WSM 4.1.1 Installation

1 Purpose

This document contains detailed instructions on installing or updating WSM 4.1 using the default or custom installation process.

The screenshots in this guide are from the Windows installer. The Unix installer provides the same steps and options but has the following differences:

1. The Unix installer is text-based.

2. The WSM Manager and the WSM Development Environment (CQDE) are not available, because they are Windows applications.

3. You cannot install a JRE with the WSM installer. The JRE or JDK must be installed before installing WSM.

2 Installation Requirements

Before you run the installer, make sure that your computer or server fulfills the following installation requirements.

2.1 Running the Installer

To run the installer, you need at least the following:

- Unix: Java 2 Runtime Environment version 1.4.1 or higher

- Windows: Minimum 8-bit color depth (256 colors) and minimum 640 X 480 screen resolution.

- IBM AIX: The Communiqué process requires a minimum of 512 MB Java heap size. Run the installer with the option JVM_MAX_HEAP=512m (either with the -D parameter or with the -f my.properties parameter). After installation, adapt the start script if necessary, so that it starts Communiqué with enough memory.

2.2 Required Disk Space

To install WSM correctly, you need sufficient space in both the temporary directory and the installation directory. The installer checks the free disk space before starting the installation and displays a warning message if it is insufficient.

WSM and Tool Sizes

WSM 3.5.5 Servlet Engine	50 MB
WSM Manager	0.6 MB
WSM Development Environment	20 MB
WSM Dispatcher	6 MB
WSM Connector Resources	6 MB
WSM Documentation	30 MB
Author Instance*	250 MB
Publish Instance*	200 MB
Temporary Directory	350 MB
Default Installation (total)	900 megabytes (MB)

*The amount of disk space you need increases with the number of WSM instances and the add-ons you install with each instance.

Note: Immediately after the installation has finished, the installer deletes the temporary installation files.

You can find the default temporary directory in the following locations:

Platform	Location
Windows 95/98	C:\Windows\Temp
Windows NT	C:\WINNT\Temp
Windows 2000/XP	C:\Documents and Settings\<user>\Local Settings\Temp
Unix	/tmp

Note: To use a different temporary directory, set the environment variable **IATEMPDIR** (Unix) or the user variable **TMP** (Windows) to the desired path before running the installer.

2.3 FileNet P8 Installation

To integrate WSM with P8, you need to add library files to the P8 installation. Refer to the file README.txt in the folder opt/p8 for details.

2.4 Using an Existing CRX Repository

If you want to install WSM on an existing CRX repository, you need to install and configure the CRX repository before you start the WSM installer.

To configure CRX for use with WSM, refer to the instructions in the file wsm-aux-4.1.1.zip.

When you install WSM, do not install the WSM Servlet Engine. The installer then offers you the option of using an existing CRX repository. If you install the Servlet Engine, the installer automatically installs a new CRX repository.

3 Setting Up the Java Environment

WSM runs on a Java Virtual Machine (JVM). There are several ways to provide a Java environment for WSM.

Note: According to the Java Concept Map, the Virtual Machine is a part of the Java Runtime Environment (JRE), which is included in the Java Software Development Kit (Java SDK or JDK). The term "JRE" may indicate a separate JRE as well as a JRE that is part of a JDK.

3.1 Installing a new Java Runtime Environment (JRE)

The WSM installer by default installs a separate Java Runtime Environment (JRE) when you install WSM. This is the quickest and safest way to run WSM. The JRE that the installer provides is a standard JRE that contains an additional library (named tools.jar), which allows WSM to compile Java code and Java Server Pages (JSP) script code.

Note: This option is not available for a Unix installation. On Unix, you have to install a JRE or JDK before you install WSM.

3.2 Using an Existing Java Runtime Environment (JRE)

If you want to use an existing JRE, you have to add the library file "tools.jar", so WSM can compile Java code and Java Server Pages (JSP) script code.

Use this option if you have an existing JRE that you want to use, for example, to reduce the complexity of your system setup and to minimize configuration and maintenance efforts of your Java platform.

To add the file "tools.jar" to your Java environment, proceed as follows:

1. Get the file "tools.jar" for the JRE version you are using. The file is included in the JDK, so you may have to download and install the JDK to get the file.

2. Put the file "tools.jar" into the /lib folder of your JRE installation folder.

During the installation process, proceed as follows:

1. When the installer offers to install a new JRE, uncheck this option.

2. When the installer asks you to specify the directory of the Java SDK you want to use, specify the home directory of the modified JRE. The installer checks whether the file "tools.jar" is available and continues if it is.

3.3 Using a Java Software Development Kit (JDK)

For development, you can run WSM on a standard JDK installation. During the installation process, proceed as follows:

1. When the installer offers to install a new JRE, uncheck this option.

2. When the installer asks you to specify the directory of the Java SDK you want to use, specify the home directory of the JDK.

4 Default Installation

The default installation works for most setups. It installs the full WSM package including two WSM instances (Author and Publish), the WSM Development Environment, the WSM Servlet Engine, a Java Virtual Machine, the WSM Manager application, add-ons, and documentation.

To perform a default installation, click **Default Installation** in the Installation Type screen. This screen appears after you have agreed to the license terms and entered the serial number.
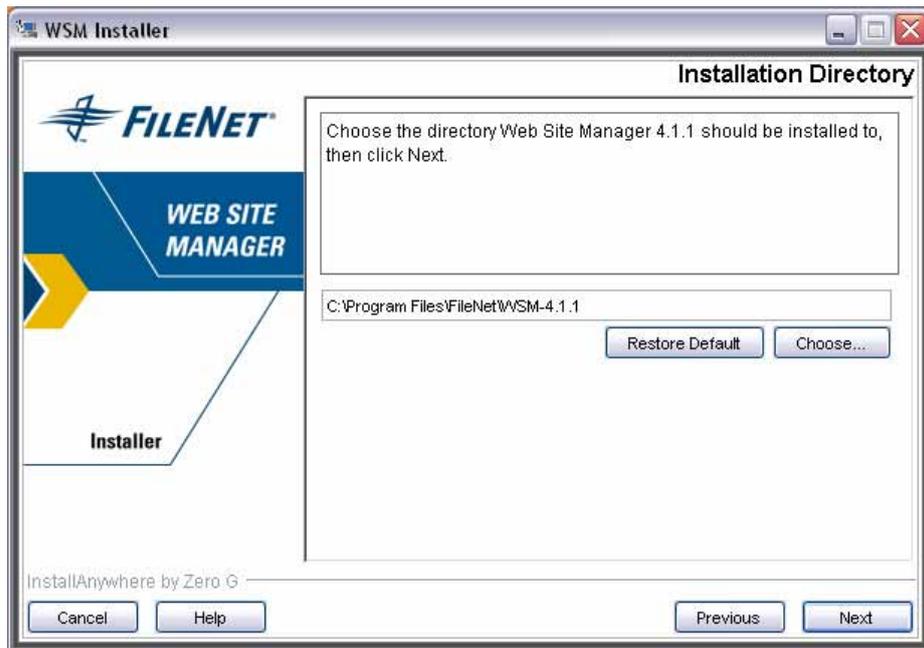
Specify the path of the folder into which you want to install WSM. The installation directory must be on the same computer that is running the installation. It is used to store WSM features, information for uninstalling, and the installation log.

