Sun N1 Service Provisioning System User's Guide and Release Notes for the WebSphere Plug-In 2.0

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

Part No: 819–2904–10
September 2005

Copyright 2005 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any. Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

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

Sun, Sun Microsystems, the Sun logo, docs.sun.com, AnswerBook, AnswerBook2, Java, N1, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

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

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

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

Copyright 2005 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. Tous droits réservés.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a. Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées du système Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, docs.sun.com, AnswerBook, AnswerBook2, Java, N1, et Solaris sont des marques de fabrique ou des marques déposées, de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

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

CETTE PUBLICATION EST FOURNIE "EN L'ETAT" ET AUCUNE GARANTIE, EXPRESSE OU IMPLICITE, N'EST ACCORDEE, Y COMPRIS DES GARANTIES CONCERNANT LA VALEUR MARCHANDE, L'APTITUDE DE LA PUBLICATION A REPONDRE A UNE UTILISATION PARTICULIERE, OU LE FAIT QU'ELLE NE SOIT PAS CONTREFAISANTE DE PRODUIT DE TIERS. CE DENI DE GARANTIE NE S'APPLIQUERAIT PAS, DANS LA MESURE OU IL SERAIT TENU JURIDIQUEMENT NUL ET NON AVENU.





050827@12762

Contents

Preface 9

1 Overview of WebSphere 2.0 Plug-In 13
Purpose of the WebSphere 2.0 Plug-In 13
What the WebSphere 2.0 Plug-In Includes 13
Requirements for Using the WebSphere 2.0 Plug-In 14

2 Release Notes for WebSphere 2.0 Plug-In 15
Installation Issues 15
Additional Patch Required 15
`NullPointer` Exception When Deploying Attached `trade3.ear` With a Modified Install Name (6302253) 16
Runtime Issues 16

3 Installing and Configuring the WebSphere Plug-In 17
Acquiring the WebSphere Plug-In 17
Adding the WebSphere Plug-In to the Sun N1 Service Provisioning System 18
▼ How to Import the WebSphere Plug-In Using the Browser Interface 18
▼ How to Import the WebSphere Plug-In using the CLI 18
▼ How to Configure the Master Server for the WebSphere Plug-In 19
Upgrading Considerations 20
▼ How to Access Previous Versions of a Component 20

4 Using the WebSphere 2.0 Plug-In 21
Configuring a WebSphere Application Server Version 5.1 Environment 21

Plug-In Conventions 21
Global Prerequisite 22
Process Overview 23
Creating WebSphere Infrastructure Components 24
▼ How to Install a Standalone WebSphere Application Server 24
▼ How to Create an Application Server Instance 26
▼ How to Install a Deployment Manager 28
▼ How to Add Nodes to the Deployment Manager 30
▼ How to Create a WebSphere Cluster 31
▼ How to Add Cluster Members 32
Configuring the WebSphere 2.0 Environment 33
▼ How to Create a JDBC Provider Component 34
▼ How to Install a JDBC Provider Component 35
▼ How to Create a Data Source Component 35
▼ How to Install a Data Source Component 36
▼ How to Configure the JVM Component 37
▼ How to Install a JVM Component 38
Managing the WebSphere 2.0 Infrastructure Components 39
▼ How to Start or Stop an Application Server for Standalone Systems 39
▼ How to Uninstall an Application Server for Standalone Systems 40
▼ How to Start or Stop an Application Server Instance 40
▼ How to Uninstall an Application Server Instance 41
▼ How to Start or Stop a Deployment Manager 41
▼ How to Uninstall a Deployment Manager 42
▼ How to Start or Stop a Cluster 42
▼ How to Uninstall a Cluster 43
▼ How to Remove Cluster Members 43
Using the Command-Line Interface to Work with WebSphere 2.0 44
Using the Sun N1 Service Provisioning System Command-Line Interface 44
Using the WebSphere 5.1 `wsadmin` Script 44
▼ How to Run the `wsadmin` Script 45
Using a Component Type to Deploy WebSphere Applications 45
Enterprise Application (EAR) Component Type 46
Plans 48
Troubleshooting 48
Uninstalling the WebSphere 2.0 Plug-in 49
▼ How to Uninstall the WebSphere 2.0 Plug-in. 49

Index 51

Tables

TABLE 4–1 Common Task Names 22

Preface

This book explains how to use the Sun N1™ Service Provisioning System software to capture and deploy WebSphere 2.0 application server products, applications, and files.

Who Should Use This Book

The main audience for the Sun N1 Service Provisioning System User's Guide for the WebSphere Plug-In 2.0 includes system administrators and operators of Sun N1 Service Provisioning System software who want to be able to incorporate WebSphere 2.0 functionality with Sun N1 Service Provisioning System software. These users are expected to be familiar with the following:

- The Sun N1 Service Provisioning System product
- Standard UNIX® and Windows commands and utilities
- The general concepts and management features available in the WebSphere 2.0 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 5.1 System Administration Guide*
- *Sun N1 Service Provisioning System 5.1 Operation and Provisioning Guide*
- *Sun N1 Service Provisioning System 5.1 Release Notes*

How This Book Is Organized

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

Chapter 2 describes late-breaking news and known issues with the WebSphere 2.0 plug-in.

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

Chapter 4 explains how to capture and deploy application server products, 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

Sun Function	URL	Description
Documentation	`http://www.sun.com/documentation/`	Download PDF and HTML documents, and order printed documents

Sun Function	URL	Description
Support and Training	`http://www.sun.com/supportraining/`	Obtain technical support, download patches, and learn about Sun courses

Typographic Conventions

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

TABLE P–1 Typographic Conventions

Typeface or Symbol	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	Command-line placeholder: replace with a real name or value	The command to remove a file is `rm` *filename*.
AaBbCc123	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the *User's Guide*. Perform a *patch analysis*. Do *not* save the file. [Note that some emphasized items appear bold online.]

