

◆ ◆ ◆ **4**
CHAPTER 4

Administering Named Configurations

A *named configuration* is a set of configuration information for GlassFish Server instances and clusters. A configuration sets port numbers for listeners that an instance or a cluster uses and defines settings for items such as the EJB container, security, logging, and monitoring. Applications and resources are *not* defined in named configurations.

GlassFish Server enables you to create configurations for use by clusters and instances. GlassFish Server creates configurations automatically for clusters and instances that do not reference an existing configuration. You can modify any existing named configuration regardless of how the configuration was created.

The following topics are addressed here:

- “About Named Configurations” on page 69
- “Creating, Listing, and Deleting Named Configurations” on page 71
- “Modifying Properties for Named Configurations and Instances” on page 75

About Named Configurations

Configurations exist in a domain. Multiple GlassFish Server instances or clusters in the domain can reference the same configuration, or they can have separate configurations. To ensure that the environment in a cluster’s instances is homogenous, all instances in the cluster inherit the cluster’s configuration.

Types of Named Configurations

Each named configuration is one of the following types of configuration:

Standalone configuration

A standalone configuration is referenced by only one instance or cluster and is not shared with any other instances or clusters.

Note – A configuration that is referenced by only one cluster is a standalone configuration, even if the cluster contains multiple instances.

Shared configuration

A shared configuration is referenced by multiple instances or clusters.

The type of an unclustered instance is determined by the type of the configuration that the instance references. For more information, see [“About GlassFish Server Instances” on page 47](#).

The default-config Configuration

The default-config configuration is a special configuration that acts as a template for creating named configurations. Clusters and instances cannot refer to the default-config configuration. The default-config configuration can only be copied to create configurations.

Automatically Created Configurations

When you create a cluster or an instance, you can choose whether to specify an existing configuration that the new cluster or instance will reference. If you choose to create a cluster or an instance without specifying an existing configuration, GlassFish Server automatically creates a configuration for the cluster or instance. For more information, see the following sections:

- [“To Create a Cluster” on page 40](#)
- [“To Create an Instance Centrally” on page 48](#)
- [“To Create an Instance Locally” on page 58](#)

GlassFish Server automatically creates a configuration by copying the default-config configuration. If you require an instance or cluster to reference a copy of a different configuration, copy the configuration and specify the copy when you create the instance or cluster. For information about how to copy a configuration, see [“To Create a Named Configuration” on page 71](#).

GlassFish Server assigns the name *cluster-or-instance-config* to an automatically created configuration. *cluster-or-instance* is the name of the cluster or instance for which the configuration is created.

Directory for Configuration Synchronization

When a named configuration is created, GlassFish Server creates a configuration directory on the domain administration server (DAS) at *domain-dir/config/config-name*.

domain-dir

The directory in which the domain's configuration is stored.

config-name

The name that was assigned to the configuration when the configuration was created.

This directory is used to synchronize configurations for all instances that reference the configuration.

Creating, Listing, and Deleting Named Configurations

GlassFish Server enables you to create configurations for use by clusters and instances, obtain information about configurations, and delete configurations that are no longer required.

The following topics are addressed here:

- [“To Create a Named Configuration” on page 71](#)
- [“To List the Named Configurations in a Domain” on page 72](#)
- [“To List the Targets of a Named Configuration” on page 73](#)
- [“To Delete a Named Configuration” on page 74](#)

▼ To Create a Named Configuration

Use the `copy-config` subcommand in remote mode to create a named configuration by copying an existing configuration.

GlassFish Server requires you to create a configuration by copying a configuration because a configuration contains many required settings. The newly created configuration is identical to the configuration that you copy until you change its configuration settings.

Tip – Create a named configuration *only* if you plan to share the configuration among multiple unclustered instances or clusters, or if you are using a configuration to preconfigure GMS settings. Otherwise, create clusters and instances without specifying an existing configuration. If no configuration is specified, GlassFish Server creates a copy of the default configuration for the cluster or instance.

For more information, see the following sections:

- [“To Preconfigure Nondefault GMS Configuration Settings” on page 34](#)
 - [“To Create a Cluster” on page 40](#)
 - [“To Create an Instance Centrally” on page 48](#)
 - [“To Create an Instance Locally” on page 58](#)
-

1 Ensure that the DAS is running.

Remote subcommands require a running server.

