`

`-Force`

Forcibly stops the PRF daemon.

`-h`

Displays the command's usage.

Exit Status

Exit Status	Explanation
`0`	The PRF daemon stopped normally.
`1`	The PRF daemon is not active. Either the `PRFSPOOL` environment variable is different from the one assumed for the PRF daemon or the value of `-PRFID` is invalid.
`2`	The `PRFSPOOL` environment variable has not been set up.
`3`	A timeout occurred.
Other than the above	An error occurred during command processing. Take the corrective actions indicated in the output message, and then re-execute the command. The error message is output to the standard error output and a log file.

Notes

- If the command times out and the PRF daemon does not terminate normally, specify the `-Force` option to force the PRF daemon to stop.
- If a big value is specified in `-PrfTraceCount` or `-PrfTraceFileSize` of the `cprfstart` command, the normal termination of PRF daemon might take time.

5.3 Commands used for output the trace information

This section describes the syntax and functionality of the commands used for output the trace information.

5.3.1 cprfed

The cprfed command edits and outputs performance analysis trace information.

Synopsis

```
cprfed [-h]|[-Dump|-CSV] [-Time start-time,end-time]
    [-ProcessID process-id[,process-id...]]
    [-TraceFile prf-trace-file-name[,prf-trace-file-name...]
    |-AllTraceFile] [-Compact] [-AscLen number-of-characters-displayed-in-ASCII]
```

Storage location

Application Server installation directory/common/PRF/bin/

Function

This command inputs PRF trace information and edits and outputs performance analysis trace information.

If the PRF daemon is running, first execute the cprfflush command to write the trace information in the buffer, and then execute this command.

Execution permission

A user who has the following permission:

- Either the same user for the processes that obtain PRF or the superuser
- In an environment that uses the same PRFSPOOL environment variable, the superuser and other users coexist and the PRF commands cannot be executed. The PRF commands must always be executed by the same user.

Environment variable

- PATH

  Add *Application Server installation directory*/common/PRF/bin and /bin.
- LD_LIBRARY_PATH(Linux)

  Specify *Application Server installation directory*/common/PRF/lib.
- PRFSPOOL

  Specifies the execution environment directory of the performance tracer (PRF daemon).

  The PRFSPOOL environment variable is set by the domain administration server when the domain administration server starts a JAVA EE server or a cluster. If a PRF association whose association source is the Java EE server exists, the PRFSPOOL environment variable is set in the following format based on the PRF name of the association destination:

  "*Java-EE-Server-log-output-destination-directory-for-the-node-in-which-the-JAVA-EE-server-is-built*/nodes/*node-name*/*PRF-name*"
- TZ(UNIX)

Specifies the time zone.

Arguments

`-h`

Displays the command's usage.

`-Dump`

Outputs detailed information in dump format.

`-CSV`

Outputs detailed information in CSV format.

`-Time` *`start-time`*`,`*`end-time`*

Specifies a start and end time when only the trace information acquired within the specified time period is to be output.

The start time and end time are each specified in the format *`hhmmss`*`[`*`MMdd`*`[`*`yyyy`*`]].`*`MMdd`* and *`yyyy`* might be omitted; if omitted, the month and year in which the command is executed are assumed.

Time character string format: *`hhmmss`*`[`*`MMdd`*`[`*`yyyy`*`]]`

`hh`: Hour (`00`-`23`)

`mm`: Minute (`00`-`59`)

`ss`: Second (`00`-`59`)

`MM`: Month (`01`-`12`)

`dd`: Date (`01`-`31`)

`yyyy`: Year (`1970`-`20xx`)

The table below shows argument specification examples and the corresponding editing ranges when the command is executed on October 11, 2003.