Shell Prompts in Command Examples

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

TABLE P–2 Shell Prompts

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

CHAPTER 1

Overview of WebSphere 2.0 Plug-In

This chapter explains general information about using N1™ Grid Service Provisioning System to provision WebSphere 2.0 applications. This chapter includes the following information:

- "Purpose of the WebSphere 2.0 Plug-In" on page 13
- "What the WebSphere 2.0 Plug-In Includes" on page 13
- "Requirements for Using the WebSphere 2.0 Plug-In" on page 14

Purpose of the WebSphere 2.0 Plug-In

The Sun N1 Service Provisioning System software provides enhanced capabilities in support for WebSphere 2.0 application server products. You can provision with the WebSphere 2.0 infrastructure by installing application servers, deployment managers, and clusters with the provisioning software. Once installed, you can manage your WebSphere 2.0 environment by starting and stopping application servers. You can also capture previously installed applications and deploy these applications throughout your enterprise.

What the WebSphere 2.0 Plug-In Includes

The WebSphere plug-in contains several components and plans that allow you to configure and install WebSphere 2.0 application server products and the applications deployed on WebSphere.

- Plans for installing enterprise applications (EAR files).
- Components and plans to support the WebSphere 2.0 infrastructure:
  - WebSphere standalone application server
  - WebSphere application server instances
  - WebSphere Network deployment manager
  - WebSphere cluster
  - WebSphere cluster members
  - WebSphere cells
- Components and plans to capture and deploy applications on the WebSphere 2.0 application servers
- Components and plans to configure the WebSphere 2.0 environment:
  - Data source
  - JDBC provider
  - Virtual Machine for the Java™ platform (Java Virtual Machine or JVM™ machine or JVM™ tool interface)
- Components and plans that allow you to run WebSphere 2.0 commands from the provisioning system's browser interface.

Requirements for Using the WebSphere 2.0 Plug-In

The WebSphere plug-in supports WebSphere Application Server Version 5.1 on the following platforms:

- Solaris 8 for SPARC
- Solaris 9 for SPARC
- RedHat Linux AS 2.1
- RedHat Linux AS 3.0
- AIX 5.1 and 5.2

Note – Although WebSphere Application Server Version 5.1 is compatible with AIX 5.3, the N1 Grid Service Provisioning System software does not support AIX 5.3.

Detailed requirements for using the WebSphere software are provided in the IBM documentation. For more information on hardware and software requirements, see the WebSphere Application Server support documentation (`http://www-306.ibm.com/software/webservers/appserv/doc/v51/prereqs/prereq51.html`).

CHAPTER 2

Release Notes for WebSphere 2.0 Plug-In

This chapter describes late-breaking news and known issues for the WebSphere 2.0 plug-in.

This chapter contains the following information:

- "Installation Issues" on page 15
- "Runtime Issues" on page 16

Installation Issues

The following issue is known to exist when installing the WebSphere 2.0 plug-in.

Additional Patch Required

To run the WebSphere 2.0 plug-in, you must install patch 119655-01 to the Sun N1 Service Provisioning System software. To download the patch, go to `http://sunsolve.sun.com`.

Note – If you install patch 119655–01 and you also plan to use the Windows 2000 Plug-In, you must also install patch 119656–01.

`NullPointer` Exception When Deploying Attached `trade3.ear` With a Modified Install Name (6302253)

A NullPointer exception is thrown when deploying attached `trade3.ear` with a modified install name. The following error message is displayed:

```
Error in deploying the application
```

Workaround: Ensure that the `installName` variable has a `.ear` extension.

Runtime Issues

There are no known runtime issues.

CHAPTER 3

Installing and Configuring the WebSphere Plug-In

This chapter explains how to install and configure the WebSphere plug-in. The chapter contains the following information:

- "Acquiring the WebSphere Plug-In" on page 17
- "Adding the WebSphere Plug-In to the Sun N1 Service Provisioning System" on page 18

Acquiring the WebSphere Plug-In

The IBM WebSphere Application Server solution is packaged as a *plug-in* to the Sun N1 Service Provisioning System software. Plug-ins are packaged in Java™ Archive (JAR) files. The plug-in files for the WebSphere Application Server solution are available from the Sun N1 Service Provisioning System Supplement CD or from the Sun Download Center.

The WebSphere Plug-In is available for import from two different JAR files. Choose the correct file depending on your situation.

- If you are importing the WebSphere Plug-In for the first time, acquire the `com.sun.was_2.0.jar` file.
- If you have already imported the previous version of the WebSphere Plug-In, acquire the `com.sun.was_1.0_2.0.jar` file.

Adding the WebSphere 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 WebSphere Plug-In you need to upgrade to the new plug-in.

▼ How to Import the WebSphere 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 5.1 System Administration Guide*

Steps

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 where you downloaded the JAR file.**
   - If you are importing the WebSphere Plug-In for the first time, select the `com.sun.was_2.0.jar` file.
   - If you have already imported a previous version of the WebSphere Plug-In, select the `com.sun.was_1.0_2.0.jar` file.
4. **Click the Continue to Import button.**

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

▼ How to Import the WebSphere Plug-In using the CLI

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

Step

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

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

  - If you are importing the WebSphere Plug-In for the first time, *plugin-filename* is `com.sun.was_2.0.jar`.

- If you have already imported the previous version of the WebSphere Plug-In, *plugin-filename* is `com.sun.was_1.0_2.0.jar`.

▼ How to Configure the Master Server for the WebSphere Plug-In

The WebSphere plug-in requires some minor modifications of the basic provisioning system configuration to allow communication between applications and to allow sufficient time for plans to run.

Steps

1. **Import the plug-in.**

   See "Acquiring the WebSphere Plug-In" on page 17.

2. **Set the WebSphere 2.0 session variables: `WS_DEFAULT_USER` and `WS_DEFAULT_PASSWORD`.**

   These variables are used during standalone application server installation, deployment manager installation, and node federation.

3. **Create a virtual host to represent the provisioning system's command-line interface (CLI).**

   Use the following attributes to define the virtual host.

   - host: **`cliHost`**
   - host type: **`com.sun.was#cliHT`**
   - `cliLoc`: the absolute path to the CLI.

     For example, *InstallPath*`/cli/bin/cr_cli`
   - parent host: The remote agent located on the Master Server.

4. **Install the CLI on the machine running the Master Server.**

5. **Install and start the Remote Agent software on the machine running the Master Server.**

6. **Go to the Master Server's Host Edit page and select the Include Remote Agent On This Physical Host option.**

7. **Modify the Master Server's `config.properties` file.**

   The `allowSessionIDOnHosts` value enables the plug-in communicate with the Master Server through the CLI and Remote Agent interface.