Click **Next** and continue with the **Finishing the Installation Process** section.

5 Custom Installation

The custom installation installs WSM with custom settings. This installation allows you to customize several aspects of the installation process. Use this installation if you do not want to install all WSM's features, install them with custom settings, or if you are updating an existing WSM installation.

5.1 Features

Select the features you want to install.

Author Instance	Installs an author instance as a separate Web application running on the server of your choice (which may be a WSM Servlet Engine or a third-party application server). You can also update an existing author instance with version 3.5.5 or higher.
Publish Instance	Installs a publish instance as a separate Web application running on the server of your choice (which may be a WSM Servlet Engine or a third-party application server). You can also update an existing publish instance with version 3.5.5 or higher

WSM Servlet Engine	Installs a new server to run WSM and all features you choose. If you already have a server and want to update existing instances, you must install the new WSM Servlet Engine as well. If you clear this box, the installer assumes that you have an existing WSM installation that you want to update. This means that your existing server is left as it is, but you will be able to choose either an **update** installation or a **new** installation for each of your existing WSM instances. An update installation of an instance preserves its existing content and upgrades it. A new installation of an instance removes the instance entirely and puts a fresh instance in its place. See the **Update Installation** section for more information.
WSM Manager	Installs a Windows application to manage multiple installations of WSM Servlet Engine, which allows you to start, stop, and configure the server on which WSM is running. This application is not available on Unix. In Unix, use the 'start' and 'stop' scripts instead.
WSM Development Environment	Installs Windows applications to provide help at developing WSM features. The IDE works remotely so you can develop on WSM instances located on other machines. If you install WSM on a computer on which the developers will work, install this option. If the computer is a central server, then you do not need to install this option. This feature is not available on Unix. To develop on a WSM instance installed on a Unix machine you must connect remotely from a WSM Development Environment running on a Windows machine.
WSM Dispatcher and Add-ons	Installs the WSM dispatcher and add-ons. The WSM dispatcher caches responses from the WSM server, thus improving performance. It is used for production systems. If the machine you are installing on is not intended as a production platform, then you do not need to install this option. The add-ons include Early Access Technology, helper tools, and plug-ins for WSM.
WSM Documentation	Installs user guides and API documentation for WSM in PDF form. The same documentation is also available online at *http://docs.day.com.*

Click **Next** and continue with the sections corresponding to the feature you chose.

5.2 Enabling the Connectors

When you install WSM, you can choose to enable the connectors or not. If you do not enable a connector, the installer still installs it, so you can enable it manually later on.
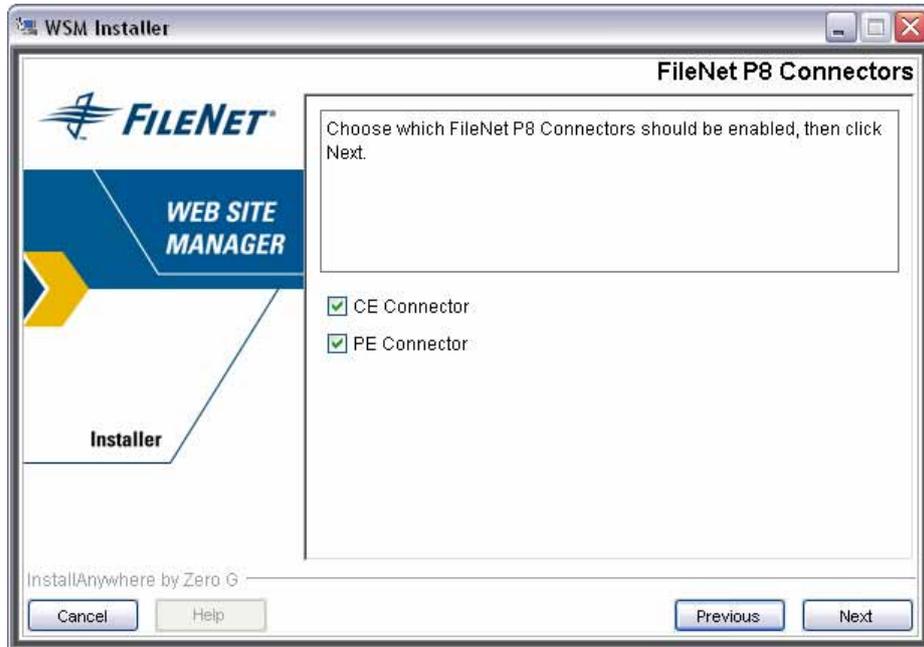
The following connectors are available:

CE Connector	The CE connector connects WSM with the Content Engine of FileNet P8. This allows WSM to access FileNet content.
PE Connector	The PE connector connects WSM with the Process Engine of FileNet P8. This allows WSM to integrate with FileNet workflows.

Note: You can install and run WSM without the connectors. If you do so, WSM will be unable to access P8 content or workflows.

