Sun N1 Service Provisioning System User's Guide for Sun Java System App Server Plug-In 3.0

Sun

microsystems

Contents

Preface

This book explains how to use the Sun N1 Service Provisioning System software to capture and deploy Sun Java System App Server applications and files.

Who Should Use This Book

The main audience for the Sun N1 Service Provisioning System User's Guide for Sun Java System App Server Plug-In 3.0 includes system administrators and operators of Sun N1 Service Provisioning System software who want to be able to incorporate Sun Java System App Server functionality with Sun N1 Service Provisioning System software. These users are expected to have the following background:

- Familiar with the Sun N1 Service Provisioning System product
- Familiar with standard UNIX® commands and utilities
- Familiar with the general concepts and management features available in the Sun Java System App Server product

Before You Read This Book

If you are not already familiar with using the Sun N1 Service Provisioning System software, read the following books:

- *Sun N1 Service Provisioning System 5.2 System Administration Guide*
- *Sun N1 Service Provisioning System 5.2 Operation and Provisioning Guide*
- *Sun N1 Service Provisioning System 5.2 Release Notes*

How This Book Is Organized

Chapter 1 provides an overview of the plug-in solution.

Chapter 2 contains installation and runtime issues.

Chapter 3 explains how to install and configure the plug-in.

Chapter 4 explains how to capture and deploy applications and files through the plug-in and describes the specific component types that are provided with the plug-in.

Related Third-Party Web Site References

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

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

Documentation, Support, and Training

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

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

Typographic Conventions

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

TABLE P–1 Typographic Conventions

Typeface	Meaning	Example
AaBbCc123	The names of commands, files, and directories, and onscreen computer output	Edit your `.login` file.
		Use `ls -a` to list all files.
		`machine_name% you have mail.`
AaBbCc123	What you type, contrasted with onscreen computer output	`machine_name%` **`su`**
		`Password:`
aabbcc123	Placeholder: replace with a real name or value	The command to remove a file is `rm` *filename*.

TABLE P–1 Typographic Conventions   *(Continued)*

Typeface	Meaning	Example
AaBbCc123	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the *User's Guide*.
		A *cache* is a copy that is stored locally.
		Do *not* save the file.
		Note: Some emphasized items appear bold online.

Shell Prompts in Command Examples

The following table shows the default UNIX system prompt and superuser prompt for the C shell, Bourne shell, and Korn shell.

TABLE P–2 Shell Prompts

Shell	Prompt
C shell	`machine_name%`
C shell for superuser	`machine_name#`
Bourne shell and Korn shell	`$`
Bourne shell and Korn shell for superuser	`#`

1

Overview of Sun Java System App Server Plug-In

This chapter explains general information about using Sun N1 Service Provisioning System to provision Sun Java System App Server applications. This chapter includes the following information:

Purpose of the Sun Java System App Server Plug-In

The Sun N1 Service Provisioning System software provides enhanced capabilities in out-of-the-box support for Sun Java System App Server applications. You can capture an Application Server application from a reference server, select precisely how this application should be configured, and deploy the application to standalone, managed, or clustered Application Server environments.

The following benefits are achieved when using the provisioning system rather than performing Application Server installations and deployments manually.

- Installation and deployment can be scheduled to multiple hosts.
- Logging and reporting that creates detailed logs of every action taken by the system across all applications and managed servers. These logs provide a complete audit history of every change made to every host.

What the Sun Java System App Server Plug-In Includes

The Sun Java System Application Server Plug-In 3.0 includes several Sun Java System App Server-specific components that allow Sun Java System App Server applications to be easily captured, configured, and deployed.

Requirements for Using the Sun Java System App Server Plug-In

Any host on which you intend to deploy the Sun Java System App Server Plug-In must meet the following requirements:

Operating system   The Sun Java System App Server Plug-In runs on the following operating systems:

- Solaris 8 for SPARC
- Solaris 9 for SPARC
- Solaris 10 for SPARC
- Solaris 9 for x86
- Solaris 10 for x86
- Redhat Enterprise Linux 2.1 Update 2
- Redhat Enterprise Linux 3.0 Update 1

Disk space   On Solaris you must have at least 250 MB of free disk space (500 MB recommended). On Redhat Enterprise Linux you must have at least 220 MB of free disk space (300 MB recommended).

RAM   512 MB minimum; 1 GB recommended

Java Runtime Environment   J2SE 1.4.2_06 or J2SE 5.0

Sun Java System App Server   The product and standalone installer files for Sun Java System App Server 8.1 2005Q1 or 2005Q2

Web Servers   Refer to "Web Servers" in *Sun Java System Application Server Enterprise Edition 8.1 2005Q1 Release Notes* or "Web Servers" in *Sun Java System Application Server Enterprise Edition 8.1 2005Q2 Release Notes* for supported web servers

2

Release Notes for the Sun Java System App Server Plug-In

This chapter describes late-breaking news and known issues with the Sun Java System App Server Plug-In.

The chapter contains the following information:

Installation Issues

There are no known issues when installing the Sun Java System App Server using the plug-in.

Runtime Issues

The following are known issues when running the Sun Java System App Server plug-in.

The Load Balancer Configuration File Is Not Updated Correctly When Multiple Clusters are Created in a Domain (6233520)

When starting a cluster, the load balancer configuration file, loadbalander.xml, is updated. If more than one cluster is configured for load balancing, then the last started cluster will be the cluster that is load balanced.

Load Balancer Configuration Entries Are Not Deleted When a Cluster is Deleted (6233521)

If you are deleting a cluster on a domain that uses the load balancer, the cluster's entry in the load balancer configuration file, `loadbalancer.xml`, is not removed. When the cluster is deleted, even though the cluster's entry remains in the `loadbalancer.xml` file, the entry will not be used.

Listing JDBC or JMS Resources for a Cluster or Server Instance Displays All the Resources in the Domain (6235760)

The output of the `listJMSResources` task for a specified target (a cluster or server instance name) displays the resources for the domain on which the cluster or instance is running.

Setting the True Flag for `createSamplesDomain` Does Not Create a Sample Domain (6239618)

When configuring the variable setting for `installHost` of Sun Java System App Server 8.1, setting `createSamplesDomain` to `true` does not actually create a sample domain.

N1 Service Provisioning System Does Not Use All the Variables Provided During Installation of Default Domain, `domain1` (6400633)

During Sun Java System App Server installation, `domain1` is created and there is no component created for the default domain, `domain1`. When the user wants to create `domain1` and enters different port numbers than the ones entered from the Sun Java System App Server Install components, the port numbers are not used since `domain1` has already been created. However, the DAS component is successfully installed but the variables used are from the Sun Java System App Server Install component.

Workaround:

1. Request the user to not use `domain1` when creating the domain in the DAS component.
2. Or, when creating `domain1` using the DAS component, use the same port numbers specified in the Sun Java System App Server Install component.

Applications Are Not Removed From the Target Host on Uninstalling the Application Server Completely (6369545)

Applications are not removed from the Sun N1 Service Provisioning System database when Cluster, Managed Server and Domain are removed.

Workaround:

Before deleting the Cluster, Managed Server or Domain, make sure the deployed Application components are removed from the Sun N1 Service Provisioning System database.

3

Installing and Configuring the Sun Java System App Server Plug-In

This chapter explains how to install and configure the Sun Java System App Server Plug-In. The chapter contains the following information:

- "Acquiring the Sun Java System Application Server Plug-In" on page 17
- "Importing the Sun Java System Application Server Plug-In to the Sun N1 Service Provisioning System" on page 19
- "Upgrading Considerations" on page 20

Acquiring the Sun Java System Application Server Plug-In

The Sun Java System App Server Plug-In is packaged as a *plug-in* to the Sun N1 Service Provisioning System software. Plug-ins are in Java™ Archive (JAR) files. The plug-in files for the Sun Java System Application Server Plug-In 3.0 are available from the Sun N1 Service Provisioning System 5.2 DVD or from the Sun Download Center.

Acquiring the Sun Java System App Server Plug-In is a two-step process. First, you must add the package file that contains the Sun N1 Service Provisioning System containers plug-in JAR file to your system. Then, you must import the Sun N1 Service Provisioning System containers plug-in JAR file to the Master Server.