2 Run the `copy-config` subcommand.

Note – Only the options that are required to complete this task are provided in this step. For information about all the options for configuring the named configuration, see the `copy-config(1)` help page.

```
asadmin> copy-config source-config-name destination-config-name
```

source-config-name

The name of the configuration that you are copying. You must specify a configuration to copy even if you are copying the default configuration. The configuration must already exist.

destination-config-name

Your choice of name for the configuration that you are creating by copying the source configuration.

Example 4-1 Creating a Named Configuration

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

```
asadmin> copy-config default-config clusterpresets-config  
Command copy-config executed successfully.
```

- See Also**
- [“To Preconfigure Nondefault GMS Configuration Settings” on page 34](#)
 - [“To Create a Cluster” on page 40](#)
 - [“To Create an Instance Centrally” on page 48](#)
 - [“To Create an Instance Locally” on page 58](#)
 - `copy-config(1)`

You can also view the full syntax and options of the subcommand by typing `asadmin help copy-config` at the command line.

▼ To List the Named Configurations in a Domain

Use the `list-configs` subcommand in remote mode to list existing named configurations in a domain.

- 1 Ensure that the DAS is running.**
Remote subcommands require a running server.
- 2 Run the `list-configs(1)` subcommand.**

```
asadmin> list-configs
```

Example 4-2 Listing All Named Configurations in a Domain

This example lists all named configurations in the current domain.

```
asadmin> list-configs
server-config
default-config
ymlcluster-config
clusterpresets-config
Command list-configs executed successfully.
```

See Also `list-configs(1)`

You can also view the full syntax and options of the subcommand by typing `asadmin help list-configs` at the command line.

▼ To List the Targets of a Named Configuration

Use the `list-clusters` subcommand and the `list-instances` subcommand in remote mode to list the targets of a named configuration.

The targets of a named configuration are the clusters and GlassFish Server instances that reference the configuration.

1 Ensure that the DAS is running.

Remote subcommands require a running server.

2 List the clusters that refer to the configuration.

```
asadmin> list-clusters config-name
config-name
    The name of the configuration whose targets you are listing.
```

3 List the instances that refer to the configuration.

```
asadmin> list-instances config-name
config-name
    The name of the configuration whose targets you are listing.
```

Example 4-3 Listing the Targets of a Named Configuration

This example shows that the cluster `ymlcluster` and the instances `yml-i1` and `yml-i2` reference the named configuration `ymlcluster-config`.

```
asadmin> list-clusters ymlcluster-config
ymlcluster partially running
Command list-clusters executed successfully.
```

```
asadmin> list-instances ymlcluster-config
yml-i1    running
yml-i2    not running
Command list-instances executed successfully.
```

- See Also**
- `list-clusters(1)`
 - `list-instances(1)`

You can also view the full syntax and options of the subcommands by typing the following commands at the command line:

- `asadmin help list-clusters`
- `asadmin help list-instances`

▼ To Delete a Named Configuration

Use the `delete-config` subcommand in remote mode to delete an existing named configuration from the configuration of the DAS.

You cannot delete the `default-config` configuration.

Note – A standalone configuration that was created automatically for a cluster or a GlassFish Server instance is deleted when the cluster or instance is deleted.

Before You Begin Ensure that no clusters or instances refer to the configuration. If a cluster or instance refers to the configuration and is no longer required, delete the cluster or instance. For information about how to delete an instance and how to delete a cluster, see the following sections:

- [“To Delete an Instance Centrally” on page 52](#)
- [“To Delete an Instance Locally” on page 61](#)
- [“To Delete a Cluster” on page 43](#)

1 Ensure that the DAS is running.

Remote subcommands require a running server.

2 Confirm that no clusters refer to the configuration that you are deleting.

```
asadmin> list-clusters config-name
```

```
config-name
```

The name of the configuration that you are deleting.

3 Confirm that no instances refer to the configuration that you are deleting.

```
asadmin> list-instances config-name
```

config-name

The name of the configuration that you are deleting.