5.2.1 Specifying the CE Connection Details

If you have chosen to enable the CE connector, you have to specify the connection details to the P8 store:
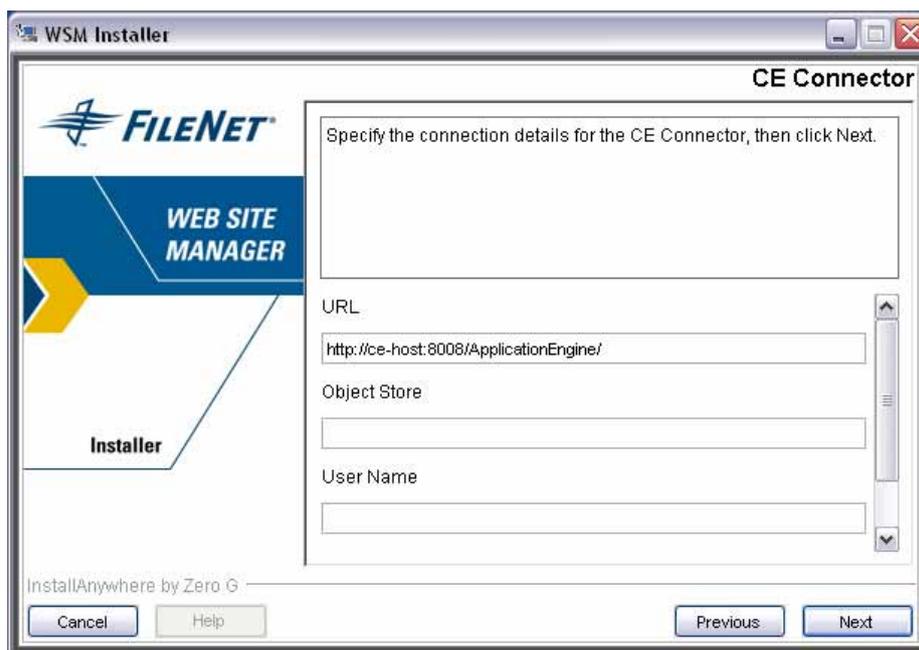
Provide the following information:

URL	The URL to the P8 repository.
Object Store	The name of the object store.
User Name	The user name with which the connector accesses the object store. The user needs read access to the P8 repository.
Password	The password for the above user name.

Note: You may have to scroll down to access the Password field.

5.2.2 Specifying the PE Connection Details

If you have chosen to enable the PE connector, you have to specify the connection details to the P8 store:
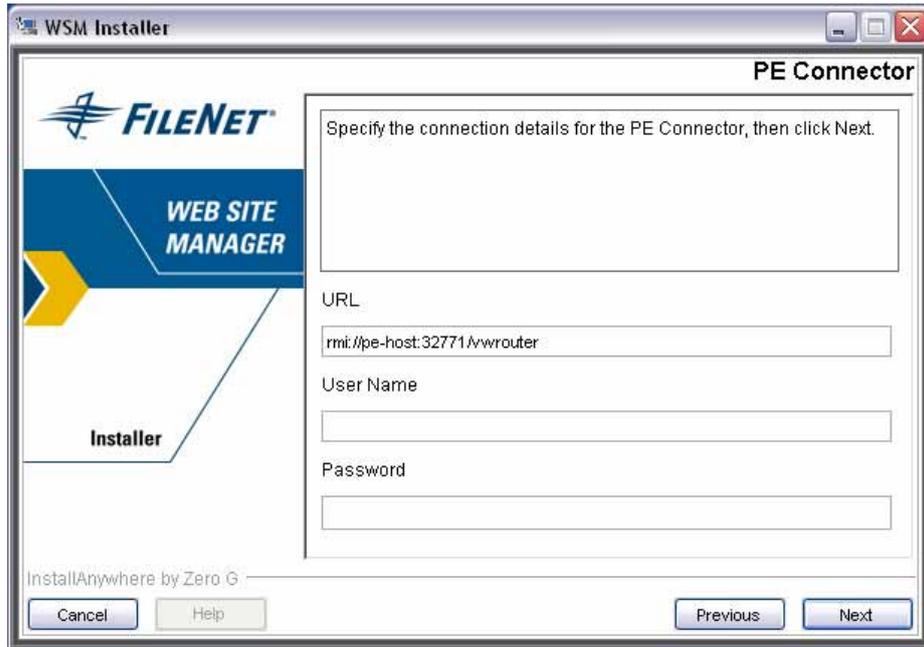
Provide the following information:

URL	The URL for the RMI connection to the process engine router.
User Name	The user name with which the connector accesses the router. The user needs read access to the P8 repository.
Password	The password for the above user name.

5.2.3 Enabling the Connectors Manually

If you have installed WSM without enabling the CE connector or the PE connector, you can enable them manually. The connectors are already installed, but not configured and not enabled.

5.2.3.1 Configuring the CE Connector

To configure the CE connector, edit the configuration file **/config/repository/ce-connector.xml**. The file looks as follows:

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>

<!DOCTYPE connector SYSTEM
"cq:/system/resources/dtd/filenet/ce-connector.dtd">

<connector>

  <content-engine
url="http://localhost:8008/ApplicationEngine">

    <objectstore name="myObjectStore" mount-
path="/etc/medialib/filenet">

      <notification username="username"

                    password="password"
```

```
                             dependency-statement="select * from
/ where Par.*.FileNetUUID = '$'" />

    </objectstore>

  </content-engine>

</connector>
```

Specify the following settings:

<content-engine url=	The URL to the P8 repository.
<objectstore name=	The name of the object store.
<notification username=	The user name with which the connector accesses the object store.
<notification password=	The password for the above user name.

5.2.3.2 Configuring the PE Connector

To configure the PE connector, edit the configuration file **/config/repository/pe-connector.xml**. The file looks as follows:

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<!DOCTYPE connector SYSTEM
"cq:/system/resources/dtd/filenet/pe-connector.dtd">

<connector>

  <process-engine url="rmi://localhost:32771/vwrouter">

    <running-workflow-searcher username="superuser"
password="superuser" query-interval="15"/>

  </process-engine>

</connector>
```

Specify the following settings:

<process-engine url=	The URL for the RMI connection to the process engine router.
<running-workflow-searcher username=	The user name with which the connector accesses the router.
<running-workflow-searcher password=	The password for the above user name.

5.2.3.3 Configuring the Security Service

When connectors are not installed with the installation program, you must configure a different security service.