Once the package file is added to your system, the Sun Java System Application Server Plug-In 3.0 is available for import from two different JAR files. Choose the correct file depending upon your situation.

- If you are importing the Sun Java System App Server Plug-In for the first time, acquire the `com.sun.sjsas81_3.0.jar` file.

- If you have already imported the previous version of the Sun Java System App Server Plug-In, acquire the `com.sun.sjsas81_2.0_3.0.jar` file.

1. Add the file containing the JAR file for Solaris, Linux, or Windows:

   - "Adding the Sun Java System Application Server Plug-In for Solaris" on page 18
   - "Adding the Sun Java System Application Server Plug-In for Linux" on page 18

- "Adding the Sun Java System Application Server Plug-In for Windows" on page 19
2. Import the JAR file into the Sun N1 Service Provisioning System. See "Importing the Sun Java System Application Server Plug-In to the Sun N1 Service Provisioning System" on page 19

Adding the Sun Java System Application Server Plug-In for Solaris

The Sun Java System App Server Plug-In for Solaris is contained in the SUNWspssas package.

▼ To Add the Sun Java System App Server Plug-In for Solaris

1   In a terminal window, become superuser.

2   Move to the directory containing the plug-in package.

3   Type the following command and press Return.

```
# pkgadd –d . SUNWspssas
```

The standalone JAR file is in the /opt/SUNWn1sps/plugins/com.sun.sjsas81/ directory. The upgrade JAR file is in the /opt/SUNWn1sps/plugins/com.sun.sjsas81/Upgrade directory.

Adding the Sun Java System Application Server Plug-In for Linux

The Sun Java System App Server Plug-In for Linux is contained in the sun-spssas-3.0-1.noarch.rpm file.

▼ To Add the Sun Java System App Server Plug-In for Linux

1   In a terminal window, become superuser.

2   Move to the directory containing the sun-spssas-3.0-1.noarch.rpm file.

3   Type the following command and press Return.

```
# rpm –i package_directory/sun-spssas-3.0-1.noarch
```

The standalone JAR file is in the /opt/sun/N1_Service_Provisioning_System/plugins/com.sun.sjsas81 directory. The upgrade JAR file is in the /opt/sun/N1_Service_Provisioning_System/plugins/com.sun.sjsas81/Upgrade directory.

Adding the Sun Java System Application Server Plug-In for Windows

The Sun Java System App Server Plug-In for Windows is contained in the `sun-spssas-3.0.msi` file.

▼ To Add the Sun Java System App Server Plug-In forWindows

1 **Move to the directory containing the** `sun-spssas-3.0.msi` **file.**

2 **Double-click the** `sun-spssas-3.0.msi` **file.**

The Installer GUI starts. The JAR file is copied to the `C:\Program Files\N1 Service Provisioning System\plugins\com.sun.sjsas81` directory. The upgrade JAR file is in the `C:\Program Files\N1 Service Provisioning System\plugins\com.sun.sjsas81\Upgrade` directory.

Importing the Sun Java System Application Server Plug-In to the Sun N1 Service Provisioning System

To make a given plug-in known to the Sun N1 Service Provisioning System, you need to import the plug-in to the Master Server. If you have already imported a previous version of the Sun Java System App Server Plug-In, you will need to upgrade to the new plug-in.

▼ To Import the Sun Java System App Server Plug-In Using the Browser Interface

To import or upgrade a plug-in, follow these steps as explained in detail in Chapter 5, "Plug-In Administration," in *Sun N1 Service Provisioning System 5.2 System Administration Guide*.

1 **In the Administrative section of the browser interface main window, click Plug-ins.**

2 **In the Action column of the Plug-ins page, click Import.**

3 **Browse to the location of the JAR file.**

- If you are importing the Sun Java System Application Server Plug-In 3.0 for the first time, select the `com.sun.sjsas81_3.0.jar` file.

- If you have already imported a previous version of the Sun Java System Application Server Plug-In 3.0, select the `com.sun.sjsas81_2.0_3.0.jar` file.

4 **Click the Continue to Import button.**

When the import completes successfully, a plug-in details page appears and shows you the objects that the plug-in provides.

▼ How to Import the Sun Java System Application Server Plug-In 3.0 Using the CLI

You can also import a plug-in by using the command line.

◗ **To import a plug-in file from the CLI, type:**

% `cr_cli -cmd plg.d.add -path` *plugin-filename* `-u` *username* `—p` *password*

- If you are importing the Sun Java System App Server Plug-In for the first time, *plugin-filename* is `com.sun.sjsas81_3.0.jar`.
- If you have already imported the previous version of the Sun Java System Application Server Plug-In 3.0, *plugin-filename* is `com.sun.sjsas81_2.0_3.0.jar`.

▼ How to Patch the Sun Java System App Server Plug-In

1 **Check the Support (**`http://www.sun.com/support/`**) site for available patches for the Sun Java System App Server Plug-In.**

2 **To apply the patch, follow the instructions in the patch** README **file.**

Upgrading Considerations

The following information is related to viewing previous versions of components after upgrading.

The following components might be affected:

- Clusters
- Domains
- JDBC Resources
- JMS Resources
- Managed Server Instances
- Threadpool Resources

▼ How to Access Previous Versions of a Component

When you upgrade to a new version of a plug-in, the Common Tasks page for the plug-in is updated to provide links to components that are installed with the new plug-in version. This feature enables

you to easily view components that contain the most current features and improvements. Components that rely on old functionality are not linked to from the Common Tasks page.

If you have trouble finding a component that you previously installed, the component was likely created from the previous version of the plug-in.

Follow these steps to view or work with a previous version of the component:

1   Click the appropriate component procedure on the Common Tasks page.

The Component Detail page appears.

2   On the Component Details page, click Version History.

A list of components and versions appears.

3   Click the appropriate link.

    a.   To perform a task with a component, click the version number or the details link that applies to the component that you want to use.

    b.   To view where a component is installed, click Where Installed.

4

Using the Sun Java System App Server Plug-In

The Sun Java System App Server Plug-In provides a number of specific component types and provides easy access to functions for working with Sun Java System App Server applications. This chapter describes the following information:

Installing and Using the Sun Java System App Server Environment with the Sun N1 Service Provisioning System

Use the Sun Java System App Server Common Tasks page as a starting point to create and manage Application Server instances and clusters.

Plug-In Conventions

The provisioning system allows you to provision and manage applications.

Plans and Component Procedures

The Sun Java System App Server plug-in provides both plans and component procedures as tools for you to perform tasks. By using plans, you link directly to the functionality you desire. Using component procedures, you have a greater number of tasks that you can perform.

Global Prerequisites

This section describes the requirements for using the Sun Java System App Server Plug-In.

The owner of the web server instance should be the same user that installs Sun Java System App Server. Otherwise, the load balancer files will not be installed or created properly.

Note – To configure topologies involving clusters spanning multiple machines, ensure that the user who owns the distributions (Master Server and Remote Agents) and the user who runs the Master Server and Remote Agents are the same.

Process Overview

Creating a Sun Java System App Server environment within the provisioning system is similar to the process of creating the environment without the provisioning system.

1. Import the Application Server plug-in.
2. Set the Application Server session variables.
3. Prepare all hosts.
4. Create an Application Server Domain Administration Server.
5. (Optional) Create an Application Server cluster.
6. Create server instances as a part of the cluster or domain.
7. (Optional) Create additional resources used by the applications.
8. (Optional) Configure the JMS server.
9. Capture application files (EARs and WARs).
10. Deploy applications to target instances or clusters.

Creating the Sun Java System App Server Domain Infrastructure

This section describes how to prepare your Sun N1 Service Provisioning System installation to take advantage of the features provided by Sun Java System App Server Plug-In.

The following are component procedures for Sun Java System App Server 8.1 Install Management:

`install: default`	Installs Sun Java System App Server on the target host and marks the component installed in the Sun N1 Service Provisioning System database.
`uninstall: default`	Same as `markOnly`; marks the component uninstalled in the Sun N1 Service Provisioning System database.
`uninstall: uninstallAppServer`	Uninstalls Sun Java System App Server from the target host and marks the component as uninstalled in the Sun N1 Service Provisioning System database.