4 Run the `delete-config(1)` subcommand.

```
asadmin> delete-config config-name
```

config-name

The name of the configuration that you are deleting.

Example 4-4 Deleting a Named Configuration

This example confirms that no clusters or instances refer to the configuration `clusterpresets-config` and then deletes the configuration.

```
asadmin> list-clusters clusterpresets-config
Nothing to list
Command list-clusters executed successfully.
asadmin> list-instances clusterpresets-config
Nothing to list.
Command list-instances executed successfully.
asadmin> delete-config clusterpresets-config
Command delete-config executed successfully.
```

- See Also**
- [“To Delete an Instance Centrally” on page 52](#)
 - [“To Delete an Instance Locally” on page 61](#)
 - [“To Delete a Cluster” on page 43](#)
 - `delete-config(1)`
 - `list-clusters(1)`
 - `list-instances(1)`

You can also view the full syntax and options of the subcommands by typing the following commands at the command line:

- `asadmin help delete-config`
- `asadmin help list-clusters`
- `asadmin help list-instances`

Modifying Properties for Named Configurations and Instances

The properties in a named configuration define port numbers for unclustered instances that reference the configuration or clustered instances that inherit the configuration. An instance initially obtains port numbers from the configuration that the instance references or inherits. To avoid port conflicts, edit the properties of named configurations and instances.

The following topics are addressed here:

- “Properties for Port Numbers in a Named Configuration” on page 76
- “To Modify a Named Configuration’s Properties” on page 77
- “To Modify Port Numbers of an Instance” on page 78

Properties for Port Numbers in a Named Configuration

The properties in a named configuration that define port numbers are as follows:

ASADMIN_LISTENER_PORT

This property 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. On UNIX, creating sockets that listen on ports 1–1024 requires superuser privileges.

HTTP_LISTENER_PORT

This property specifies the port number of the port that is used to listen for HTTP requests. Valid values are 1–65535. On UNIX, creating sockets that listen on ports 1–1024 requires superuser privileges.

HTTP_SSL_LISTENER_PORT

This property specifies the port number of the port that is used to listen for HTTPS requests. Valid values are 1–65535. On UNIX, creating sockets that listen on ports 1–1024 requires superuser privileges.

IIOP_LISTENER_PORT

This property specifies the port number of the port that is used for IIOP connections. Valid values are 1–65535. On UNIX, creating sockets that listen on ports 1–1024 requires superuser privileges.

IIOP_SSL_LISTENER_PORT

This property specifies the port number of the port that is used for secure IIOP connections. Valid values are 1–65535. On UNIX, creating sockets that listen on ports 1–1024 requires superuser privileges.

IIOP_SSL_MUTUALAUTH_PORT

This property specifies the port number of the port that is used for secure IIOP connections with client authentication. Valid values are 1–65535. On UNIX, creating sockets that listen on ports 1–1024 requires superuser privileges.

JAVA_DEBUGGER_PORT

This property specifies the port number of the port that is used for connections to the [Java Platform Debugger Architecture \(JPDA\)](#) debugger. Valid values are 1–65535. On UNIX, creating sockets that listen on ports 1–1024 requires superuser privileges.

JMS_PROVIDER_PORT

This property specifies the port number for the Java Message Service provider. Valid values are 1–65535. On UNIX, creating sockets that listen on ports 1–1024 requires superuser privileges.

JMX_SYSTEM_CONNECTOR_PORT

This property specifies the port number on which the JMX connector listens. Valid values are 1–65535. On UNIX, creating sockets that listen on ports 1–1024 requires superuser privileges.

OSGI_SHELL_TELNET_PORT

This property specifies the port number of the port that is used for connections to the [Apache Felix Remote Shell](#). This shell uses the Felix shell service to interact with the OSGi module management subsystem. Valid values are 1–65535. On UNIX, creating sockets that listen on ports 1–1024 requires superuser privileges.

▼ To Modify a Named Configuration's Properties

Use the `get` subcommand and the `set` subcommand in remote mode to modify a named configuration's properties.

You might copy a configuration for use by instances that reside on the same host as instances that refer to the original configuration. In this situation, edit the properties of one of the configurations to ensure that instances that will refer to the configuration have the correct initial settings.