Within the repository.xml file, replace the following line:

```
<service class="com.day.cq.jcr.adapter.SecurityService"
config="securityservice.xml" />
```

with:

```
<service class="com.day.cq.jcr.adapter.P8SecurityService"
config="p8-security.xml"
name="com.day.cq.jcr.adapter.SecurityService" />
```

5.2.3.4 Configuring the Script Root

In the file `/config/delivery/ecma.xml`, add the following line:

```
<scriptroot path="/apps" type="observation" glob="*.esp"
/>

<scriptroot path="/apps" type="observation" glob="*.ecma"
/>

<scriptroot path="/libs" type="observation" glob="*.esp"
/>

<scriptroot path="/libs" type="observation" glob="*.ecma"
/>

<scriptroot path="/system" type="observation"
glob="*.esp" />

<scriptroot path="/system" type="observation"
glob="*.ecma" />

<scriptroot path="/apps" type="path" glob="*" />

<scriptroot path="/libs/FileNet" type="path" />

<scriptroot path="/libs" type="path" glob="*" />
```

5.2.3.5 Enabling the Connectors

To enable the connectors, you have to register them as services in the file **/config/repository/repository.xml**. WSM comes with an example configuration file at **/config/repository/repository_filenet.xml**.

To enable the connectors, add the <service> elements from the file **/config/repository/repository_filenet.xml** to the file **/config/repository/repository.xml**.

5.2.3.6 User Rights

If you have enabled the connectors, make sure you have set the user rights as follows:

- You authenticate users using the same LDAP repository that you use for P8 document authentication.

- You place the WSM users in the user group "P8 Author".

If you use a different authentication setup, some or all P8-related functions of WSM may not work.

5.3 WSM Servlet Engine

The WSM Servlet Engine is the server within which each WSM Instance runs as a web application.

5.3.1 IP Address and Port
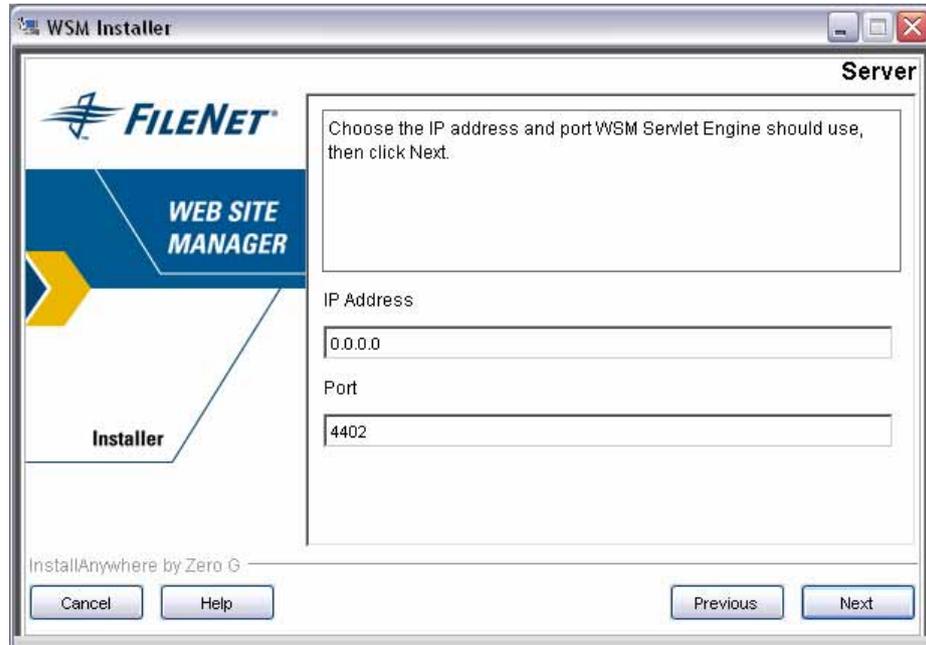
IP Address	Identifies the **IP address** that the WSM Servlet Engine uses. This is the address your browser uses to access WSM. An IP address consists of four numbers ranging from 0 to 255, separated by dots. For example, the IP address for the local computer (unreachable from other computers) is 127.0.0.1.
Port	Identifies the **Port**. The port is a number between 1 and 65535; for example, Web servers usually run on port 80.

Note: Default IP Binding. The default "fake" IP address "0.0.0.0" specifies that the Servlet Engine will listen at all addresses that it finds on the host machine. If you specify a real IP address, then the server will listen only at that particular address.

If you are unsure about which IP address or port to use, contact your system administrator.

Click **Next** to continue.

5.3.2 Name and Autostart
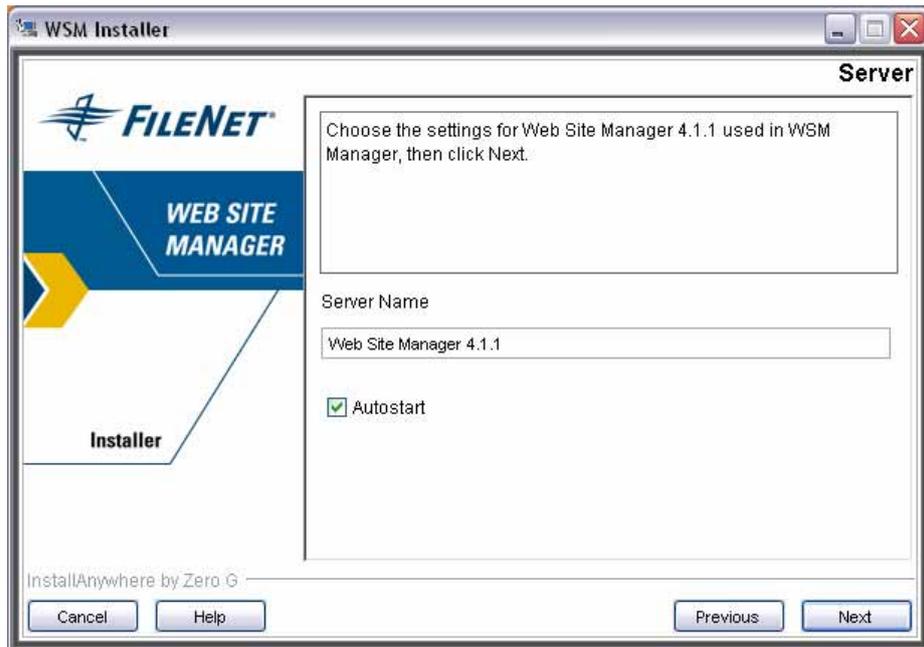
Server Name	Identifies the server name for WSM in WSM manager.
Autostart	Specifies if Servlet Engine starts whenever you open the WSM Manager. Make sure multiple autostarting servers use different port numbers to avoid conflicts.