Argument specification	Editing range
`-Time 102345,`	From 45 seconds past 10:23 on October 11, 2003, until the end of the trace file
`-Time ,102345`	From the start of the trace file until 45 seconds past 10:23 on October 11, 2003
`-Time 102345,102350`	From 45 seconds past 10:23 on October 11, 2003, until 50 seconds past 10:23 on October 11, 2003
`-Time 1023451012,1023501013`	From 45 seconds past 10:23 on October 12, 2003, until 50 seconds past 10:23 on October 13, 2003
`-Time 10234510122002,10235010132002`	From 45 seconds past 10:23 on October 12, 2002, until 50 seconds past 10:23 on October 13, 2002

`-ProcessID` *`process-id`*

Specifies a process ID when information for only the specified process ID is to be output. Specify the process ID as a decimal number.

`-TraceFile` *`prf-trace-file-name`*

Specifies a PRF trace file. A maximum of 256 trace files can be specified. This option cannot be specified together with the `-AllTraceFile` option.

Each PRF trace file is created automatically in the PRF trace file output destination directory. The file name of a PRF trace file is `prf_`*`???`*, where *`???`* is an integer between `001` and the maximum number of trace file sectors.

PRF trace file output destination directory:

`$PRFSPOOL/utt/prf/`*`prf-identifier`*`/dcopltrc/`

When specifying *prf-trace-file-name*, either you must specify an absolute path, or you must set the PRF trace file output destination directory as the current directory and specify a file name only.

`-AllTraceFile`

Loads all files under the current directory as input files. This option cannot be specified together with the `-TraceFile` option. Edit only 256 files if the current directory has more than 256 files. If the results of the `cprfed` command are redirected to the current directory, the redirected files are also counted as input files.

`-Compact`

Sets the character string length to 0 for the trace output of the following items:

Client AP IP, Client AP PID, Client AP CommNo., OPT, and ASCII

This option must be specified together with the -CVS option. If the -CVS option is not specified, this option is ignored.

`-AscLen` *number-of-characters-displayed-in-ASCII*

Specifies the maximum number of output characters for the `ASCII` output item in the dump information. If omitted, the character string length is set to 0.

The contents of the `ASCII` output item is always enclosed in double quotations. If a value 1 is specified, double quotation marks are added, and three characters are output. However, if the `ASCII` output item does not contain output information, the character string length will be 0.

This option must be used together with the -Compact option. If the -Compact option is not specified, this option is ignored and all information of the `ASCII` output item is displayed.

You can specify the following values:

- `0` to `512`

Default value: `0`

The table below shows the combinations of command options that can be specified.

Command option	-h	-Dump	-CSV	-Time	-ProcessID	-TraceFile	-AllTraceFile	-Compact
-Dump	#-h	--	--	--	--	--	--	--
-CSV	#-h	#-CSV	--	--	--	--	--	--
-Time	#-h	Y	Y	--	--	--	--	--
-ProcessID	#-h	Y	Y	Y	--	--	--	--
-TraceFile	#-h	Y	Y	Y	Y	--	--	--
-AllTraceFile	#-h	Y	Y	Y	Y	N	--	--
-Compact	#-h	#-Dump	Y	Y	Y	Y	Y	--
-AscLen	#-h	#-Dump	Y	Y	Y	Y	Y	Y

Legend:

Y: Can be specified simultaneously

#: Valid with the option whose name follows the #

N: Cannot be specified simultaneously

--: Not applicable

Exit Status

Exit Status	Explanation
0	Normal termination occurred.
Other than 0	An error occurred during command processing. Take the corrective action indicated in the output message, and then re-execute the command. The error message is output to the standard error output and a log file.

Output format

Output format 1 (dump format)

```
PRF: aaaaaa Process: bbbbbbbbbb Thread: ccccccccc(CC..CC) Trace: ddddddddd
ProcessName: vv..vv
Event: eeeee Time: ffff/gg/hh ii:jj:kk lll/mmm/nnn
Rc: oo..oo
ClientAP:pp..pp qq..qq - rr..rr RootAP: ss..ss tt..tt - uu..uu
INT: XX..XX OPR: YY..YY
Offset +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +a +b +c +d +e +f 0123456789abcdef
  AAAA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AAAAAAAAAAAAAAAA
  AAAA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AAAAAAAAAAAAAAAA
         :
  AAAA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AA AAAAAAAAAAAAAAAA
```