▼ How to Install the Sun Java System App Server Software

1 **Select the AppServer 8.1 icon under the Common Tasks section of the left control panel.**

2 **Click the SJSAS 8.1 Install Management: Install or Uninstall link.**

3 **Click the** Run **action next to** install: default**.**

4 **In the Plan Parameters area, select the variable settings for the Install component that you plan to deploy.**

The Install component installs the Sun Java System App Server on a Remote Agent. After you have installed the Install component on a Remote Agent, you do not have to reinstall that component again.

When you run the plan to install a new domain or server instance, the provisioning system checks to see if an Install component exists in the installPath directory. If the Install component is already installed, the provisioning system does not reinstall the Install component.

- If the variable settings have been established for this component, select the appropriate settings from the menu.

- If the settings are not available from the menu, click Select From List.

  The Select Variable Settings From List window displays.

  - To create a new set of variable settings, select Create Set.

    The following list provides commonly-updated variables for the Install component.

variable set name	Required. A name for the new variable set you create.
installDirectory	Required. The location where the Sun Java System App Server software will be installed.
sourceImage	Required. The location of the Sun Java System App Server package installer.
	Typically this is an NFS-mounted directory containing the extracted Sun Java System App Server installer. For example (assuming the sourceImage binary was downloaded into the /net/mymachine/nfs directory), if the value of sourceImage is /net/mymachine/nfs/sjsappserver8.1, you would do the following before installing the Sun Java System App Server software:

    ```
    # mkdir /net/mymachine/nfs/sjsappserver8.1
    ```

```
# cd /net/mymachine/nfs/sjsappserver8.1
```

```
# unzip
../sjsas_ee-8_1_01_2005Q1-solaris-sparc.bin
```

Upon unzipping the binary, you will find a file called
setup. Use chmod 755 setup to turn on executable
permissions on the file. Executable permissions need to
be enabled as well to the other files in the
<sourceImage>/package/jre/bin/java directory.

installId	Required. A unique ID to associate domains, clusters, node agents, and Application Server instances with an Install component. If installing on multiple hosts, set installId to :[target:sys.hostName].
adminUserId	The user name of the administrator of the domain or cluster.
	Typically this will be picked up from the session variables.
adminPassword	The password for the administrator login of the domain or cluster.
	Typically this will be picked up from the session variables.
httpLoadBalancerPluginType	If you are using a web server load balancer with the domain or cluster, enter the type of web server here. Valid entries are Sun ONE Web Server or Apache HTTP Server.
	The web server must already be installed on the target host.
webserverInstallDirectory	Location of the web server, used when installing the load balancer plug-in. For Sun ONE Web Server, enter the instance directory. For Apache HTTP Server, enter the install directory.
webserverConfigDirectory	The directory where the load balancer configuration file, loadbalancer.xml will be located.
	Only set this variable if the load balancing plug-in is already installed on the web server gateway host.
adminPasswordEncryptedFlag	If set to true, the administrator's password will be encrypted. The default is false.

asadminPort	The port on which the Application Server domain administration server listens. The default is 4849. The port is used to create the default domain, domain1.
adminWebPort	The port on which deployed web applications run. The default is 8090. The port is used to create the default domain, domain1.
instanceHTTPSPort	The port on which secure HTTP (HTTPS) applications are run. The default is 1043. The port is used to create the default domain, domain1.
createSamplesDomain	If true, a domain containing Application Server sample applications will be installed on the instance. The default is false.
storeAdminAuth	If true, the administrator's password will be written to a file. The default is false.
tempStateFile	The location of temporary files used during installation. We recommend that you do not change this variable.
masterpassword	The password for the keystore certificate. The default value is changeit.
domainDirectory	The directory location where the default domain, domain1, will be created.

- To use variable components from another component, click Import Set.

  For more information about importing variable sets, see "How to Run a Plan" in *Sun N1 Service Provisioning System 5.2 Operation and Provisioning Guide*.

5 Select the target host.

Install the Application Server software on the Remote Agent.

6 Deselect the Target Host Set **checkbox.**

7 If you want to run a detailed check (preflight) before installing, select Perform Detailed Preflight**.**

8 Click the Run Plan (includes preflight) button.

▼ How To Create a Sun Java System App Server Domain Administration Server

The Domain Administration Server manages Application Server domains. It must be installed before you can create individual clusters and/or managed server instances. When you install a Domain Administration Server, you create an initial domain and an administration server that controls that domain.

Before You Begin To create an Application Server domain, you must belong to a user group that has the Run Component Procedures permission on the folder that contains the component. You must also have the Allow on Host Set permission for the host set on which you plan to deploy the component.

1   **From the Common Tasks section in the browser interface, click** AppServer 8.1**.**

2   **Under SJSAS 8.1 AppServer infrastructure→SJSAS 8.1 domains click** Manage**.**

3   **Under Component Procedures click the** Run **action next to** install: default**.**

4   **In the Plan Parameters area, select the variable settings for the Domain Administration Server you want to install.**

- If the variable settings have been established for this component, select the appropriate settings from the menu.

- If the settings are not available from the menu, click Select From List.

  The Select Variable Settings From List window displays.

  - To create a new set of variable settings, select Create Set.

    The following list provides commonly-updated variables for the Install component.

variable set name	Required. A name for the new variable set you create.
installIdentifier	A unique ID to associate a Domain Administration Server with an install image.
domainName	The name of the domain you wish to create.
adminUserId	The user name of the administrator of the domain or cluster.
adminPassword	The password for the administrator login of the domain or cluster.
adminHost	The name of the host on which the Domain Administration Server will be installed.
portRangePrefix	The HTTP port range prefix. For example, a value of 100 means the HTTP port range begin with 10000. The default value is 100.

httpListenerPort	The HTTP listener port, used with portRangePrefix. The valid range is 1–65553. For port numbers 1–1024, superuser permissions are required. The default is portRangePrefix plus 00. If portRangePrefix is set to 100, the HTTP listener port is 10000.
httpsListenerPort	The HTTPS listener port, used with portRangePrefix. The valid range is 1–65553. For port numbers 1–1024, superuser permissions are required. The default is portRangePrefix plus 05. If portRangePrefix is set to 100, the HTTPS listener port is 10005.
instancePort	The default port number for the Domain Administration Server.
iiopListenerPort	The Internet Inter-Orb Protrocol (IIOP) listener port, used with portRangePrefix. The valid range is 1–65553. For port numbers 1–1024, superuser permissions are required. The default is portRangePrefix plus 10. If portRangePrefix is set to 100, the IIOP listener port is 10010.
iiopsslListenerPort	The SSL-enabled Internet Inter-Orb Protrocol (IIOP) listener port, used with portRangePrefix. The valid range is 1–65553. For port numbers 1–1024, superuser permissions are required. The default is portRangePrefix plus 15. If portRangePrefix is set to 100, the secure IIOP listener port is 10015.
iiopsslmutualauthPort	The SSL-enabled Internet Inter-Orb Protrocol (IIOP) listener port for mutual authorization, used with portRangePrefix. The valid range is 1–65553. For port numbers 1–1024, superuser permissions are required. The default is portRangePrefix plus 20. If portRangePrefix is set to 100, the mutual authorization secure IIOP listener port is 10020.
jmsConnectorPort	The port on which the Java Messaging Service (JMS) connector listens. The valid range is 1–65553. For port numbers 1–1024, superuser permissions are required. The default is portRangePrefix plus 25. If portRangePrefix is set to 100, the JMS connector port is 10025.
installDirectory	The location where the Application Server is installed.
masterpassword	The password used for the keystore certificate. The default value is changeit.

- To use variable components from another component, click Import Set.

  For more information about importing variable sets, see "How to Run a Plan" in *Sun N1 Service Provisioning System 5.2 Operation and Provisioning Guide*.

5   Select the target host.

Install the Domain Administration Server and the domain on the Remote Agent.

6   Deselect the `Target Host Set` **checkbox.**

7   If you want to run a detailed check (preflight) before installing, select `Perform Detailed Preflight`**.**

8   Click the Run Plan (includes preflight) button.

▼ How To Create a Sun Java System App Server Instance

You can run multiple Application Server instances in each domain.

Before You Begin   Before you create an Application Server instance, you must do the following:

- Create the domain on which the server instance will run.

  For information on creating a domain, see "Creating the Sun Java System App Server Domain Infrastructure" on page 24

- The Domain Administration Server must be running.