Click **Next** to continue.

5.3.3 Choose Server Password

Password	Choose a password for the servlet engine. The default is admin.
Confirm Password	Retype the password to confirm your entry.

Click **Next** to continue.

5.4 New Instance

The screenshots below show a new Author installation. A new Publish installation is identical except that the default names and paths of the components differ.

5.4.1 Instance Name and Path

Instance Name	Choose a name for the instance. This is used on the server as a description of the Web application installed for the instance. Do not use special characters.
Context Path	Defines the address of the Web application on the server. It is unique on the server and starts with a forward slash. For example /author or /root. Do not use special characters.

Click **Next** to continue.

5.4.2 Choose Superuser ID and Password

Superuser ID	Choose an ID for the superuser. The superuser is a user with unlimited privileges on WSM. Typically, it is the top-most user in the hierarchy.
Superuser Password	Defines the password for the superuser.

Click **Next** to continue.

5.4.3 Instance Home Directory
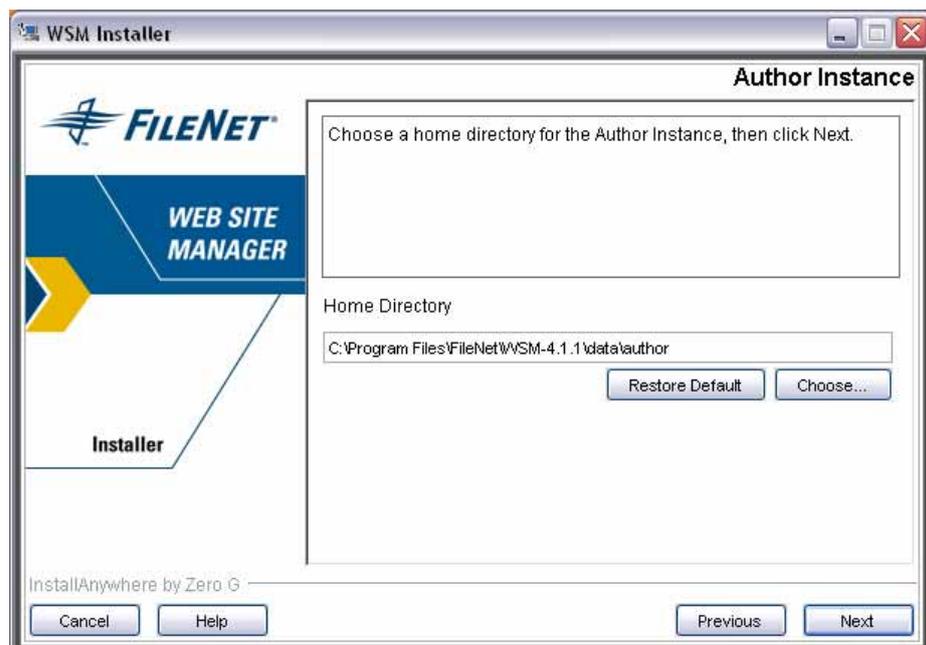
Home Directory	Defines the home directory used to store all files and directories related to the instance. Each WSM instance has its own installation directory.

Note: It is difficult to change the home directory once the installation is complete.

Click **Next** to continue.

5.4.4 Modules

Each WSM instance can be set up with a number of supplemental modules that you can choose here.
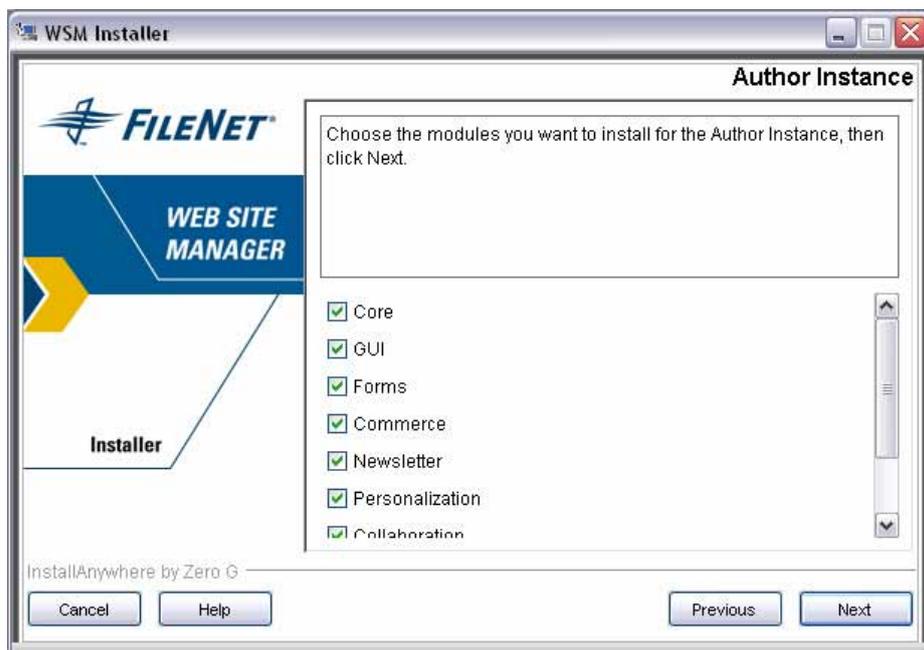
Core	Installs the WSM core system and libraries. This module is mandatory.
Skins	Installs the WSM core user interfaces. This module is mandatory.
Forms	Installs the WSM forms module.
Commerce	Installs the WSM commerce module.
Newsletter	Installs the WSM newsletter module.
Personalization	Installs the WSM personalization module.
Collaboration	Installs the WSM collaboration module
Help	Installs the WSM instance help module
Development	Enables access to the instance using the WSM development environment.