   The `defaultPlanTimeout` value enables you to increase the amount of time that a plan can run. The value is represented in seconds. Add this line if you need to increase the default plan run time from the 30–minute (1800 second) default value. The above example is an arbitrary value of 100 minutes (6000 seconds).

The `pe.nonPlanExecNativeTimeout` value is in seconds. Add this line if your site needs to increase the time for execNative timeouts.

```
pe.nonPlanExecNativeTimeout=2400
```

Add the following lines to the file.

```
config.allowSessionIDOnHosts=masterserver
pe.defaultPlanTimeout=6000
```

The `config.properties` file's default location is `/opt/SUNWn1sps/N1_Service_Provisioning_System_5.1/server/config`.

8. **Restart the Master Server.**

Upgrading Considerations

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

▼ 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:

Steps

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.**

CHAPTER 4

Using the WebSphere 2.0 Plug-In

The WebSphere 2.0 plug-in provides several specific plans and component types and provides easy access to functions that are useful for working with WebSphere 2.0 applications. This chapter describes the following information:

- "Configuring a WebSphere Application Server Version 5.1 Environment" on page 21
- "Creating WebSphere Infrastructure Components" on page 24
- "Configuring the WebSphere 2.0 Environment" on page 33
- "Managing the WebSphere 2.0 Infrastructure Components" on page 39
- "Using a Component Type to Deploy WebSphere Applications" on page 45
- "Plans" on page 48
- "Troubleshooting" on page 48

Configuring a WebSphere Application Server Version 5.1 Environment

Plug-In Conventions

The provisioning system enables you to provision and manage applications.

Plans and Component Procedures

The WebSphere 2.0 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. Any task that you can perform using a plan included with the WebSphere Plug-In, you can perform by using a component procedure.

Task Names

The WebSphere 2.0 plug-in organizes tasks in four categories: application tasks, infrastructure tasks, configuration tasks, and advanced tasks. These four task categories follow the same convention for describing the task that you must perform.

TABLE 4–1 Common Task Names

Task Name	Description
View All	Links to all components of a particular type within a specified folder.
Create	Links to the component Details page where you can add a new component of this type to the provisioning system. You can create new components by importing them from an existing application server or by browsing the file system.
Start or Stop	Links to the component Details page to run the Start or Stop component procedure.
Install, Uninstall, Remove, Join, Leave	Links to plans that produce the described functionality.

Global Prerequisite

All tasks within the provisioning system require that you have specific permissions. To perform tasks with the WebSphere 2.0 plug-in, 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.

For more information about permissions, see Chapter 3, “Controlling Access Using Permissions,” in *Sun N1 Service Provisioning System 5.1 System Administration Guide*.

For information about how to update your group membership, see “How to Change a User’s Group Membership” in *Sun N1 Service Provisioning System 5.1 System Administration Guide*.

Process Overview

Creating a WebSphere 2.0 environment within the provisioning system is similar to the process of creating the environment without the provisioning system.

1. Import the WebSphere 2.0 plug-in.

   See "Adding the WebSphere Plug-In to the Sun N1 Service Provisioning System" on page 18.

2. Configure the provisioning system's Master Server.

   See "How to Configure the Master Server for the WebSphere Plug-In" on page 19.

3. Set the WebSphere 2.0 session variables: `WS_DEFAULT_USER` and `WS_DEFAULT_PASSWORD`.

   These variables are used during standalone application server installation, deployment manager installation, and node federation.

   See Chapter 5, "Session Variables," in *Sun N1 Service Provisioning System 5.1 Plan and Component Developer's Guide*.

4. Prepare all hosts.

   The WebSphere 2.0 plug-in contains new system services that must be pushed out to all hosts in the provisioning system. See "Preparing a Physical Host" in *Sun N1 Service Provisioning System 5.1 System Administration Guide*.

5. Create a standalone WebSphere application server.

   See "How to Install a Standalone WebSphere Application Server" on page 24.

6. Create additional application server instances.

   See "How to Create an Application Server Instance" on page 26.

7. (Optional) Create the deployment manager.

   See "How to Install a Deployment Manager" on page 28.

8. If necessary, add each WebSphere application server to the deployment manager.

   See "How to Add Nodes to the Deployment Manager" on page 30.

9. (Optional) Create a cluster and add cluster members to the cluster.

   See "How to Create a WebSphere Cluster" on page 31 and "How to Add Cluster Members" on page 32.

10. If necessary, configure the application's connection to the database.

    a. Configure the JDBC Provider.

       See "How to Create a JDBC Provider Component" on page 34.

    b. Configure the data source.

       See "How to Create a Data Source Component" on page 35.

11. Configure the Java Virtual Machine (JVM) component.

    See "How to Configure the JVM Component" on page 37.

12. Capture the application files.

13. Deploy applications to target application servers or clusters.

    An application can target either an application server or a cluster.

    See "Installing an Enterprise Application" on page 46.

Creating WebSphere Infrastructure Components

Before you can deploy applications to WebSphere application servers, you must either install the WebSphere components by using the provisioning system or capture the existing WebSphere configuration.

▼ How to Install a Standalone WebSphere Application Server

You must install a standalone application server before you can set up application server instances to service your applications. A standalone application server can hold a set of application server instances. The standalone application server can be managed as a single entity or by a deployment manager. All information about the application server is contained both on the application server and, if used, on the deployment manager.

When you create a new standalone application server by installing the component, you also:

- Install the WebSphere 2.0 application software
- Configure the standalone application server
- Create the standalone application server's virtual host
- Create the component and virtual host for one application server instance.

You can use this procedure to create a standalone WebSphere application server or you can use this procedure to register an existing standalone application server within the provisioning system.

Note – This task is performed as root.

Before You Begin

To create a standalone application 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 deploy the component.

Steps

1. **Ensure that your WebSphere session variables, `WS_DEFAULT_USER` and `WS_DEFAULT_PASSWORD`, are set.**

   For more information about session variables, see Chapter 5, "Session Variables," in *Sun N1 Service Provisioning System 5.1 Plan and Component Developer's Guide*.

2. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**

3. **Click the WebSphere Application Server for Standalone Systems: Install link.**

   The plan's Details page displays.

4. **Click Run.**

   The plan's Run page displays.

5. **In the Plan Parameters area, select the variable settings for the `StandaloneInstall` component that you plan to deploy.**

   - **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 `StandaloneInstall` component. You can keep the default values for most variables.

Note – If you change the default port values for the standalone application server, ensure that they do not conflict with the deployment manager's port values.

---

variable set name	Required. A name for the new variable set you create.
`installPath`	Required. The location where you plan to install WebSphere 2.0. The default value for `installPath` is the install path that was used during the most recent installation of a WebSphere component.
`installerHome`	Required. The location of the standalone application server installer.

Note – Do not include extra spaces after the path name.

`name`	Required. The name of the standalone application server virtual host.

- **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 5.1 Operation and Provisioning Guide*.

6. **In the Plan Parameters area, select the `Default` variable set for the `AppInstance` component that you plan to deploy.**

Note – If you decide to create a new variable set, do not change the application server instance name. The name must be `server1`, otherwise the installation will fail.

---

7. **Select the target host.**

   The standalone application server is installed on the remote agent that you select as target host.

Note – The target host must be a member of the `com.sun.was#InstallTargetsHS` host set.

---

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

9. **If you are capturing an existing standalone application server within the provisioning system, select the `markOnly Install` option.**

   This option enables you to create a component that represents an existing WebSphere 2.0 object and then manage the object through the provisioning system.

10. **Click Run Plan (Includes Preflight).**

11. **View the Hosts page to verify that the standalone application server and application server instance virtual hosts were created.**

▼ How to Create an Application Server Instance

When you create a standalone application server, you automatically create one application server instance, `server1`. If you must create additional application server instances, use this procedure.

You can use this procedure to create an application server instance or you can use this procedure to capture an existing application server instance within the provisioning system.

Note – This task is performed as root.

Before You Begin

Before you create an application server instance, you must create a standalone application server. For more information, see "How to Install a Standalone WebSphere Application Server" on page 24.

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**
2. **Click the Application Server Instances: Install link.**
   The plan's Details page displays.
3. **Click Run.**
   The plan's Run page displays.
4. **In the Plan Parameters area, select the variable settings for the `AppInstance` component that you plan to deploy.**
   Each new application server instance needs a unique name. You must create a new variable set to indicate the name for each new application server instance.
   - **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 `AppInstance` component. The `AppInstance` variable settings rely primarily on the standalone application server's variable setting values. You can keep the default values for most variables.

variable set name	Required. A name for the new variable set you create.
`name`	Required. The name of the application server instance.

     - **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 5.1 Operation and Provisioning Guide*.
5. **Select the target host.**
   Install the application server instance on the virtual host of a standalone application server.
6. **Deselect the Target Host Set checkbox.**

7. **If you are capturing an existing standalone application server within the provisioning system, select the `markOnly Install` option.**

   This option enables you to create a component that represents an existing WebSphere 2.0 object and then manage the object through the provisioning system.

8. **Click Run Plan (Includes Preflight).**

9. **View the Hosts page to verify that the new application server instance virtual host was created.**

▼ How to Install a Deployment Manager

When your configuration contains multiple standalone application servers you can manage them with a deployment manager. The deployment manager tracks which applications are being deployed to which application server instance.

You can use this procedure to create a deployment manager or you use this procedure to register an existing deployment manager within the provisioning system.

Note – This task is performed as root.

Steps

1. **Ensure that your WebSphere session variables, `WS_DEFAULT_USER` and `WS_DEFAULT_PASSWORD`, are set.**

   For more information about session variables, see Chapter 5, "Session Variables," in *Sun N1 Service Provisioning System 5.1 Plan and Component Developer's Guide*.

2. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**

3. **Click the Network Deployment Manager: Install link.**

   The plan's Details page displays.

4. **Click Run.**

   The plan's Run page displays.

5. **In the Plan Parameters area, select the variable settings for the `NetworkDeploymentInstall` component that you plan to deploy.**

   - **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 `NetworkDeploymentInstall` variable settings rely primarily on the standalone application server's variable setting values.

Note – If you change the default port values for the deployment manager, ensure that they do not conflict with the standalone application server's port values.

---

variable set name	Required. A name for the new variable set you create.
`installPath`	Required. The location where you plan to install the WebSphere deployment manager. The default value for `installPath` is `/opt/was51nd`.
`installerHome`	Required. The location of the deployment manager installer **Note –** Do not include extra spaces after the path name.
`name`	Required. The name of the deployment manager's virtual host.
`httpServerPort`	Required for regular web access. The web server port number.
`httpsTrnsprtPort`	Required for secure web access. The secure web server port number.
`bootstrap`	Required. The bootstrap port. You must change the port number if you install a standalone application server and a deployment manager on the same remote agent.

- **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 5.1 Operation and Provisioning Guide*.

6. **Select the target host.**

   Install the deployment manager on a remote agent.

Note – The target host must be a member of the `com.sun.was#InstallTargetHS` host set.

---

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

8. **If you are capturing an existing standalone application server within the provisioning system, select the `markOnly Install` option.**

   This option enables you to create a component that represents an existing WebSphere 2.0object and then manage the object through the provisioning system.

9. **Click Run Plan (Includes Preflight).**

10. **Verify that the new deployment manager virtual host was created by viewing the Hosts page.**

▼ How to Add Nodes to the Deployment Manager

For a deployment manager to track which applications are deployed to different standalone application servers, you must add the standalone application servers to the deployment manager. The process of adding nodes to the deployment manager is also called *federation*.

Note – This task is performed as root.

Before You Begin

Before you can add nodes to a deployment manager, the deployment manager must be installed. For information on installing a deployment manager, see "How to Install a Deployment Manager" on page 28.

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**

2. **Click the Network Deployment Manager: Federate link.**

   The plan's Details page displays.

3. **Click Run.**

   The plan's Run page displays.

4. **Select the target host.**

   Select the standalone application server virtual host that you plan to federate.