  To manually start a Domain Administration Server, see "How to Start a Domain Administration Server" on page 37

- You must belong to a user group that has the Run Component Procedures permission on the folder that contains the component. You must also have the Allow on Host Set permission for the host set on which you plan to deploy the component.

1   From the Common Tasks section in the browser interface, click AppServer 8.1.

2   Under SJSAS 8.1 AppServer infrastructure→SJSAS 8.1 Managed Server Instances click `Manage`**.**

3   Under Component Procedures click the `Run` **action next to** `install: default`**.**

4   In the Plan Parameters area, select the variable settings for the managed server instance you want to install.

- If the variable settings have been established for this component, select the appropriate settings from the menu.

- If the settings are not available from the menu, click `Select From List`.

  The Select Variable Settings From List window displays.

  - To create a new set of variable settings, select `Create Set`.

    The following list provides commonly-updated variables for the `Install` component.

variable set name	Required. A name for the new variable set you create.
installIdentifier	A unique ID to associate an Application Server instance with an Install Component.
domainName	The domain name in which this server instance will be located
targetName	The name of the server.
portRangePrefix	The HTTP port range prefix. For example, a value of 100 means the HTTP port range begin with 10000. The default value is 100.
httpListenerPort	The HTTP listener port, used with portRangePrefix. The valid range is 1–65553. For port numbers 1–1024, superuser permissions are required. The default is portRangePrefix plus 00. If portRangePrefix is set to 100, the HTTP listener port is 10000.
httpsListenerPort	The HTTPS listener port, used with portRangePrefix. The valid range is 1–65553. For port numbers 1–1024, superuser permissions are required. The default is portRangePrefix plus 05. If portRangePrefix is set to 100, the HTTPS listener port is 10005.
iiopListenerPort	The Internet Inter-Orb Protocol (IIOP) listener port, used with portRangePrefix. The valid range is 1–65553. For port numbers 1–1024, superuser permissions are required. The default is portRangePrefix plus 10. If portRangePrefix is set to 100, the IIOP listener port is 10010.
iiopsslListenerPort	The SSL-enabled Internet Inter-Orb Protocol (IIOP) listener port, used with portRangePrefix. The valid range is 1–65553. For port numbers 1–1024, superuser permissions are required. The default is portRangePrefix plus 15. If portRangePrefix is set to 100, the secure IIOP listener port is 10015.
iiopsslmutualauthPort	The SSL-enabled Internet Inter-Orb Protocol (IIOP) listener port for mutual authorization, used with portRangePrefix. The valid range is 1–65553. For port numbers 1–1024, superuser permissions are required. The default is portRangePrefix plus 20. If portRangePrefix is set to 100, the mutual authorization secure IIOP listener port is 10020.
jmsConnectorPort	This variable is not in use.
clusterName	The name of the cluster to which this server instance belongs. If no cluster is specified the server instance is a standalone server. By default no cluster is specified.

- To use variable components from another component, click Import Set.

For more information about importing variable sets, see "How to Run a Plan" in *Sun N1 Service Provisioning System 5.2 Operation and Provisioning Guide*.

5   Select the target host.

Install the server instance on the Remote Agent.

6   Deselect the `Target Host Set` **checkbox.**

7   If you want to run a detailed check (preflight) before installing, select `Perform Detailed Preflight`**.**

8   Click the Run Plan (includes preflight) button.

▼ How To Create a Sun Java System App Server Cluster

A cluster is a collection of zero or more server instances with the following properties:

- All instances in the cluster reference the same configuration.
- All instances in the cluster have the same set of deployed applications (for example, J2EE application EAR, web module WAR, or ejb jar file).
- All instances in the cluster have the same set off resources, resulting in the same JNDI namespace.

The server instances that constitute a cluster can run on the same machine, or can be located on different machines. Each cluster member in a cluster must run the same version of Application Server.

A cluster provides:

- Increase application availability with the server's failover capabilities.
- Increase throughput by scaling applications across multiple servers.

Before You Begin   Before you create a cluster, you must create a Domain Administration Server.

To create a cluster, you must belong to a user group that has the Run Component Procedures permission on the folder that contains the component. You must also have the Allow on Host Set permission for the host set on which you plan to deploy the component.

1   From the Common Tasks section in the browser interface, click AppServer 8.1.

2   Under SJSAS 8.1 AppServer infrastructure→SJSAS 8.1 clusters click `Manage`**.**

3   Under Component Procedures click the `Run` **action next to** `install: default`**.**

4   In the Plan Parameters area, select the variable settings for the cluster you want to install.

- If the variable settings have been established for this component, select the appropriate settings from the menu.
- If the settings are not available from the menu, click Select From List.

  The Select Variable Settings From List window displays.

  - To create a new set of variable settings, select Create Set.

    The following list provides commonly-updated variables for the Install component.

variable set name	Required. A name for the new variable set you create.
installIdentifier	A unique ID to associate a Cluster with an Install Component.
domainName	The domain name in which this cluster will be located
targetName	The name of the cluster.
portRangePrefix	The HTTP port range prefix. For example, a value of 100 means the HTTP port range begin with 10000. The default value is 100.
httpListenerPort	The HTTP listener port, used with portRangePrefix. The valid range is 1–65553. For port numbers 1–1024, superuser permissions are required. The default is portRangePrefix plus 00. If portRangePrefix is set to 100, the HTTP listener port is 10000.
httpsListenerPort	The HTTPS listener port, used with portRangePrefix. The valid range is 1–65553. For port numbers 1–1024, superuser permissions are required. The default is portRangePrefix plus 05. If portRangePrefix is set to 100, the HTTPS listener port is 10005.
iiopListenerPort	The Internet Inter-Orb Protocol (IIOP) listener port, used with portRangePrefix. The valid range is 1–65553. For port numbers 1–1024, superuser permissions are required. The default is portRangePrefix plus 10. If portRangePrefix is set to 100, the IIOP listener port is 10010.
iiopsslListenerPort	The SSL-enabled Internet Inter-Orb Protocol (IIOP) listener port, used with portRangePrefix. The valid range is 1–65553. For port numbers 1–1024, superuser permissions are required. The default is portRangePrefix plus 15. If portRangePrefix is set to 100, the secure IIOP listener port is 10015.
iiopsslmutualauthPort	The SSL-enabled Internet Inter-Orb Protocol (IIOP) listener port for mutual authorization, used with portRangePrefix. The valid range is 1–65553. For port numbers 1–1024, superuser permissions are required. The default is portRangePrefix plus 20. If portRangePrefix is set to 100, the mutual authorization secure IIOP listener port is 10020.

jmsConnectorPort	This variable is not in use.	
httpLoadBalancer	If true, applications deployed to this cluster will be load balanced. The default is false.	
httpsLoadBalancer	If true, applications responding to HTTPS requests will be load balanced. This variable has no effect if httpLoadBalancer is false. The default is false.	
routeCookie	If true, the load balancer will route cookies. This variable has no effect if httpLoadBalancer is false. The default is false.	

- To use variable components from another component, click Import Set.

  For more information about importing variable sets, see "How to Run a Plan" in *Sun N1 Service Provisioning System 5.2 Operation and Provisioning Guide.*

5   **Select the target host.**

Install the cluster on the Remote Agent.

6   **Deselect the** Target Host Set **checkbox.**

7   **If you want to run a detailed check (preflight) before installing, select** Perform Detailed Preflight.

8   **Click the Run Plan (includes preflight) button.**

▼ How to Uninstall the Sun Java System App Server Software

1   **Click the AppServer 8.1 link under the Common Tasks section of the left control panel.**

2   **Click the SJSAS 8.1 Install Management: Install or Uninstall link.**

3   **Click the Run action next to uninstall: uninstallAppServer.**

Managing Sun Java System App Server Instances and Clusters

This section describes how to maintain Sun Java System App Server managed server instances and clusters.