Output format 2 (CSV format)

```
PRF,Process,Thread(hashcode),Trace,ProcessName,Event,Date,Time,Time(msec/usec/
nsec),Rc,ClientAP IP,ClientAP PID, ClientAP CommNo.,RootAP IP,RootAP PID,RootAP
CommNo.,INT,OPR,OPT,ASCII
aaaaaa,bbbbbbbbbb,ccccccccc(CC..CC),ddddddddd,vv.vv,eeeee,ffff/gg/
hh,ii:jj:kk,lll/mmm/
nnn,oo..oo,pp..pp,qq..qq,rr..rr,ss..ss,tt..tt,uu..uu,XX..XX,YY..YY,AA..AA,BB..BB
```

Legend:

Symbol	Explanation
aa....aa	Record status (up to 6 characters): • Rec: Record status is normal • ErrRec: Record status is abnormal
bb....bb	Process ID of the process from which trace information was collected (up to 10 decimal digits)
cc....cc	Thread ID of the thread in the process from which trace information was collected (up to 18 hexadecimal digits)
dd....dd	Trace serial number in the thread (up to 10 decimal digits)
ee....ee	Event ID (6 hexadecimal digits (including the leading 0x))
ffff	Time at which the trace was collected (year)
gg	Time at which the trace was collected (month)
hh	Time at which the trace was collected (date)
ii	Time at which the trace was collected (hour)
jj	Time at which the trace was collected (minute)
kk	Time at which the trace was collected (second)
lll	Time at which the trace was collected (millisecond)

Symbol	Explanation
mmm	Time at which the trace was collected (microsecond)
nnn	Time at which the trace was collected (nanosecond)
oo....oo	Return code (10 hexadecimal digits (including the leading 0x))
pp....pp	Client application IP address (`xxx.xxx.xxx.xxx` format)
qq....qq	Client application process ID (up to 10 decimal digits)
rr....rr	Client communication number (18 digits (including the leading 0x))
ss....ss	Root application IP address (`xxx.xxx.xxx.xxx` format)
tt....tt	Root application process ID (up to 10 decimal digits)
uu....uu	Root communication number (18 digits (including the leading 0x))
vv....vv	Process name (up to 32 characters)
XX....XX	Interface name (up to 33 characters)#
YY....YY	Operation name (up to 33 characters)#
AA....AA	Outputs the information acquired in each event, in the dump format (maximum 514 characters). The dump format information is output enclosed in double quotation marks (`" "`).
BB....BB	Outputs the information acquired in each event, as ASCII characters (maximum 514 characters). ASCII characters in the dump format are output enclosed in double quotation marks (`" "`). A code (control code) that cannot be displayed as an ASCII character is output as a period (`.`). A double quotation mark (`"`) is output as two ASCII characters (`""`).
CC....CC	The hash value of a thread (up to 18 hexadecimal digits)

#

If the interface name or operation name exceeds 32 characters, the name is edited as the following and output in 33 characters:

first-16-characters*last-16-characters

first-32-characters*

*last-32-characters

Output examples

- If `cprfed -TraceFile prf_001` is executed:

```
PRF: Rec Process: 26006      Thread: 0x15(0x12345678)    Trace: 1023
ProcessName: tscd
Event: 0x1002 Time: 2000/02/12 13:43:44 363/200/000
Rc: 0
ClientAP: 172.17.113.19   26303      - 00000000000003ff RootAP: 172.17.113.19
26006       - 00000000000003ff
INT : testint                              OPR: test5
```

- If `cprfed -Dump -TraceFile prf_001` is executed:

```
PRF: Rec Process: 26006      Thread: 0x15(0x12345678)    Trace: 1023
ProcessName: tscd
Event: 0x1002 Time: 2000/02/12 13:43:44 363/200/000
Rc: 0
ClientAP: 172.17.113.19   26303        - 00000000000003ff RootAP:
172.17.113.19     26006       - 00000000000003ff
INT : testint                              OPR: test5
```

```
Offset  +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +a +b +c +d +e +f  0123456789abcdef
 +0000  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
 +0010  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
 +0020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
 +0030  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ............
```