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

6. **Type the name of the deployment manager virtual host with which you plan to federate the standalone application server.**

7. **If you want to copy applications from the federated node to the deployment manager, ensure that the `includeApps` option is set to `true`.**

8. **If you are creating a representation of an existing, federated standalone application server, select the `markOnly Install` field.**

   This option enables you to represent an existing WebSphere 2.0 object and then manage through the provisioning system.

9. **Click Run Plan (Includes Preflight).**

   After federation, the standalone application server virtual host's host type attributes, `cellName` and `dmHost`, are updated with the appropriate information.

▼ How to Create a WebSphere Cluster

When you create a WebSphere cluster through the provisioning system, you create an empty container that can hold cluster members. To add cluster members to a cluster, see "How to Add Cluster Members" on page 32.

Note – This task is performed as root.

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**

2. **Click the Cluster Management: Create link.**

   The plan's Details page displays.

3. **Click Run.**

   The plan's Run page displays.

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

   Each new cluster needs a unique name. You must create a new variable set to indicate the new name for each cluster.

   - **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 `Cluster` component. You can keep the default values for most variables.

variable set name	Required. A name for the new variable set you create.
`name`	Required. The name of the cluster.

`clusterSPSName` Required. The name of the cluster virtual host.

- **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 5.1 Operation and Provisioning Guide*.

5. **Select the target host.**
   Target the virtual host of the deployment manager that will manage the cluster.
6. **Deselect the Target Host Set checkbox.**
7. **If you are creating a representation of an existing cluster, select the `markOnly Install` option.**
   This option enables you to represent an existing WebSphere 2.0 object and then manage through the provisioning system.
8. **Click Run Plan (Includes Preflight).**
9. **View the Hosts page to verify that the cluster virtual host was created.**

▼ How to Add Cluster Members

The procedure creates new cluster members in selected clusters.

Note – This task is performed as root.

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**
2. **Click the Cluster Member Management: Join link.**
   The plan's Details page displays.
3. **Click Run.**
   The plan's Run page displays.
4. **In the Plan Parameters area, select the variable settings for the `ClusterMember` component that you plan to deploy.**
   Each new cluster member needs a unique name. You must create a new variable set to indicate the name for each new cluster member.

   - **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 `ClusterMember` component. You can keep the default values for most variables.

variable set name	Required. A name for the new variable set you create.
`name`	Required. The name of the cluster member.
`cmemberSPSName`	Required. The name of the cluster member virtual host.

- **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 5.1 Operation and Provisioning Guide*.

5. **Select the target host.**
   You target the cluster virtual host where you plan to add the cluster member.
6. **Deselect the Target Host Set checkbox.**
7. **If you are creating a representation of an existing cluster member, select the `markOnly Install` option.**
   This option enables you to represent an existing WebSphere 2.0 object and then manage through the provisioning system.
8. **Click Run Plan (Includes Preflight).**
9. **Click the View Cluster Members link on the WebSphere 5.1 Common Tasks page to verify that the cluster members were created.**

Configuring the WebSphere 2.0 Environment

The WebSphere 2.0 plug-in enables you to create components that configure various aspects of the WebSphere 2.0 environment. This section describes the following tasks:

- "How to Create a JDBC Provider Component" on page 34
- "How to Install a JDBC Provider Component" on page 35
- "How to Create a Data Source Component" on page 35
- "How to Install a Data Source Component" on page 36
- "How to Configure the JVM Component" on page 37

- "How to Install a JVM Component" on page 38

▼ How to Create a JDBC Provider Component

You create the JDBC provider component and then install it. See "How to Install a JDBC Provider Component" on page 35.

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**
2. **Click the JDBC Provider: Create link.**
   The component's Edit page displays.
3. **Type the new JDBC Provider's component name.**
4. **(Optional) Type the label and description of the JDBC Provider.**
5. **Modify the values of the component variables, then click Check In.**
   The following list describes the variables that you must update for the component. You can keep the default values for most variables.

`providerName`	Required. The name for the JDBC Provider that you are creating.
`implementationClassName`	Required. The Java class name of the JDBC driver implementation
`dbClasspath`	Required. The list of paths or JAR file names that form the location for the resource provider classes.

6. **If you do not want to use the default folder, select the folder in which you plan to place the Data Source component by clicking the Change Folder link.**
   Select the folder from the Change Folder window and click Change to Selected Folder.

   Note – You cannot create a new component in a folder that is owned by a plug-in.

   ---

7. **Click Continue to Check In.**
   The Components page displays.

▼ How to Install a JDBC Provider Component

Before You Begin

The JDBC Provider component that you plan to install must exist within the provisioning system. See "How to Create a JDBC Provider Component" on page 34.

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**
2. **Click the JDBC Provider: View All link.**

   The component's Edit page displays.
3. **If necessary, find the component by changing to the root folder and selecting the Show Flat View option.**

   All JDBC Provider components registered with the provisioning system display in the Component table.
4. **In the row of the JDBC Provider component that you plan to install, click Details.**

   The component Details page displays.
5. **In the Component Procedures table, select the Default: Install procedure and click Run.**

   The plan Run page displays.
6. **Select the variable set.**
7. **Select the target host.**

   You can target the data source component to one of the following hosts.

   - Standalone application server virtual host
   - Deployment Manager virtual host
   - Cluster member virtual host
   - Application server instance virtual host
8. **Deselect the Target Host Set checkbox.**
9. **Click Run Plan (Includes Preflight).**

▼ How to Create a Data Source Component

You create a data source component then install it. See "How to Install a Data Source Component" on page 36.

Before You Begin

Before you create a data source component, you must first create and install the JDBC provider component. See "How to Create a JDBC Provider Component" on page 34.

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**
2. **Click the Data Source: Create link.**
   The component's Edit page displays.
3. **Type the new data source's component name.**
4. **(Optional) Type the label and description of the data source.**
5. **If necessary, modify the values of the component variables, then click Check In.**
   The following list describes the variables that you must update for the component. You can keep the default values for most variables.

`jdbcProvider`	Required. The name of the JDBC provider under which the Data Source is created.
`jndiName`	Required. The JNDI name for the data source.
`dsHelperClassName`	Required. The name of the datastore helper that is used to perform specific database functions
`authDataAlias`	Required. The alias to use for run time database authentication.
`dbName`	Required. The database name.
`url`	Required. The URL indicating the database from which the Data Source will obtain connection.