Managing Sun Java System App Server Domains and Domain Administration Servers

The following are component procedures for Sun Java System App Server 8.1 Domain Management:

install: default	Creates a domain on the target host and marks the component installed in the Sun N1 Service Provisioning System database.
uninstall: default	Same as markOnly; marks the component and also the dependant components as uninstalled in the Sun N1 Service Provisioning System database.
uninstall: deleteDomain	Deletes the domain from the target host and marks the component as uninstalled in the Sun N1 Service Provisioning System database.
startCompleteDomain	Starts the domain and the dependant components.
stopCompleteDomain	Stops the domain and the dependant components.
verifyDomainServerRunning	Verifies that the domain is running.
startDomainServer	Starts the domain admin server.
stopDomainServer	Stops the domain admin server.
listDomains	Lists all domains.

▼ How to Remove a Sun Java System App Server Domain and Domain Administration Server

By removing an Application Server domain through the provisioning system, you automatically perform the following tasks:

- Uninstall the Domain Administration Server.
- Uninstall all Managed Servers components in the domain.
- Uninstall all cluster components in the domain.
- Remove application components and applications.

Before You Begin   To remove an Application Server domain, you must belong to a user group that has the Run Component Procedures permission on the folder that contains the component. You must also have the Allow on Host Set permission for the host set on which you plan to run the component.

1. **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2. **Under SJSAS 8.1 AppServer infrastructure→SJSAS 8.1 domains click** Manage**.**

3. **Under Component Procedures click the** Run **action next to** uninstall: default**.**

4 **Under Current Installations, select the domains that you would like to remove.**

5 **Click the Run Selected Installations button.**

6 **Click the Run Plan (Includes Preflight) button.**

▼ How to Start a Complete Domain

Starting a complete domain starts the domain's Domain Administration Server if it is not currently running, all clusters in the domain, and all managed server instances in the domain.

Before You Begin To start an Application Server domain, you must belong to a user group that has the Run Component Procedures permission on the folder that contains the component. You must also have the Allow on Host Set permission for the host set on which you plan to run the component.

1 **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2 **Under SJSAS 8.1 AppServer infrastructure→SJSAS 8.1 domains, click** Manage**.**

3 **Under Component Procedures, click the** Run **action next to** startCompleteDomain**.**

4 **Under Current Installations, select the domain that you would like to start.**

5 **Click the Run Selected Installations button.**

6 **Click the Run Plan (Includes Preflight) button.**

▼ How to Stop a Complete Domain

Stopping a completed domain stops all running clusters and all running managed server instances for a particular domain.

Before You Begin To stop an Application Server domain, you must belong to a user group that has the Run Component Procedures permission on the folder that contains the component. You must also have the Allow on Host Set permission for the host set on which you plan to run the component.

1 **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2 **Under SJSAS 8.1 AppServer infrastructure→SJSAS 8.1 domains click** Manage**.**

3 **Under Component Procedures click the** Run **action next to** stopCompleteDomain**.**

4 **Under Current Installations, select the domain that you would like to stop.**

5   Click the Run Selected Installations button.

6   Click the Run Plan (Includes Preflight) button.

▼ How to Start a Domain Administration Server

Before You Begin   To start an Application Server Domain Administration Server, you must belong to a user group that has the Run Component Procedures permission on the folder that contains the component. You must also have the Allow on Host Set permission for the host set on which you plan to run the component.

1   From the Common Tasks section in the browser interface, click AppServer 8.1.

2   Under SJSAS 8.1 AppServer infrastructure→SJSAS 8.1 domains click Manage**.**

3   Under Component Procedures click the Run **action next to** startDomainServer**.**

4   Under Current Installations, select the domains that you would like to start.

5   Click the Run Selected Installations button.

6   Click the Run Plan (Includes Preflight) button.

▼ How to Stop a Domain Administration Server

Before You Begin   To stop an Application Server Domain Administration Server, you must belong to a user group that has the Run Component Procedures permission on the folder that contains the component. You must also have the Allow on Host Set permission for the host set on which you plan to run the component.

1   From the Common Tasks section in the browser interface, click AppServer 8.1.

2   Under SJSAS 8.1 AppServer infrastructure→SJSAS 8.1 domains click Manage**.**

3   Under Component Procedures click the Run **action next to** stopDomainServer**.**

4   Under Current Installations, select the domains that you would like to stop.

5   Click the Run Selected Installations button.

6   Click the Run Plan (Includes Preflight) button.

▼ How to Verify a Domain Administration Server is Running

You can verify that a Domain Administration Server is running on a particular installation.

Before You Begin To verify a Domain Administration Server is running, you must belong to a user group that has the Run Component Procedures permission on the folder that contains the component. You must also have the Allow on Host Set permission for the host set on which you plan to run the component.

1 **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2 **Under SJSAS 8.1 AppServer infrastructure→SJSAS 8.1 domains click** Manage**.**

3 **Under Component Procedures, click the** Run **action next to** verifyDomainServerRunning**.**

4 **Under Current Installations, select the domains that you would like to verify.**

5 **Click the Run Selected Installations button.**

6 **Click the Run Plan (Includes Preflight) button.**

Managing Sun Java System App Server Clusters

The following are component procedures for Sun Java System App Server 8.1 Cluster Management:

install: default	Creates a cluster on the target host and marks the component as installed in the Sun N1 Service Provisioning System database.
install: markOnly	Marks the component as installed in the Sun N1 Service Provisioning System database. This procedure does not create any dependency on other components.
uninstall: default	Similar to markOnly; marks the component and also dependant components as uninstalled in the Sun N1 Service Provisioning System database.
uninstall: markOnly	Marks the component as uninstalled in the Sun N1 Service Provisioning System database. This procedure does not mark the dependant components as uninstalled in the Sun N1 Service Provisioning System database.
uninstall: delete	Deletes the cluster from the target host. Marks the component as uninstalled, removes the component, and removes dependant components in the Sun N1 Service Provisioning System database.
start	Starts the cluster. The cluster must contain one or more Managed Server components.
stop	Stops the cluster.
verifyClusterRunning	Verifies that the cluster is running.

▼ How to Remove a Sun Java System App Server Cluster

When you remove a cluster, all managed server instances associated with that cluster are also removed.

Before You Begin   To remove a cluster, you must belong to a user group that has the Run Component Procedures permission on the folder that contains the component. You must also have the Allow on Host Set permission for the host set on which you plan to deploy the component.

1   **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2   **Under SJSAS 8.1 AppServer infrastructure→SJSAS 8.1 clusters click** Manage**.**

3   **Under Component Procedures, click the** Run **action next to** uninstall: default**.**

4   **Under Current Installations, select the clusters that you would like to remove.**

5   **Click the Run Selected Installations button.**

6   **Click the Run Plan (Includes Preflight) button.**

▼ How to Start a Sun Java System App Server Cluster

Before You Begin   To start a cluster, you must belong to a user group that has the Run Component Procedures permission on the folder that contains the component. You must also have the Allow on Host Set permission for the host set on which you plan to deploy the component.

A cluster is not a running instance, so when you start the cluster, you are actually starting the managed server instances associated with the cluster.

1   **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2   **Under SJSAS 8.1 AppServer infrastructure→SJSAS 8.1 clusters click** Manage**.**

3   **Under Component Procedures, click the** Run **action next to** start**.**

4   **Under Current Installations, select the clusters that you would like to start.**

5   **Click the Run Selected Installations button.**

6   **Click the Run Plan (Includes Preflight) button.**

▼ How to Stop a Sun Java System App Server Cluster

Before You Begin   To stop a cluster, you must belong to a user group that has the Run Component Procedures permission on the folder that contains the component. You must also have the Allow on Host Set permission for the host set on which you plan to deploy the component.

1   From the Common Tasks section in the browser interface, click AppServer 8.1.

2   Under SJSAS 8.1 AppServer infrastructure→SJSAS 8.1 clusters click Manage.

3   Under Component Procedures, click the Run action next to stop.

4   Under Current Installations, select the clusters that you would like to stop.

5   Click the Run Selected Installations button.

6   Click the Run Plan (Includes Preflight) button.

▼ How to Verify a Sun Java System App Server Cluster is Running

Before You Begin   To verify a cluster, you must belong to a user group that has the Run Component Procedures permission on the folder that contains the component. You must also have the Allow on Host Set permission for the host set on which you plan to deploy the component.