Note:	Use "Development" only for development and testing environments. Do not use it in productive, secure environments.

Click **Next** to continue.

5.4.5 Module Examples
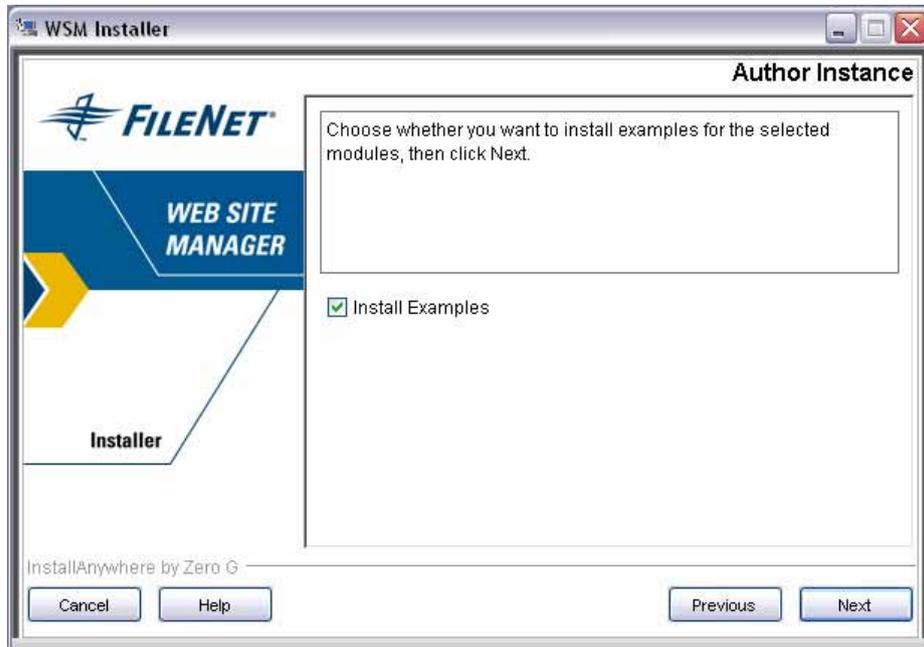
Install Examples	Installs example templates, configuration, and content for the selected modules. This is recommended for development and testing environments.

Click **Next** to continue.

5.4.6 Repository

Choose whether you want to install a new repository or use an existing one to store the content for this WSM instance.

Install new Repository	Installs the CRX repository.
Use existing Repository	Allows you to set the name and path for an existing repository.

Note: An existing repository must be CRX 1.2 or higher. If you are unsure of the type and version of your existing repository, contact your system administrator.

Click **Next** to continue.

5.4.7 Repository Name and Path

Instance name	Choose a name for the repository. This is used on the server as a description of the Web application installed for the instance. Do not use special characters.
Context path	Defines the address of the Web application on the server. It is unique on the server and starts with a forward slash. For example `/author` or `/root`. Do not use special characters.

Note: Any existing Web application at the selected context path will be removed.

Click **Next** to continue.

5.4.8 Repository Home Directory

Home Directory	Defines the home directory used to store all files and directories related to the repository of the instance.

Note: It is difficult to change the home directory once the installation is complete.

Click **Next** to continue.

5.4.9 JNDI Repository Connection
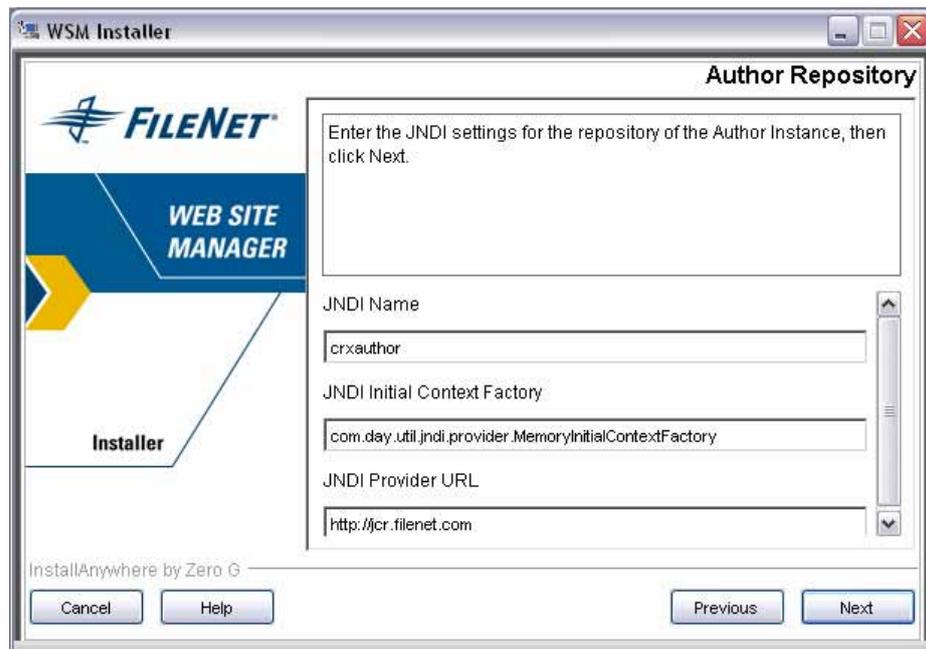
JNDI Name	Identifies the instance repository through JNDI. The JNDI name must be unique and the repository must run the same Java VM.
JNDI Initial Context Factory	Contains the class name of the initial context factory.
JNDI Provider URL	Contains the name of the environment property for specifying configuration information for the service provider to use.

Note: If you are unsure of the JNDI settings, keep the defaults or contact your system administrator.

Click **Next** to continue.

5.4.10 User Name and Password

User ID	Identifies a user with unlimited privileges on the repository. Typically, it is a member of the administrator group.
Password	Choose a password for the user.

Note: These credentials are stored in the instance Web application.

Click **Next** to continue.

5.4.11 Instance Summary

This window summarizes the installation options for this WSM instance.

Click **Next** to continue.

5.4.12 Deploying on a Third-Party Application Server

If you are installing this instance on a third-party application server, the installer bundles up the instance in a **war** file and asks you where you want to place the file. Once you get the file, deploy it in your third-party application server before proceeding with the installation process. For more information, see the **Installing WSM on an Application Server** section.