6. **Modify the folder in which you plan to place the Data Source component by clicking the Change Folder link.**
   Select the folder from the Change Folder window and click Change to Selected Folder.

Note – You cannot create a new component in a folder that is owned by a plug-in.

---

7. **Click Continue to Check In.**
   The Components page displays.

▼ How to Install a Data Source Component

Before You Begin The Data Source component that you plan to install must exist within the provisioning system. See "How to Install a Data Source Component" on page 36.

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**

2. **Click the Data Source: View All link.**

   The component's Edit page displays.

3. **If necessary, change to the root folder and select the Show Flat View option to find the component.**

   All data source components registered with the provisioning system display in the Component table.

4. **In the row of the Data Source component that you plan to install, click Details.**

   The component Details page displays.

5. **In the Component Procedures table, select the Default: Install procedure and click Run.**

   The plan Run page displays.

6. **Select the variable set.**

7. **Select the target host.**

   You can target the data source component to one of the following hosts.

   - Standalone application server virtual host
   - Deployment Manager virtual host
   - Cluster member virtual host
   - Application server instance virtual host

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

9. **Click Run Plan (Includes Preflight).**

▼ How to Configure the JVM Component

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**

2. **Click the JVM: Configure link.**

   The component's Edit page displays.

3. **Type the new JVM component name.**

4. **(Optional) Type the label and description of the JVM.**

5. **Modify the values of the component variables, then click Check In.**

   The following list describes the variables that you can update for this component. You can keep the default values for most variables.

   `initHeapSize` Required. The initial heap size.

`maxHeapSize` Required. The maximum heap size.

6. **If you do not want to use the default folder, select the folder in which you plan to place the Data Source component by clicking the Change Folder link.**

   Select the folder from the Change Folder window and click Change to Selected Folder.

Note – You cannot create a new component in a folder that is owned by a plug-in.

---

7. **Click Continue to Check In.**

   The Components page displays.

▼ How to Install a JVM Component

Before You Begin

The JVM component that you plan to install must exist within the provisioning system. See "How to Configure the JVM Component" on page 37.

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**
2. **Click the JVM: View Configurations link.**

   The component's Edit page displays.
3. **If necessary, find the component by changing to the root folder and selecting the Show Flat View option.**

   All JVM components registered with the provisioning system display in the Component table.
4. **In the row of the JVM component that you plan to install, click Details.**

   The component Details page displays.
5. **In the Component Procedures table, select the Default: Install procedure and click Run.**

   The plan Run page displays.
6. **Select the variable set.**
7. **Select the target host.**

   The target host is either an application server or a cluster member.
8. **Deselect the Target Host Set checkbox.**
9. **Click Run Plan (Includes Preflight).**

Managing the WebSphere 2.0 Infrastructure Components

In addition to installing WebSphere 2.0 infrastructure components, you can use the provisioning system to start, stop, and uninstall these components. This section describes the following tasks:

- "How to Start or Stop an Application Server for Standalone Systems" on page 39
- "How to Uninstall an Application Server for Standalone Systems" on page 40
- "How to Start or Stop an Application Server Instance" on page 40
- "How to Uninstall an Application Server Instance" on page 41
- "How to Start or Stop a Deployment Manager" on page 41
- "How to Uninstall a Deployment Manager" on page 42
- "How to Start or Stop a Cluster" on page 42
- "How to Uninstall a Cluster" on page 43
- "How to Remove Cluster Members" on page 43

▼ How to Start or Stop an Application Server for Standalone Systems

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**
2. **Click the WebSphere Application Server for Standalone Systems: Start or Stop link.**

   The component's Details page displays.
3. **In the Component Procedures table, select the startServer or stopServer procedure and click Run.**

   The plan's Run page displays hosts that have the standalone application server installed.
4. **Select the host that you plan to start or stop.**

   You select the host and its installation path from the list of current installations.
5. **Click Run Selected Installations.**
6. **Verify that the plan details are correct and click Run Plan (Includes Preflight).**

▼ How to Uninstall an Application Server for Standalone Systems

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**
2. **Click the WebSphere Application Server for Standalone Systems: Uninstall link.**
   The plan's Details page displays.
3. **Click Run.**
   The plan's Run page displays the remote agents that have the standalone application servers installed.
4. **Select the target host.**
   You target the remote agent on which the standalone application server is installed.
5. **Deselect the Target Host Set checkbox.**
6. **To specify the version of the application server that you plan to uninstall, type the application server's install path.**
   If this value is omitted, the most recently installed component in any path is used. The value is converted to universal format prior to component resolution. This attribute can reference simple substitution variables.
7. **If you want to remove the standalone application server from the provisioning system but want to keep the application server running in the WebSphere environment, select the `markOnly` option**
8. **Click Run Plan (Includes Preflight).**

▼ How to Start or Stop an Application Server Instance

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**
2. **Click the Application Server Instances: Start or Stop link.**
   The component's Details page displays.
3. **In the Component Procedures table, select the startAppInstance or stopAppInstance procedure and click Run.**
   The plan's Run page displays.

4. **Select the standalone application server virtual host on which the application server instance is installed.**
5. **Click Run Selected Installations.**
6. **Verify that the plan details are correct and click Run Plan (Includes Preflight).**

▼ How to Uninstall an Application Server Instance

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**
2. **Click the Application Server Instances: Uninstall link.**
   The plan's Details page displays.
3. **Click Run.**
   The plan's Run page displays.
4. **Select the target host.**
   You target the standalone application server virtual host that you plan to uninstall.
5. **Deselect the Target Host Set checkbox.**
6. **Type the install path of the application instance that you plan to uninstall.**
   If this value is omitted, the most recently installed component in any path is used. The value is converted to universal format prior to component resolution. This attribute can reference simple substitution variables.
7. **If you want to remove the application server instance from the provisioning system but want to keep the application server instance running in the WebSphere environment, select the `markOnly` option.**
8. **Click Run Plan (Includes Preflight).**

▼ How to Start or Stop a Deployment Manager

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**
2. **Click the Network Deployment Manager: Start or Stop link.**
   The component's Details page displays.
3. **In the Component Procedures table, select the startServer or stopServer procedure and click Run.**

The plan's Run page displays.