1   From the Common Tasks section in the browser interface, click AppServer 8.1.

2   Under SJSAS 8.1 AppServer infrastructure→SJSAS 8.1 clusters click Manage.

3   Under Component Procedures, click the Run action next to verifyClusterRunning.

4   Under Current Installations, select the clusters that you would like to verify.

5   Click the Run Selected Installations button.

6   Click the Run Plan (Includes Preflight) button.

Managing Sun Java System App Server Instances

The following are component procedures for Sun Java System App Server 8.1 Managed Server Instance Management:

install: default   Creates a server instance on the target host, marks this component as installed in the Sun N1 Service Provisioning System database and creates the dependency.

`install: markOnly`	Marks the component as installed in the Sun N1 Service Provisioning System database and does not create any dependency.
`uninstall: default`	Marks the component and dependent components as uninstalled in the Sun N1 Service Provisioning System database.
`uninstall: markOnly`	Marks the component as uninstalled in the Sun N1 Service Provisioning System database. This procedure does not mark the dependant components as uninstalled in the Sun N1 Service Provisioning System database.
`uninstall: delete`	Deletes the server instance from the target host. Marks the component as uninstalled, removes the component, and removes dependant components in the Sun N1 Service Provisioning System database.
`start`	Starts the server instances.
`stop`	Stops the server instances.
`verifyServerRunning`	Verifies that the server is running.

▼ How to Remove a Sun Java System App Server Instance

Before You Begin   To remove a server instance you must belong to a user group that has the Run Component Procedures permission on the folder that contains the component. You must also have the Allow on Host Set permission for the host set on which you plan to deploy the component.

1. **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2. **Under SJSAS 8.1 AppServer infrastructure→SJSAS 8.1 Managed Server Instances click** Manage**.**

3. **Under Component Procedures, click the** Run **action next to** uninstall: default**.**

4. **Under Current Installations, select the server instances that you would like to remove.**

5. **Click the Run Selected Installations button.**

6. **Click the Run Plan (Includes Preflight) button.**

▼ How to Start a Sun Java System App Server Instance

Before You Begin   To start a server instance, you must belong to a user group that has the Run Component Procedures permission on the folder that contains the component. You must also have the Allow on Host Set permission for the host set on which you plan to deploy the component.

1. **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2. **Under SJSAS 8.1 AppServer infrastructure→SJSAS 8.1 Managed Server Instances click** Manage**.**

3   Under Component Procedures, click the Run **action next to** start**.**

4   Under Current Installations, select the server instances that you would like to start.

5   Click the Run Selected Installations button.

6   Click the Run Plan (Includes Preflight) button.

▼ How to Stop a Sun Java System App Server Instance

Before You Begin   To stop a server instance you must belong to a user group that has the Run Component Procedures permission on the folder that contains the component. You must also have the Allow on Host Set permission for the host set on which you plan to deploy the component.

1   From the Common Tasks section in the browser interface, click AppServer 8.1.

2   Under SJSAS 8.1 AppServer infrastructure→SJSAS 8.1 Managed Server Instances click Manage**.**

3   Under Component Procedures, click the Run **action next to** stop**.**

4   Under Current Installations, select the server instances that you would like to stop.

5   Click the Run Selected Installations button.

6   Click the Run Plan (Includes Preflight) button.

▼ How to Verify a Sun Java System App Server Instance is Running

Before You Begin   To verify a server instance is running you must belong to a user group that has the Run Component Procedures permission on the folder that contains the component. You must also have the Allow on Host Set permission for the host set on which you plan to deploy the component.

1   From the Common Tasks section in the browser interface, click AppServer 8.1.

2   Under SJSAS 8.1 AppServer infrastructure→SJSAS 8.1 Managed Server Instances click Manage**.**

3   Under Component Procedures, click the Run **action next to** verifyServerRunning**.**

4   Under Current Installations, select the server instance that you would like to verify.

5   Click the Run Selected Installations button.

6   Click the Run Plan (Includes Preflight) button.

Capturing and Managing Sun Java System App Server Applications

This section describes how to manage Sun Java System App Server applications.

The Sun Java System App Server Plug-In supports two types of application deployments:

- Enterprise Applications (EARs)
- Web Applications (WARs)

Managing the Application is a two step process involving:

1. Capturing the Application in to the Sun N1 Service Provisioning System Master Server.

2. Installing and Uninstalling the Application.

▼ How to Capture Sun Java System App Server Enterprise Applications

1 From the Common Tasks section in the browser interface, click AppServer 8.1.

2 Under SJSAS Application Tasks→Enterprise Applications (EARs) click `Create New`**.**

3 Enter a component directory for this component. The `com/sun/sjsas81` **namespace is not valid.**

4 Enter a label for this component.

5 Enter a description for this component.

6 Enter the host where the EAR file is located.
To select the host from a list, click `Select From List`.

7 Navigate to the location of the EAR.

    a. If you know the path to the EAR, enter it in `com.sun.sjsas81#EntappCT path`**.**

    b. To browse the selected host, select the directory and click `Open Highlighted Item`**.**

8 Select the EAR and click `Open Highlighted Item`**.**

9 Click `Check In Selected Item`**.**

10 Confirm the information on the check-in page, then click `Continue to Check-in`**.**

▼ How to Capture Sun Java System App Server Web Applications

1 **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2 **Under SJSAS Application Tasks→Web Applications (WARs) click** `Create New`**.**

3 **Enter a component directory for this component. The** `com/sun/sjsas81` **namespace is not valid.**

4 **Enter a label for this component.**

5 **Enter a description for this component.**

6 **Enter the host where the WAR file is located.**
To select the host from a list, click `Select From List`.

7 **Navigate to the location of the WAR.**

    a. **If you know the path to the WAR, enter it in** `com.sun.sjsas81#WebappCT` **path.**

    b. **To browse the selected host, select the directory and click** `Open Highlighted Item`**.**

8 **Select the WAR and click** `Open Highlighted Item`**.**

9 **Click** `Check In Selected Item`**.**

10 **Confirm the information on the check-in page, then click** `Continue to Check-in`**.**

Managing Sun Java System App Server Enterprise Applications

The following are component procedures for the Sun Java System App Server EAR Management:

`install: default`	Deploys the EAR to the target host and marks the component installed in the Sun N1 Service Provisioning System database.
`install: markOnly`	Marks the component installed in the Sun N1 Service Provisioning System database.
`uninstall: default`	Undeploys the EAR from the target host and marks the component as uninstalled in the Sun N1 Service Provisioning System database.

uninstall: markOnly	Marks the component as uninstalled in the Sun N1 Service Provisioning System database.

▼ How to Install a Sun Java System App Server Enterprise Application

1 From the Common Tasks section in the browser interface, click AppServer 8.1.

2 Under SJSAS Application Tasks→Enterprise Applications (EARs) click View All**.**

3 Click the name of the EAR you want to install.

4 Under Component Procedures click the Run **action next to** install: default**.**

5 In the Plan Parameters area, select the variable settings for the EAR you want to install.

- If the variable settings have been established for this component, select the appropriate settings from the menu.
- If the settings are not available from the menu, click Select From List.

  The Select Variable Settings From List window displays.

  - To create a new set of variable settings, select Create Set.

    The following list provides commonly-updated variables for the Install component.

variable set name	Required. A name for the new variable set you create.
installIdentifier	A unique ID to associate the EAR with an Install component.
domainName	The domain name in which this EAR will be installed
targetName	The cluster name or server instance where the EAR will be deployed.

  - To use variable components from another component, click Import Set.

    For more information about importing variable sets, see "How to Run a Plan" in *Sun N1 Service Provisioning System 5.2 Operation and Provisioning Guide*.

6 Select the target host.

7 Deselect the Target Host Set **checkbox.**

8 If you want to run a detailed check (preflight) before installing, select Perform Detailed Preflight**.**

9 Click the Run Plan (includes preflight) button.

▼ **How to Uninstall a Sun Java System App Server Enterprise Application**

1 **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2 **Under SJSAS Application Tasks→Enterprise Applications (EARs) click** `View All`**.**

3 **Click the name of the EAR you want to uninstall.**

4 **Under Component Procedures click the** `Run` **action next to** `uninstall: default`**.**

5 **Select the installation from which you want the EAR removed.**

6 **Click the Run Selected Installations button.**

7 **Click the Run Plan (includes preflight) button.**

Managing Sun Java System App Server Web Applications

The following are component procedures for the Sun Java System App Server WAR Management:

`install: default`	Deploys the WAR to the target host and marks the component installed in the Sun N1 Service Provisioning System database.
`install: markOnly`	Marks the component installed in the Sun N1 Service Provisioning System database.
`uninstall: default`	Undeploys the WAR from the target host and marks the component as uninstalled in the Sun N1 Service Provisioning System database.
`uninstall: markOnly`	Marks the component as uninstalled in the Sun N1 Service Provisioning System database.