5.5 Installation Directory
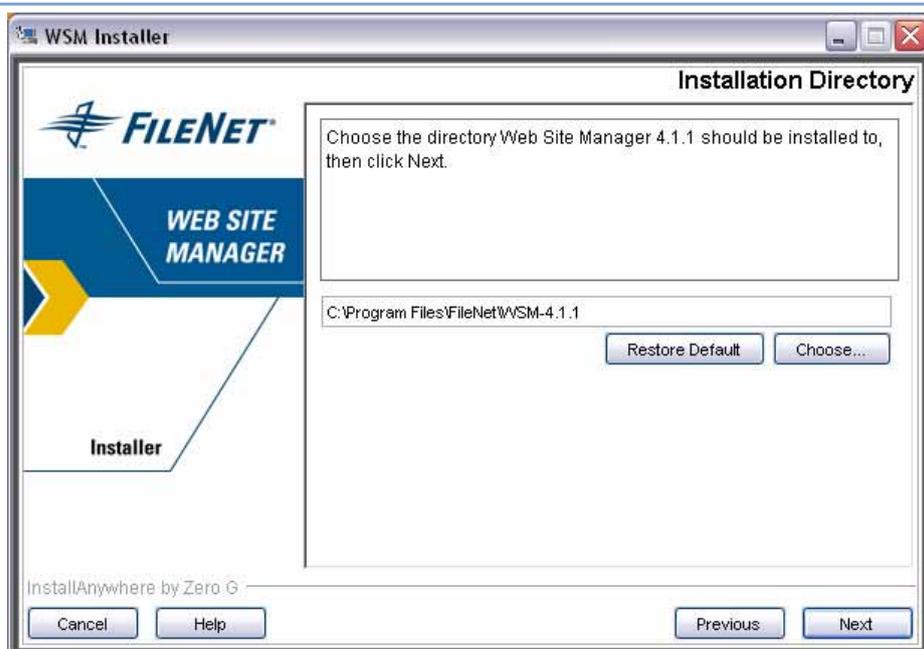
Specify the path of the folder into which you want to install WSM. The installation directory must be on the same computer that is

running the installation. It is used to store WSM features, information necessary for uninstalling, and the installation log.

Click **Next** to continue.

5.6 Java VM

WSM is a Java application and requires a Java Virtual Machine to run. You can either install a new Virtual Machine or use an existing one.

Install Java VM 1.4.2_06	Installs a new, dedicated Java VM for this WSM installation. If you want to use an existing Java VM, clear the check box. In the next step, the installer asks you for the path to the virtual machine you want to use.

Note: On Unix, you have to use an existing Virtual Machine.

Click **Next** to continue.

6 Installing WSM on an Application Server

This section tells you how to install WSM on an application server. Consult the notes section to see which application servers are supported. Consult the application server documentation for more information on installing Web applications.

6.1 Starting the Application Server Installer

The WSM installer helps you to install WSM on an application server. In the custom installer, uncheck **WSM Servlet Engine** to start the application server installation process.
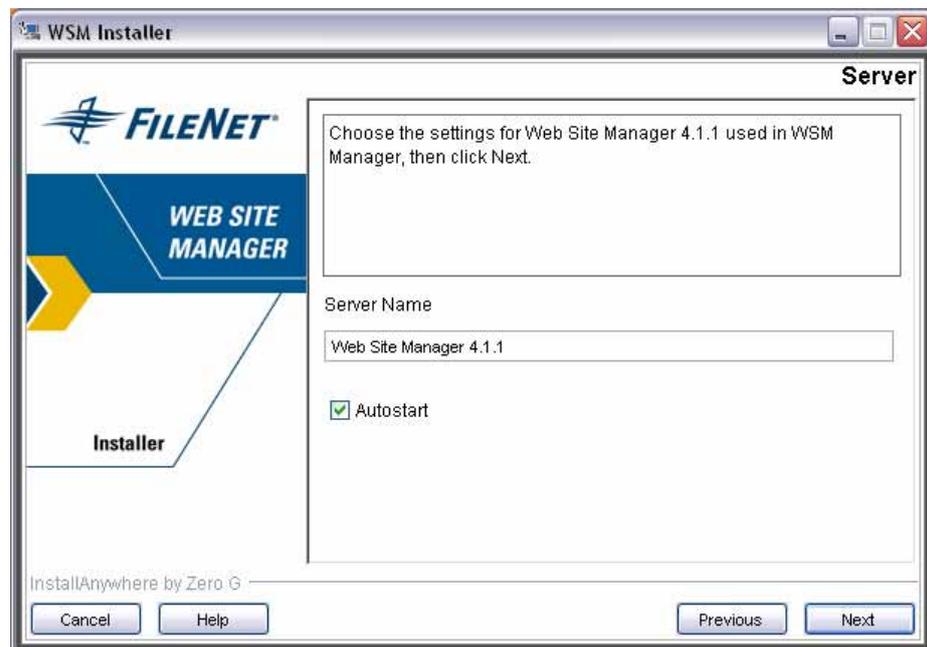
6.2 Specifying the Application Server
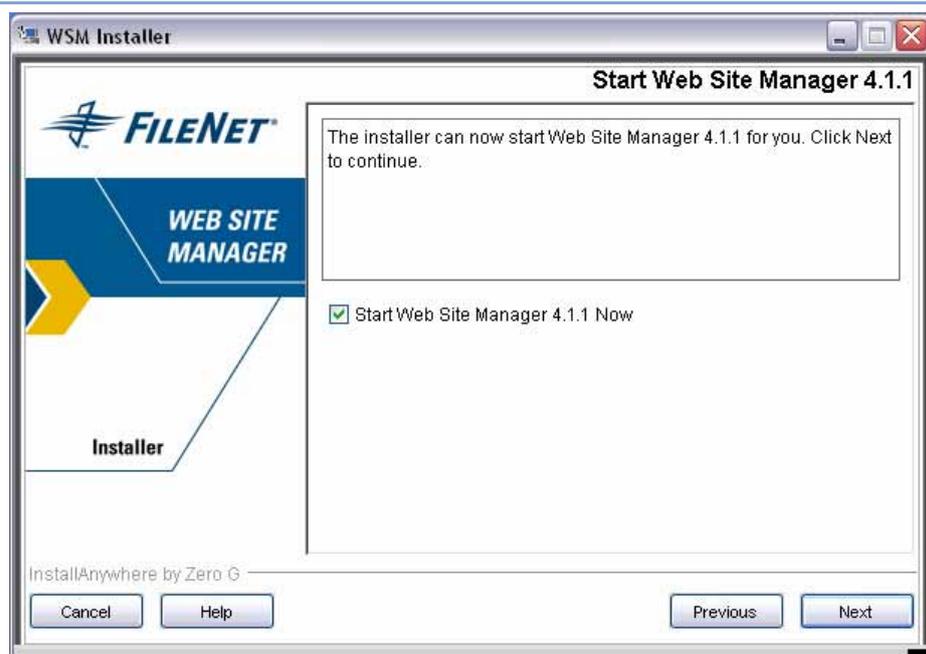
In the next step, you need to specify a running application server or servlet engine. The installer then guides you through the necessary steps of setting up WSM for the application server.