If you change the port number in a configuration, the port number is changed for any instance that references or inherits the configuration.

- 1 Ensure that the DAS is running.**
Remote subcommands require a running server.
- 2 For each property that you are modifying, determine the current value and set the new value.**

- a. Determine the current value of the property.**

```
asadmin> get configs.config.config-name.system-property.property-name.value
```

config-name

The name of the configuration whose properties you are modifying.

property-name

The name of the property that you are modifying. For a list of available properties, see [“Properties for Port Numbers in a Named Configuration”](#) on page 76.

- b. Set the property to its new value.**

```
asadmin> set  
configs.config.config-name.system-property.property-name.value=new-value
```

config-name

The name of the configuration whose properties you are modifying.

property-name

The name of the property that you are modifying. For a list of available properties, see [“Properties for Port Numbers in a Named Configuration”](#) on page 76.

new-value

The value to which you are setting the property.

Example 4–5 **Modifying a Property of a Named Configuration**

This example changes the value of the `JMS_PROVIDER_PORT` property in the `clusterpresets-config` configuration from 27676 to 27678.

```
asadmin> get
configs.config.clusterpresets-config.system-property.JMS_PROVIDER_PORT.value
configs.config.clusterpresets-config.system-property.JMS_PROVIDER_PORT.value=27676
Command get executed successfully.
asadmin> set
configs.config.clusterpresets-config.system-property.JMS_PROVIDER_PORT.value=27678
configs.config.clusterpresets-config.system-property.JMS_PROVIDER_PORT.value=27678
Command set executed successfully.
```

- See Also**
- `get(1)`
 - `set(1)`

You can also view the full syntax and options of the subcommands by typing the following commands at the command line:

- `asadmin help get`
- `asadmin help set`

▼ To Modify Port Numbers of an Instance

Use the `get` subcommand and the `set` subcommand in remote mode to modify the port numbers of an instance.

The port numbers of an instance are initially set in the configuration that the instance references or inherits from its parent cluster. Multiple instances that reside on the same host must each listen on a unique port number. Therefore, if multiple instances that reference or inherit the same configuration reside on the same host, a port conflict prevents all except one of the instances from starting. To avoid port conflicts, modify the port numbers on which individual instances listen.

If you modify an instance's port number and later modify the port number in the instance's configuration, the instance's port number remains unchanged.

The port numbers of an instance are stored as Java system properties. These properties are passed to the Java application launcher through the `-D` option of the Java application launcher when GlassFish Server is started.

1 Ensure that the DAS is running.

Remote subcommands require a running server.

2 For each port number that you are modifying, determine the current value and set the new value.**a. Determine the current value of the port number.**

```
asadmin> get  
servers.server.instance-name.system-property.port-property.value
```

instance-name

The name of the instance whose port numbers you are modifying.

port-property

The name of the property that corresponds to the port number that you are modifying.

For a list of available properties, see [“Properties for Port Numbers in a Named Configuration” on page 76](#).

b. Set the port number to its new value.

```
asadmin> get  
servers.server.instance-name.system-property.port-property.value=new-value
```

instance-name

The name of the instance whose port numbers you are modifying.

port-property

The name of the property that corresponds to the port number that you are modifying.

For a list of available properties, see [“Properties for Port Numbers in a Named Configuration” on page 76](#).

new-value

The value to which you are setting the port number.

Example 4–6 Modifying a Port Number for an Instance

This example changes the port number of the HTTP port or the HTTPS port for administration of the `pmdsainst` instance from 24849 to 24859.

```
asadmin> get  
servers.server.pmdsainst.system-property.ASADMIN_LISTENER_PORT.value  
servers.server.pmdsainst.system-property.ASADMIN_LISTENER_PORT.value=24849  
Command get executed successfully.  
asadmin> set  
servers.server.pmdsainst.system-property.ASADMIN_LISTENER_PORT.value=24859  
servers.server.pmdsainst.system-property.ASADMIN_LISTENER_PORT.value=24859  
Command set executed successfully.
```

- See Also**
- `get(1)`
 - `set(1)`

You can also view the full syntax and options of the subcommands by typing the following commands at the command line:

- `asadmin help get`
- `asadmin help set`