- If `cprfed -CSV -TraceFile prf_001` is executed:

```
PRF,Process,Thread(hashcode),Trace,ProcessName,Event,Date,Time,Time(msec/usec/
nsec),Rc,ClientAP IP,ClientAP PID,ClientAP CommNo.,RootAP IP,RootAP PID,RootAP
CommNo.,INT,OPR,OPT,ASCII
Rec,1800,0x2e00(0x0e9576fd),1,Client,0x1009,2003/10/12,16:26:52,734/000/000,****,
10.209.15.65,1800,0x0000000000000001,10.209.15.65,1800,0x0000000000000001,ABC,****
,abcdefg,"31323334"
```

Notes

- In the performance analysis trace, the exclusion control is not performed during the PRF trace output, so that the bottlenecks do not occur because of the exclusion during high load. Therefore, some of the trace information might not be output properly.
- If a character other than an ASCII character is included in a file edited to CSV format, use an editor that supports Unicode, or Excel 2003 or later, to reference the file.

5.3.2 cprfflush

forcibly output buffer contents to a file

Synopsis

```
cprfflush [-PRFID prf-identifier] [-h]
```

Storage location

Application Server installation directory/common/PRF/bin/

Function

This command requests the PRF daemon to output the trace information stored in the buffer.

The output range begins with the trace following the last trace that was output during the previous file output operation to the most recently collected trace.

Execution permission

A user who has the following permission:

- Either the same user for the processes that obtain PRF or the superuser
- In an environment that uses the same PRFSPOOL environment variable, the superuser and other users coexist and the PRF commands cannot be executed. The PRF commands must always be executed by the same user.

Precondition

- The `cprfflush` command cannot be executed while another instance of the `cprfflush` command is executing.

- The same PRFSPOOL environment variable as the one assumed by the request destination PRF daemon must be set. Otherwise, the command returns an error.

Environment variable

- PATH

  Add *Application Server installation directory*/common/PRF/bin and /bin.

- LD_LIBRARY_PATH(Linux)

  Specify *Application Server installation directory*/common/PRF/lib.

- PRFSPOOL

  Specifies the execution environment directory of the performance tracer (PRF daemon).

  The PRFSPOOL environment variable is set by the domain administration server when the domain administration server starts a JAVA EE server or a cluster. If a PRF association whose association source is the Java EE server exists, the PRFSPOOL environment variable is set in the following format based on the PRF name of the association destination:

  "*Java-EE-Server-log-output-destination-directory-for-the-node-in-which-the-JAVA-EE-server-is-built*/nodes/*node-name*/*PRF-name*"

- TZ(UNIX)

  Specifies the time zone.

Arguments

-PRFID *prf-identifier*

For the PRF identifier, specify the PRF server name that is given when the PRF is created by using the create-prf subcommand of asadmin.

Specifies a PRF daemon identifier, as 1 to 31 alphanumeric characters and the underscore (_). Do not specify a character string that begins with TSC or tsc or with CTM or ctm.

Default value: PRF_ID

-h

Displays the command's usage.

Exit Status

Exit Status	Explanation
0	The command terminated normally.
1	The PRF daemon is not active. Either the PRFSPOOL environment variable is different from the one assumed for the PRF daemon or the value of -PRFID is invalid.
2	The PRFSPOOL environment variable has not been set up.
Other than the above	An error occurred during command processing. Take the corrective action indicated in the output message, and then re-execute the command. The error message is output to the standard error output and a log file.

5.3.3 cprflevel

The cprflevel command displays or changes the PRF trace collection level.