Note: Setting up WSM on an application server requires manual changes on the application server. Ensure that you read the instructions in the installer carefully before you click Next. Also, make sure that you type all paths correctly, as they may be difficult to change later.

6.3 Installing the WSM Libraries

Before you can install the WSM Web applications, you need to install the WSM libraries on the application server. You can find the libraries in the folder **/server/lib** of your WSM program folder. Add the following .jar files (Java archive files) to the libraries of your application server:

- All files that begin with **crx**, such as crx-api-1.0.1.jar and crx-jndi-1.0.1.jar

- All files that begin with **jcr**, such as jcr-1.0.jar and jcr-rmi-1.0.jar

You can typically find these files in the folder /server/lib/container. The number and names of these files may change between WSM versions.

Note: You may have to restart the application server for the changes to take effect. If so, do this before you install the Web applications in the next step.

6.4 Installing the WSM Web Applications

The WSM Web applications are stored in the following Web application archive files (.war files).

- **crxauthor.war**: The repository of the authoring environment

- **crxpublish.war**: The repository of the publishing environment

- **author.war**: The WSM authoring environment

- **publish.war**: The WSM publishing environment

- **ROOT.war**: The WSM welcome screen; you do not need this file for WSM to run, and it may not be available in custom installations of WSM

The files are stored in the folder **/server/webapps** of your WSM program folder. Install the files on the application server in the above order. The repository files must be installed when you install the other files.

After you have installed the Web applications, **restart the Application Server**. If you do not do this, WSM may throw an internal server error when you try to use it.

6.5 Troubleshooting

Due to differences between application servers, you may receive the following error message when you try to start WSM:

```
Repository not found at startup
```

You can usually fix this by specifying the JNDI configuration in the Web application configuration file. The entries are already in the files but are commented out. To use the entries, proceed as follows:

1. In the archive **crxauthor.war**, open the file **WEB-INF/web.xml** for editing (see below for how to do this).

2. In the section **REPOSITORY SERVLET**, remove the comment marks around the JNDI configuration.

3. In the section **JCR EXPLORER SERVLET**, remove the comment marks around the JNDI configuration.

4. Save the changes back into the archive file (see below).

5. Repeat the process for the archive **crxpublish.war**.

Note: On Unix, you can extract a file from a .jar repository as follows: jar -xvf xrxauthor.war WEB-INF/web.xml. After modifying the file, you can upload it again using the following command: jar -uvf xrxauthor.war WEB-INF/web.xml. On Windows, you can use a tool such as WinRAR to edit files in a .jar archive file.

7 WSM LDAP Setup

If you want to access a FileNet P8 repository with WSM, you need to use LDAP authentication. Refer to the WSM configuration guide for details on how to set it up.

To give authors access to the FileNet P8 functions in WSM, you need to place the authors in the group **P8 Authors**. Use the following configuration in the file /config/repository/authenticators/ldapdefaultfinder.xml (the configuration may vary depending on the structure of your LDAP repository):

```
<autocreate csd="ldapuser">
  <setatom name="dn" attribute="distinguishedName" />
  <syncatom name="UserID" attribute="cn" />
  <syncatom name="EMail" attribute="mail" />
  <syncatom name="Fullname" attribute="displayName" />
  <syncatom name="Language" attribute="preferredLanguage"
/>
  <addgroup handle="/access/groups/author" />
  <addgroup handle="/access/groups/p8author" />
  <addacl type="allow" glob="/content" rights="rwxcd" />
</autocreate>
```

8 Starting WSM

After you have successfully installed WSM, you can start it in a number of ways.

8.1 After Installation (Windows)

Start WSM 4.1 Now	In Windows, you can choose to start WSM immediately by selecting this checkbox. This will cause WSM Manager to start the installed WSM instances that you can then connect to with your browser.

8.2 With the Manager (Windows)

To start WSM using the Manager application, proceed as follows:

1. In the Start menu, click **All Programs**, click the **WSM** program group and then click **Start WSM**. The manager opens.

2. Click the WSM instance you have installed and then click the **Start** button.

8.3 With the Start Script (Unix)

In a Unix environment, run the script <cq-install>/server/start. To stop WSM, use the script <cq-install>/server/stop).

9 Notes

This section contains hints and notes about the installation process.

9.1 Libraries

WSM uses the following external libraries in the folder /server/runtime/0/_crxauthor/WEB-INF/lib and /server/runtime/0/_crxpublish/WEB-INF/lib of the WSM installation folder:

- crx2filenet-1.1.jar

- activation-1.0.2.jar

- mail-1.3.2.jar

- javaapi-3.5.1.jar

- p8cjares-3.5.1.jar

- filenet-soap-3.5.1.jar

- jaxen-1.1-beta-6.jar

The following libraries are in the folder /data/author/bin/lib and data/publish/bin/lib of the WSM installation folder:

- cq-filenet-4.1.jar
- pe-3.5.1.jar

The following file is in the folder /server/lib/common of the WSM installation folder:

- xercesImpl-2.6.2.jar

9.2 Installation Logs

In case of a successful installation process, the installer logs do not contain errors or warnings.