4. **Select the virtual host on which the deployment manager is installed.**
5. **Click Run Selected Installations.**
6. **Verify that the plan details are correct and click Run Plan (Includes Preflight).**

▼ How to Uninstall a Deployment Manager

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**
2. **Click the Network Deployment Manager: Uninstall link.**
   The plan's Details page displays.
3. **Click Run.**
   The plan's Run page displays.
4. **Select the remote agent on which you installed the deployment manager.**
5. **Type the install path of the deployment manager that you plan to uninstall.**
   If this value is omitted, the most recently installed component in any path is used. The value is converted to universal format prior to component resolution. This attribute can reference simple substitution variables.
6. **If you want to remove the deployment manager from the provisioning system but want to keep the deployment manager running in the WebSphere environment, select the `markOnly` option.**
7. **Click Run Plan (Includes Preflight).**

▼ How to Start or Stop a Cluster

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**
2. **Click the Cluster Management: Start or Stop link.**
   The component's Details page displays.
3. **In the Component Procedures table, select the startCluster or stopCluster procedure and click Run.**
   The plan's Run page displays.

4. **Select the cluster that you plan to start or stop.**
   You select the deployment manager and the cluster name from the list of current installations.
5. **Click Run Selected Installations.**
6. **Verify that the plan details are correct and click Run Plan (Includes Preflight).**

▼ How to Uninstall a Cluster

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**
2. **Click the Cluster Management: Remove link.**
   The plan's Details page displays.
3. **Click Run.**
   The plan's Run page displays.
4. **Select the target host.**
   You target the deployment manager virtual host.
5. **Deselect the Target Host Set checkbox.**
6. **Type the install path of the cluster that you plan to uninstall.**
   If this value is omitted, the most recently installed component in any path is used. The value is converted to universal format prior to component resolution. This attribute can reference simple substitution variables.
7. **If you want to remove the cluster from the provisioning system but want to keep the cluster running in the WebSphere environment, select the `markOnly` option**
8. **Click Run Plan (Includes Preflight).**

▼ How to Remove Cluster Members

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**
2. **Click the Cluster Member Management: Leave link.**
   The plan's Details page displays.
3. **Click Run.**
   The plan's Run page displays.

4. **Select the cluster virtual host as the target host.**
5. **Deselect the Target Host Set checkbox.**
6. **If you want to remove the cluster member from the provisioning system but want to keep the cluster member running in the WebSphere environment, select the `markOnly` option.**
7. **Click Run Plan (Includes Preflight).**

Using the Command-Line Interface to Work with WebSphere 2.0

Using the Sun N1 Service Provisioning System Command-Line Interface

By default, the provisioning system's command-line interface is located in the following directory:
`/opt/SUNWn1sps/N1_Grid_Service_Provisioning_system_5.0/cli/bin/cr_cli`

WebSphere 2.0 plug-in objects are located in the `/com/sun/was/5.1` folder.

You can interact with plug-in components and plans much like any other custom component or plan. However, if you want to make changes to the components or plans, you must save the component or plan to a new folder and make your changes there. For more information about using the provisioning system's command-line interface, see Chapter 1, "Using the Command-Line Interface," in *Sun N1 Service Provisioning System 5.1 Command-Line Interface Reference Manual*

Using the WebSphere 5.1 `wsadmin` Script

You can run the WebSphere 2.0scripting program, `wsadmin`, through the Sun N1 Service Provisioning System. For more information about how to use the scripting program, see IBM's WebSphere 5.1 documentation.

▼ How to Run the `wsadmin` Script

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**
2. **Click the Command Line Interface: Run Command link.**
   The plan's Details page displays.
3. **Click Run.**
4. **Select the target host.**
   You can target the virtual host of the standalone application server or deployment manager.
5. **Type the arguments for the `wsadmin` script in the Plan Variables field.**
6. **Click Run Plan (Includes Preflight).**

Example 4–1 Creating a Cluster Using the `wsadmin` Script

This example illustrates input for the `wsadmin` script.

If you wanted to create a cluster using the command line, you could type the following string in the Enter the Arguments to `wsadmin` Script field.

```
-c 'set cell [$AdminConfig getid /Cell:cellname/]'
-c '$AdminConfig create ServerCluster $cell {{name clustername}}'
-c '$AdminConfig save'
```

In this example, *cellname* is the name of the existing cell and *clustername* is the cluster that you plan to create.

Using a Component Type to Deploy WebSphere Applications

The WebSphere Plug-In includes several WebSphere-specific component types. These component types enable you to quickly model many of the most common WebSphere application components and to automatically associate install, uninstall, export, and snapshot behavior with a particular resource. Many of the component types are used by the software when you create and manage servers and clusters. As a result, there is only one component type that you will work with on a regular basis, as described in the following section:

■ "Enterprise Application (EAR) Component Type" on page 46

Note – The WebSphere Plug-In can only install EAR files that were assembled using the IBM assemble tool. These files contain configuration settings that the plug-in requires.

Enterprise Application (EAR) Component Type

A component of this type models an enterprise application. The component can contain either an enterprise application archive (EAR) or the expanded version of an EAR as a package.

Browsing for an Enterprise Application

You can use one of the following browsers to select your enterprise application for this component type:

- A WebSphere Appserver browser from which you can select one of the installed applications and its relevant settings. The Appserver browser provides an alphabetical list of enterprise applications on the application server. You can choose one enterprise application for a component.
- A file system browser from which you can select the EAR file to create a component that does not include settings. The file system browser provides a list of files and directories from which you can choose. You can choose to see only those files or type `*.ear` in the browser.

Note – The WebSphere Plug-In can only install EAR files that were assembled using the IBM assemble tool. These files contain configuration settings that the plug-in requires.

Installing an Enterprise Application

To install an enterprise application component, you must use a WebSphere application server instance or cluster as the target host. The installation process performs three main tasks:

1. Installs the EAR file on the target.
2. Registers the EAR with the WebSphere Application Server.

You can install the enterprise application by navigating to the application's component Details page and running the Install component procedure.