Synopsis

```
cprflevel [-PRFID prf-identifier] [-h] [-PrfChangeLevel
    prf-trace-collection-level[,[prf-trace-collection-level]...]]
    [-PrfLevelIndex index-number] [-PrfLevelAll]
```

Storage location

Application Server installation directory/common/PRF/bin/

Function

This command displays or changes the PRF trace collection level.

When the -PrfChangeLevel option is omitted, the PRF trace collection level set for the specified PRF identifier is displayed. When the -PrfChangeLevel option is specified, the PRF trace collection level set for the specified PRF identifier is changed to the value specified here.

When the -PrfLevelIndex option is specified, a list of PRF trace collection levels for the specified index is displayed. When the -PrfLevelAll option is specified, the PRF trace collection levels for all indexes are displayed.

Execution permission

A user who has the following permission:

- Either the same user for the processes that obtain PRF or the superuser
- In an environment that uses the same PRFSPOOL environment variable, the superuser and other users coexist and the PRF commands cannot be executed. The PRF commands must always be executed by the same user.

Precondition

- You must execute this command while the PRF daemon is active. If this command is executed while the PRF daemon is not active, an error message is output.

  The same PRFSPOOL environment variable as the one assumed by the request destination PRF daemon must be set.

Environment variable

- PATH

  Add *Application Server installation directory*/common/PRF/bin and /bin.

- LD_LIBRARY_PATH(Linux)

  Specify *Application Server installation directory*/common/PRF/lib.

- PRFSPOOL

  Specifies the execution environment directory of the performance tracer (PRF daemon).

  The PRFSPOOL environment variable is set by the domain administration server when the domain administration server starts a JAVA EE server or a cluster. If a PRF association whose association source is the Java EE server exists, the PRFSPOOL environment variable is set in the following format based on the PRF name of the association destination:

  "*Java-EE-Server-log-output-destination-directory-for-the-node-in-which-the-JAVA-EE-server-is-built*/nodes/*node-name*/*PRF-name*"

- TZ(UNIX)

Specifies the time zone.

Arguments

`-PRFID` *`prf-identifier`*

For the PRF identifier, specify the PRF server name that is given when the PRF is created by using the `create-prf` subcommand of `asadmin`.

Specifies a PRF daemon identifier, as 1 to 31 alphanumeric characters and the underscore (_). Do not specify a character string that begins with `TSC` or `tsc` or with `CTM` or `ctm`.

Default value: `PRF_ID`

`-h`

Displays the command's usage.

`-PrfChangeLevel` *`prf-trace-collection-level`* `[,[`*`prf-trace-collection-level`*`]...]`

Specifies PRF trace collection levels, each as a 4-byte hexadecimal number (8-digit value). You can add 0x at the beginning of the 4-byte hexadecimal number, however, 0x will be ignored.

You can specify multiple trace collection level values, corresponding to the index numbers (left to right), delimited by a comma (`,`). If a level value specification is to be omitted, do not specify any value. To omit specification of level values for all index numbers beginning after a specific index number, omit all the subsequent comma delimiters. To set the PRF trace collection level for index No. 1 to (1) and the PRF trace collection level for index No. 2 to (2), specify the arguments as (1),(2),....

The table below shows examples of PRF trace collection level specifications.

PRF trace collection level specification example	Option specification
Specify a PRF trace collection level for index No. 1	`-PrfTraceLevel 0x44445555`
Specify PRF trace collection levels for index Nos. 1 and 2	`-PrfTraceLevel 0x44445555,0x55554444`
Specify a PRF trace collection level for index No. 2	`-PrfTraceLevel ,0x55554444`

Assignment of PRF trace collection levels at each functionality layer is explained here.

A PRF trace collection level is specified as an eight-digit hexadecimal number for each index. Each digit is allocated two functionality layers. When a hexadecimal number is expressed as a decimal number, the functionality layer allocated to the upper 2 bits is called the upper layer, and the functionality layer allocated to the lower 2 bits is called the lower layer.

The table below shows the functionality layers that are allocated to the upper and lower layers for each digit.