▼ **How to Install a Sun Java System App Server Web Application**

1 **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2 **Under SJSAS Application Tasks→Web Applications (WARs) click** `View All`**.**

3 **Click the name of the WAR you want to install.**

4 **Under Component Procedures click the** `Run` **action next to** `install: default`**.**

5 **In the Plan Parameters area, select the variable settings for the WAR you want to install.**

- If the variable settings have been established for this component, select the appropriate settings from the menu.

- If the settings are not available from the menu, click Select From List.

  The Select Variable Settings From List window displays.

  - To create a new set of variable settings, select Create Set.

    The following list provides commonly-updated variables for the Install component.

    variable set name   Required. A name for the new variable set you create.

    installIdentifier   A unique ID to associate the WAR with an Install component.

    domainName   The domain name in which this WAR will be installed

    targetName   The cluster name or server instance where the WAR will be deployed.

  - To use variable components from another component, click Import Set.

    For more information about importing variable sets, see "How to Run a Plan" in *Sun N1 Service Provisioning System 5.2 Operation and Provisioning Guide*.

6 **Select the target host.**

7 **Deselect the** Target Host Set **checkbox.**

8 **Under Plan Variables set the context root of the web application.**

9 **If you want to run a detailed check (preflight) before installing, select** Perform Detailed Preflight**.**

10 **Click the Run Plan (includes preflight) button.**

▼ How to Uninstall a Sun Java System App Server Web Application

1 **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2 **Under SJSAS Application Tasks→Web Applications (WARs) click** View All**.**

3 **Click the name of the WAR you want to uninstall.**

4 **Under Component Procedures click the** Run **action next to** uninstall: default**.**

5 **Select the installation from which you want the WAR removed.**

6 **Click the Run Selected Installations button.**

7 **Click the Run Plan (includes preflight) button.**

Managing Sun Java System App Server Resources

This section describes how to manage Sun Java System App Server resources: JMS resources, JDBC resources, and thread pools.

Managing Java Messaging Service (JMS) Resources

The following are component procedures for Sun Java System App Server 8.1 JMS Resource Management:

`install: default`	Marks the component as installed in the Sun N1 Service Provisioning System database.
`uninstall: default`	Marks the component as uninstalled in the Sun N1 Service Provisioning System database.
`createJMSResource`	Creates a JMS Resource on the target host.
`deleteJMSResource`	Deletes a JMS Resource from the target host.
`listJMSResource`	Lists JMS Resources.

▼ How to Install a JMS Resource on a Host

1 **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2 **Under SJSAS AppServer Resources→SJSAS 8.1 JMS Resource click** `Manage JMS Resources`**.**

3 **Under Component Procedures click the** Run **action next to** `install: default`**.**

4 **In the Plan Parameters area, select the variable settings for the JMS Resource you want to install.**

- If the variable settings have been established for this component, select the appropriate settings from the menu.
- If the settings are not available from the menu, click `Select From List`.

   The Select Variable Settings From List window displays.

   - To create a new set of variable settings, select `Create Set`.

      The following list provides commonly-updated variables for the `Install` component.

variable set name	Required. A name for the new variable set you create.
`installIdentifier`	A unique ID to associate the JMS Resource with an `Install` component.
`domainName`	The domain name in which this JMS Resource will be located

- To use variable components from another component, click Import Set.

  For more information about importing variable sets, see "How to Run a Plan" in *Sun N1 Service Provisioning System 5.2 Operation and Provisioning Guide*.

5 **Select the target host.**

6 **Deselect the** `Target Host Set` **checkbox.**

7 **If you want to run a detailed check (preflight) before installing, select** `Perform Detailed Preflight`**.**

8 **Click the Run Plan (includes preflight) button.**

▼ How to Create a JMS Resource

1 **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2 **Under SJSAS AppServer Resources→SJSAS 8.1 JMS Resource click** `Manage JMS Resources`**.**

3 **Under Component Procedures click the** `Run` **action next to** `createJMSResource`**.**

4 **Select the installation on which you want to create the JMS resource.**

5 **Under Plan Variables:**

   a. Select the resource type. It can be one of:

   - `Topic`
   - `Queue`
   - `ConnectionFactory`
   - `TopicConnectionFactory`
   - `QueueConnectionFactory`

   b. Enter the JNDI name for the resource.

   c. Enter the target cluster or server name where the resource will be created.

   d. (Optional) Enter a description of the resource.

   e. Enter any JMS properties, separated by colons (`:`).

   Valid property names are:

   - `ClientId`
   - `AddressList`

- MessageServiceAddressList
- UserName
- Password
- ReconnectEnabled
- ReconnectAttempts
- ReconnectInterval
- AddressListBehavior
- AddressListIterations

6   **Click Run Plan (includes preflight) button.**

▼ How to Delete a JMS Resource

1   **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2   **Under SJSAS AppServer Resources→SJSAS 8.1 JMS Resource click** Manage JMS Resources**.**

3   **Under Component Procedures click the** Run **action next to** deleteJMSResource**.**

4   **Select the installation on which you want to delete the JMS resource.**

5   **Under Plan Variables:**

   a.   **Enter the JNDI name of the resource you want to delete.**

   b.   **Enter the target name of the resource you want to delete.**

6   **Click the Run Plan (includes preflight) button.**

Managing Java Database Connectivity (JDBC) Resources

The following are component procedures for Sun Java System App Server 8.1 JDBC Resource and Connection Pool Management:

install: default	Marks the component as installed in the Sun N1 Service Provisioning System database.
uninstall: default	Marks the component as uninstalled in the Sun N1 Service Provisioning System database.
createJDBCResource	Creates a JDBC Resource on the target host.
deleteJDBCResource	Deletes a JDBC Resource from the target host.

`listJDBCResources`	Lists JDBC Resources.
`createJDBCConnectionPool`	Creates a JDBC connection pool on the target host.
`deleteJDBCConnectionPool`	Deletes a JDBC connection pool from the target host.
`listJDBCConnectionPools`	Lists JDBC connection pools.

▼ How to Install a JDBC Resource on a Host

1 **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2 **Under SJSAS AppServer Resources→SJSAS 8.1 JDBC Resource click** `Manage JDBC Resources`**.**

3 **Under Component Procedures click the** `Run` **action next to** `install: default`**.**

4 **In the Plan Parameters area, select the variable settings for the JDBC Resource you want to install.**

- If the variable settings have been established for this component, select the appropriate settings from the menu.

- If the settings are not available from the menu, click `Select From List`.

  The Select Variable Settings From List window displays.

  - To create a new set of variable settings, select `Create Set`.

    The following list provides commonly-updated variables for the `Install` component.

variable set name	Required. A name for the new variable set you create.
`installIdentifier`	A unique ID to associate the JDBC Resource with an `Install` component.
`domainName`	The domain name in which this JDBC Resource will be located

  - To use variable components from another component, click `Import Set`.

    For more information about importing variable sets, see "How to Run a Plan" in *Sun N1 Service Provisioning System 5.2 Operation and Provisioning Guide*.

5 **Select the target host.**

6 **Deselect the** `Target Host Set` **checkbox.**

7 **If you want to run a detailed check (preflight) before installing, select** `Perform Detailed Preflight`**.**

8 **Click the Run Plan (includes preflight) button.**

▼ How to Create a JDBC Resource

Before You Begin   A JDBC Connection Pool must be created before creating a JDBC Resource.