Note – For the changes to take effect, you might need to restart the application server after the application is installed

---

Uninstalling an Enterprise Application

When you uninstall an enterprise application component, the enterprise application no longer targets the virtual host. If the enterprise application is not targeted elsewhere, the registration component is removed from the application server .

You can uninstall the enterprise application by navigating to the application's component Details page and running the Uninstall component procedure.

Starting and Stopping an Enterprise Application

You can start and stop enterprise applications that have been installed with the provisioning system.

▼ How to Start or Stop an Enterprise Application

Steps

1. **In the Common Tasks section in the browser interface, click WebSphere 5.1.**
2. **Click the Enterprise Applications (EAR): View All link.**
   The Components page displays.
3. **Select the enterprise application that you plan to start or stop, and click Details.**
   The component Details page displays.
4. **In the Component Procedures table, select the Start or Stop procedure and click Run.**
   The plan's Run page displays.
5. **Select the virtual host of the application server that hosts the enterprise application that you plan to start or stop.**
6. **Click Run Selected Installations.**
   The plan Run page displays.
7. **Click Run Plan (Includes Preflight).**

Installing Web Application (WAR) and Java Archive (JAR) File

WebSphere does not support WAR and JAR files. If you need to install a WAR or JAR file through the provisioning system, use the WebSphere Assemble tool to convert the file to the EAR format. After you export the file, you can use the provisioning system to deploy it.

Plans

The WebSphere Plug-In relies on plans to perform many of the tasks associated with installing and removing application servers, deployment managers, and clusters.

WebSphere 2.0 plug-in plans are located in the `/com/sun/was/5.1` folder.

Troubleshooting

If you are experiencing problems managing WebSphere through the provisioning software, messages will likely appear on the screen to tell you that a problem exists.

```
Problems encountered during plan run or preflight

The plan (or preflight) "/com/sun/was/5.1/DeleteCluster" finished
with 1 failed host(s). (017034)

Cannot perform operation on component component
/com/sun/was/5.1/Cluster because it was not installed on host sun-1 (017057)
Unable to resolve installed component for component
/com/sun/was/5.1/Cluster. (602021)
```

Although the message in this example is easily deciphered, not all messages are this straightforward. To find out more information about a specific problem, follow the Run History Details links. Drill down to the final error block and view the output of `stderr` and `stdout` on the error condition.

Uninstalling the WebSphere 2.0 Plug-in

▼ How to Uninstall the WebSphere 2.0 Plug-in.

Before You Begin

You cannot uninstall a plug-in if any object owned by the plug-in is being used by the provisioning system.

To uninstall a plug-in, you must be a member of the `admin` user group.

Steps

1. **Ensure that all objects owned by the plug-in are not in use.**

   The following components must be uninstalled from the hosts on which they were installed.

   - WebSphere applications
   - WebSphere application servers
   - Deployment Managers
   - Clusters
   - Cluster Members

   You cannot delete a plug-in if any object owned by the plug-in is being used by the provisioning system.

2. **In the Application Deployment section of the browser interface, click Plans.**
3. **If necessary, navigate to the `/com/sun/was/5.1` folder.**
4. **Select `UninstallSystemServicesPlan` and click Details.**

   The plan's Details page is displayed.
5. **Click Run.**
6. **Verify that the `com.sun.was#InstallTargetHS` target host set is selected.**
7. **Click Run Plan (Includes Preflight).**

   The WebSphere system services are deleted.
8. **In the Administrative section of the browser interface, click Plug-ins.**
9. **Select the plug-in that you want to uninstall and click Details.**

   The plug-in Details page displays.
10. **At the bottom of the Details page, click Delete.**

    You are asked to verify that you want to delete the plug-in.

11. Click Continue to Delete.

After the plug-in is deleted, the Plug-in page is updated.

Index

A

adding
  cluster members, 32-33
  nodes to the deployment manager, 30-31
application server instances
  creating, 26-28
  starting, 40-41
  stopping, 40-41
  uninstalling, 41

B

browsing, enterprise applications, 46

C

cluster members
  adding, 32-33
  removing, 43-44
cluster nodes, *See* cluster members
clusters
  creating, 31-32
  starting, 42-43
  stopping, 42-43
  uninstalling, 43
component type, enterprise application, 46-48
configuring, JVM components, 37-38
creating
  application server instances, 26-28
  clusters, 31-32
  data source components, 35-36
creating (Continued)
  deployment managers, 28-30
  JDBC Provider components, 34
  standalone application servers, 24-26

D

data source components
  creating, 35-36
  installing, 36-37
deleting, *See* uninstalling
deployment managers
  creating, 28-30
  starting, 41-42
  stopping, 41-42
  uninstalling, 42
deployment server requirements, 14

E

EAR, *See* enterprise application
enterprise application, component type, 46-48
enterprise applications
  browsing, 46
  installing, 46-47
  starting, 47
  stopping, 47
  uninstalling, 47

F

federating, *See* adding

I

importing the plug-in, 18-20
installing
  data source components, 36-37
  enterprise applications, 46-47
  JDBC Provider components, 35
  JVM components, 38

J

JAR file, 17
Java archive, *See* JAR file
Java Virtual Machine, *See* JVM component
JDBC Provider components
  creating, 34
  installing, 35
JVM components
  configuring, 37-38
  installing, 38

N

nodes, adding to the deployment
  manager, 30-31

P

plans, using, 48
plug-in
  deployment server requirements, 14
  importing, 18-20
  parts, 13-14
plug-in files
  importing, 18-20
  location of, 17

R

removing
  See also uninstalling
  cluster members, 43-44

S

standalone application servers
  creating, 24-26
  starting, 39
  stopping, 39
  uninstalling, 40
starting
  application server instances, 40-41
  clusters, 42-43
  deployment managers, 41-42
  enterprise applications, 47
  standalone application servers, 39
stopping
  application server instances, 40-41
  clusters, 42-43
  deployment managers, 41-42
  enterprise applications, 47
  standalone application servers, 39

T

troubleshooting, 48

U

uninstalling
  application server instances, 41
  clusters, 43
  deployment managers, 42
  enterprise applications, 47
  standalone application servers, 40
  WebSphere 2.0 plug-in, 49-50
using, plans, 48

W

WebSphere 2.0 plug-in, uninstalling, 49-50