Index	Digits	Upper layer	Lower layer
Index 1	Digit 1	(None)	Java EE Server
	Digit 2	(None)	Java VM
	Digit 5	(None)	uCosminexus TP1 Connector, TP1/Client/J

To specify a PRF trace collection level, decide whether to specify Standard, Detail, or Maintenance for the PRF trace collection level of the upper and lower layers for each digit, and specify it as a hexadecimal number. The table below shows the correspondence between combinations of PRF trace collection levels for the upper and lower layers with specification values.

Note that because the Maintenance level is used for collecting maintenance information when an error occurs, it should not be specified during normal operation.

Upper layer	Lower layer	Specification value
Standard	Standard	0

Upper layer	Lower layer	Specification value
Standard	Detail	1
Standard	Maintenance	2
Detail	Standard	4
Detail	Detail	5
Detail	Maintenance	6
Maintenance	Standard	8
Maintenance	Detail	9
Maintenance	Maintenance	a

Specification examples are shown below.

Index	Specification example	Explanation
Index 1	`00000000`	Acquires Standard-level trace information from each functionality layer of index 1.
	`10000000`	Acquires Detail-level trace information from Java EE Server layer only, and acquires Standard-level trace information from other functionality layers.
	`11000000`	Acquires Detail-level trace information from both Java EE Server and the Java VM layers.

`-PrfLevelIndex` *`index-number`*

Specifies the index number of the PRF trace collection level to be changed or referenced, as an integer in the range 1 to 16.

Default value: `1`

When this argument is specified together with `-PrfChangeLevel`, the PRF trace collection level for the specified index number is changed to the value specified in `-PrfChangeLevel`.

When this argument is specified but `-PrfChangeLevel` is not specified, the PRF trace collection level for the specified index number is displayed.

When you specify this argument together with `-PrfChangeLevel`, be sure to specify only a single value in `-PrfChangeLevel`.

Example: Change the value for the second index number to 0x55554444

```
cprflevel -PrfChangeLevel 0x55554444 -PrfLevelIndex 2
```

If multiple values are specified in `-PrfChangeLevel`, an option error occurs.

Example: Specify multiple values in `-PrfChangeLevel`

```
cprflevel -PrfChangeLevel ,0x55554444,0x55554444 -PrfLevelIndex 2
```

This is because multiple values are specified in `-PrfChangeLevel` even though `2` is specified in `-PrfLevelIndex`. If the specification of `-PrfLevelIndex` is omitted, `1` is assumed.

`-PrfLevelAll`

Displays all level values. This option cannot be specified together with `-PrfChangeLevel` or `-PrfLevelIndex`. The table below shows the combinations of this option and `-PrfLevelIndex`.

cprflevel current level value display method		-PrfLevelIndex	
		Specified	Not specified
-PrfLevelAll	Specified	Option error	Displays all level values

cprflevel current level value display method		-PrfLevelIndex	
		Specified	Not specified
	Not specified	Displays the level value of only the specified index	Displays the level value of index 1 only

The table below shows the command option combinations that can be specified.

Command option	-h	-PRFID	-PrfChangeLevel	-PrfLevelIndex	-PrfLevelAll
-h	--	#-h	#-h	#-h	#-h
-PRFID	#-h	--	Y	Y	Y
-PrfChangeLevel	#-h	Y	--	Y	N
-PrfLevelIndex	#-h	Y	Y	--	N
-PrfLevelAll	#-h	Y	N	N	--

Legend:

Y: Can be specified simultaneously

#: Valid with the option whose name follows the #

N: Cannot be specified simultaneously

--: Not applicable

Input/output examples

- Input example 1
  ```
  cprflevel
  ```
- Output example 1
  ```
  KFCT73415-I 2661 1: prf trace level is 0x12345678
  ```
- Input example 2
  ```
  cprflevel -PrfLevelAll
  ```
- Output example 2
  ```
  KFCT73418-I 2661 1: all prf trace levels are:
  [ 1]=0x12345678 [ 2]=0x00000000 [ 3]=0x00000000 [ 4]=0x00000000
  [ 5]=0x00000000 [ 6]=0x00000000 [ 7]=0x00000000 [ 8]=0x00000000
  [ 9]=0x00000000 [10]=0x00000000 [11]=0x00000000 [12]=0x00000000
  [13]=0x00000000 [14]=0x00000000 [15]=0x00000000 [16]=0x00000000
  ```
- Input example 3
  ```
  cprflevel -PrfLevelIndex 5
  ```
- Output example 3
  ```
  KFCT73419-I 2661 1: prf trace level of [5] is 0x00000000.
  ```
- Input example 4
  ```
  cprflevel -PrfChangeLevel 000000001
  ```
- Output example 4
  ```
  KFCT73416-I 2661 1: prf trace level was changed from 0x00000000 to
  0x00000001.
  ```
- Input example 5
  ```
  cprflevel -PrfChangeLevel 0x00000055,0x55554444
  ```

- Output example 5

```
KFCT73420-I 2661 1: prf trace level was changed:
[ 1]=0x00000000 -> [ 1]=0x00000055
[ 2]=0x00000000 -> [ 2]=0x55554444
```

- Input example 6

```
cprflevel -PrfChangeLevel 0x44445555 -PrfLevelIndex 2
```

- Output example 6

```
KFCT73420-I 2661 1: prf trace level was changed: [ 2]=0x00000000 ->
[ 2]=0x44445555
```

Exit Status

Exit Status	Explanation
0	The command terminated normally.
1	The PRF daemon is not active. Either the PRFSPOOL environment variable is different from the one assumed for the PRF daemon or the value of -PRFID is invalid.
2	The PRFSPOOL environment variable has not been set up.
Other than the above	An error occurred during command processing. Take the corrective action indicated in the output message, and then re-execute the command. The error message is output to the standard error output and a log file.

Notes

- When the current PRF trace collection level is to be displayed, the message that is output is KFCT73415-I, KFCT73418-I, or KFCT73419-I depending on the combination of the -PrfLevelIndex and -PrfLevelAll specifications The following shows the message that is output:

Message output for displaying PRF trace collection level		-PrfLevelAll	
		Specified	Not specified
-PrfLevelIndex	Specified	-	KFCT73419-I
	Not specified	KFCT73418-I	KFCT73415-I

- When changing the PRF trace collection level, the message that is output by the -PrfChangeLevel specification is either KFCT73416-I or KFCT73420-I. When both of the conditions listed below are satisfied, KFCT73416-I is output; otherwise, KFCT73420-I is output:
  - Only a single value is specified, as in -PrfChangeLevel 0x44445555, and no comma (,) is used.
  - -PrfLevelIndex is not specified.
- When you use this command to change levels, the changes will be applied from the next request onwards.

Appendix

A. Commands used as counter measures

This section describes the syntax and functionality of the commands used as counter measures.

A.1 apsversion

The `apsversion` command displays the versions of configured Application Server products.

Synopsis

```
apsversion
```

Storage location

Application Server installation directory`/common/bin/`

Function

The `apsversion` command checks the versions and revisions of Application Server products. This command checks the versions and revisions of products below Application Server product installation directory containing this command.

Execution permission

General user

Precondition

Application Server products have been installed successfully.

Output Format

The output data is displayed in the following format in the standard output of the command prompt:

```
Display Name : display_name
product_type1 product_version.revision1 product_name1
product_type2 product_version.revision2 product_name2
product_type3 product_version.revision3 product_name3
:
```

Example of output:

```
Display Name : APServer
P-2443-5KA4   1000  Hitachi Application Server
P-CC2443-C1A4 1000  Hitachi Application Server - Base
P-CC2443-C2A4 1000  Hitachi Developer's Kit for Java
P-CC2443-C3A4 1000  Hitachi Web Server
P-CC2443-C4A4 1000  Hitachi Java EE Server
```

Index

D

E

F

G

H

I

J

L

M

O

P

R

S

T

U

V

W

X