1   **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2   **Under SJSAS AppServer Resources→SJSAS 8.1 JDBC Resource click** `Manage JDBC Resources`**.**

3   **Under Component Procedures click the** `Run` **action next to** `createJDBCResource`**.**

4   **Select the installation on which you want to create the JDBC resource.**

5   **Under Plan Variables:**

   a.   **Set the name of the Connection Pool.**

   b.   **Enter the JNDI name for the resource.**

   c.   **Enter the target cluster or server name where the resource will be created.**

   d.   **(Optional) Enter a description of the resource.**

   e.   **Enter any JDBC properties, separated by colons (`:`).**

6   **Click the Run Plan (includes preflight) button.**

▼ How to Delete a JDBC Resource

1   **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2   **Under SJSAS AppServer Resources→SJSAS 8.1 JDBC Resource click** `Manage JDBC Resources`**.**

3   **Under Component Procedures click the** `Run` **action next to** `deleteJDBCResource`**.**

4   **Select the installation on which you want to delete the JDBC resource.**

5   **Under Plan Variables:**

   a.   **Enter the JNDI name of the resource you want to delete.**

   b.   **Enter the target cluster or server instance name of the resource you want to delete.**

6   **Click the Run Plan (includes preflight) button.**

▼ How to List JDBC Resources

1 **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2 **Under SJSAS AppServer Resources→SJSAS 8.1 JDBC Resource click** `Manage JDBC Resources.`

3 **Under Component Procedures click the** `Run` **action next to** `listJDBCResources.`

4 **Select the installation on which you want to list the JDBC resources.**

5 **Under Plan Variables enter the target cluster or server instance name on which you want to list the JDBC Resources.**

6 **Click the Run Plan (includes preflight) button.**

▼ How to Create a JDBC Connection Pool

1 **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2 **Under SJSAS AppServer Resources→SJSAS 8.1 JDBC Resource click** `Manage JDBC Resources.`

3 **Under Component Procedures click the** `Run` **action next to** `createJDBCConnectionPool.`

4 **Select the installation on which you want to create the connection pool.**

5 **Under Plan Variables:**

    a. **Set the name of the Connection Pool.**

    b. **(Optional) Enter a description of the connection pool.**

    c. **Enter the JDBC data source resource manager for your connection pool.**

    d. **Enter the interface the data source class implements.**

    Valid entries are:

      - `DataSource`
      - `ConnectionPoolDataSource`
      - `XADataSource`

    e. **Enter the minimum number of connections created by the pool.**

    f. **Enter the maximum number of connections maintained by the pool.**

g. **Enter the number of connections to be removed when the time out timer expires.**

h. **Enter the maximum number of seconds a connection can be idle in the pool.**

i. **Enter the transaction isolation level for the connection pool.**

   Valid entries are:

   - `read-uncommitted`
   - `read-committed`
   - `repeatable-read`
   - `serializable`

   If the entry is blank, the default isolation level of the JDBC driver is used.

j. **Specify whether the connections should be validated before being allocated to an application. Valid entries are** `true` **or** `false`**.**

k. **Enter any attribute name/value pairs for the JDBC driver.**

6 **Click the Run Plan (includes preflight) button.**

▼ How to Delete a JDBC Connection Pool

1 **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2 **Under SJSAS AppServer Resources→SJSAS 8.1 JDBC Resource click** `Manage JDBC Resources`**.**

3 **Under Component Procedures click the** `Run` **action next to** `deleteJDBCConnectionPool`**.**

4 **Select the installation on which you want to delete the JDBC Connection Pools.**

5 **Under Plan Variables:**

   a. **Enter the name of the connection pool you want to delete.**

   b. **Specify whether all connector resources associated with the connection pool should also be deleted. Valid entries are** `true` **or** `false`**.**

6 **Click the Run Plan (includes preflight) button.**

▼ How to List JDBC Connection Pools

1   From the Common Tasks section in the browser interface, click AppServer 8.1.

2   Under SJSAS AppServer Resources→SJSAS 8.1 JDBC Resource click `Manage JDBC Resources.`

3   Under Component Procedures click the `Run` **action next to** `listJDBCConnectionPools.`

4   Select the installation on which you want to list the JDBC Connection Pools.

5   Click the Run Plan (includes preflight) button.

Managing Thread Pool Resources

The following are component procedures for Sun Java System App Server 8.1 Thread Pool Management:

`install: default`	Marks the component as installed in the Sun N1 Service Provisioning System database.
`uninstall: default`	Marks the component as uninstalled in the Sun N1 Service Provisioning System database.
`createThreadpool`	Creates a Thread Pool Resource on the target host.
`deleteThreadpool`	Deletes a Thread Pool Resource from the target host.
`listThreadpools`	Lists Thread Pool Resources.

▼ How to Install a Thread Pool Resource on a Host

1   From the Common Tasks section in the browser interface, click AppServer 8.1.

2   Under SJSAS AppServer Resources→SJSAS 8.1 Threadpool Resource click `Manage Threadpool Resources.`

3   Under Component Procedures click the `Run` **action next to** `install: default.`

4   In the Plan Parameters area, select the variable settings for the Thread Pool Resource you want to install.

- If the variable settings have been established for this component, select the appropriate settings from the menu.

- If the settings are not available from the menu, click `Select From List`.

   The Select Variable Settings From List window displays.

- To create a new set of variable settings, select Create Set.

  The following list provides commonly-updated variables for the Install component.

  variable set name   Required. A name for the new variable set you create.

  installIdentifier   A unique ID to associate the thread pool resource with an Install component.

  domainName   The domain name in which this thread pool resource will be located

- To use variable components from another component, click Import Set.

  For more information about importing variable sets, see "How to Run a Plan" in *Sun N1 Service Provisioning System 5.2 Operation and Provisioning Guide.*

5   **Select the target host.**

6   **Deselect the** Target Host Set **checkbox.**

7   **If you want to run a detailed check (preflight) before installing, select** Perform Detailed Preflight**.**

8   **Click the Run Plan (includes preflight) button.**

▼ How to Create a Thread Pool Resource

1   **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2   **Under SJSAS AppServer Resources→SJSAS 8.1 Threadpool Resource click Manage Threadpool Resources.**

3   **Under Component Procedures click the** Run **action next to** createThreadpool**.**

4   **Select the installation on which you want to create the thread pool.**

5   **Under Plan Variables:**

    a. Enter the thread pool ID.

    b. Enter the maximum number of threads in the thread pool servicing requests.

    c. Enter the minimum number of threads in the thread pool servicing requests.

    d. Enter the idle time out value, in seconds, after which idle threads will be returned to the pool.

    e. Enter the total number of work queues serviced by the thread pool.

      f.   **Enter the cluster or server instance name on which the thread pool will be created.**

6   **Click the Run Plan (includes preflight) button.**

▼ How to Delete a Thread Pool Resource

1   **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2   **Under SJSAS AppServer Resources→SJSAS 8.1 Threadpool Resource click** `Manage Threadpool Resources`**.**

3   **Under Component Procedures click the** `Run` **action next to** `deleteThreadpool`**.**

4   **Select the installation on which you want to delete the thread pool.**

5   **Click the Run Plan (includes preflight) button.**

▼ How to List Threadpools on a Target

1   **From the Common Tasks section in the browser interface, click AppServer 8.1.**

2   **Under SJSAS AppServer Resources→SJSAS 8.1 Threadpool Resource click** `Manage Threadpool Resources`**.**

3   **Under Component Procedures click the** `Run` **action next to** `listThreadpools`**.**

4   **Select the installation on which you want to list the thread pools.**

5   **Under Plan Variables enter the target cluster or server instance name on which you want to list the thread pools.**

6   **Click the Run Plan (includes preflight) button.**

Troubleshooting

This section describes common problem you might encounter while using the Sun Java System Application Server Plug-In 3.0.

If You Encounter Errors When Running a Task

If you encounter an error when running a task, double check that you didn't make these common mistakes.

Check Your Variable Values

When you create a variable set used by a plan, it is easy to enter the wrong variable value. For example, you might incorrectly enter a cluster name instead of a domain name.

Unavailable Ports

When entering ports in a variable set, check to make sure the ports are available on the target host.

Component Directory Namespace Problems When Installing Applications

The `com.sun.sjsas81` namespace is invalid for user-supplied applications. When you enter a component directory for an application, make sure you change the namespace, as `com.sun.sjsas81` is selected by default.

Problems During Installation

If you encounter errors when trying to install plug-in components, make sure the `sourceImage` variable points to a directory containing a valid install image of Sun Java System App Server.

Problems in Plug-In Deletion

The user should make sure he uninstalls/deletes even the hidden components before deleting a plug-in. More details on plug-in deletion can be found in a `ReadMe.txt` file after importing the plug-in.

Index