HP OpenView Administrator Workstation Using Radia

Radia Publisher Guide

Version: 4.0i

for the Windows® operating system

Manufacturing Part Number: T3420-90012

July 2005

Legal Notices

Warranty

Hewlett-Packard makes no warranty of any kind with regard to this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

A copy of the specific warranty terms applicable to your Hewlett-Packard product can be obtained from your local Sales and Service Office.

Restricted Rights Legend

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

Hewlett-Packard Company
United States of America

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

Copyright Notices

© Copyright 1998-2004 Hewlett-Packard Development Company, L.P.

No part of this document may be copied, reproduced, or translated into another language without the prior written consent of Hewlett-Packard Company. The information contained in this material is subject to change without notice.

Trademark Notices

Linux is a registered trademark of Linus Torvalds.

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

Acknowledgements

PREBOOT EXECUTION ENVIRONMENT (PXE) SERVER
Copyright © 1996-1999 Intel Corporation.

TFTP SERVER
Copyright © 1983, 1993
The Regents of the University of California.

Technical Support

Please select Support & Services from the following web site:

<http://www.hp.com/managementsoftware/services>

There you will find contact information and details about the products, services, and support that HP OpenView offers.

The support site includes:

- Downloadable documentation
- Troubleshooting information
- Patches and updates
- Problem reporting
- Training information
- Support program information

About this Guide

Who this Guide is for

The Radia Publisher assists everyone in your organization responsible for publishing Windows Installer based applications as well as Operating System images to the Radia Database.

What this Guide is about

The Radia Publisher offers a streamlined publishing interface implementing a process of minimal steps and configuration to ready applications for distribution into your environment. The *Radia Publisher Guide* introduces each aspect and steps you through each process involved with publishing Windows Installer applications and Operating System images.

Conventions

You should be aware of the following conventions used in this book.

Table P.1 ~ Styles		
Element	**Style**	**Example**
References	*Italic*	See *About the Radia Publisher* section in this book.
Dialog boxes and windows	**Bold**	The **Radia System Explorer Security Information** dialog box opens.
Code	Andale Mono	radia_am.exe
Selections	**Bold**	Open the **\Admin** directory on the installation CD-ROM.

Table P.2 ~ Usage		
Element	**Style**	**Example**
Drives (system, mapped, CD)	Italicized placeholder	*SystemDrive*:\Program Files\Novadigm might refer to C:\Program Files\Novadigm on your computer. *CDDrive*:\client\radia_am.exe might refer to D:\client\radia_am.exe on your computer.
Files (in the Radia Database)	All uppercase	PRIMARY
Domains (in the Radia Database)	All uppercase	PRIMARY.SOFTWARE May also be referred to as the SOFTWARE domain in the PRIMARY file.
Classes (in the Radia Database)	All uppercase	PRIMARY.SOFTWARE.ZSERVICE May also be referred to as the ZSERVICE class in the SOFTWARE domain in the PRIMARY file.

The table below describes terms that may be used interchangeably throughout this book.

Table P.3 ~ Terminology*	
	* Depends on the context. May not always be able to substitute.
Term	**May also be called**
Application	software, service
Client	Radia Application Manager and/or Radia Software Manager
Computer	workstation, server
NOVADIGM domain	PRDMAINT domain
	Note: As of the 4.0 release of the database, the NOVADIGM domain is being renamed the PRDMAINT domain. Therefore, if you are using an earlier version, you will see the NOVADIGM domain in the database.
Radia Configuration Server	Manager, Active Component Server
Radia Database	Radia Configuration Server Database

Contents

1

Introduction

At the end of this chapter, you will:

- Be familiar with the Radia Publisher.
- Understand the process of publishing to the Radia Database.
- Understand the different types of applications and files that can be published using the Radia Publisher.

The Radia Publisher uses a streamlined process for publishing applications and files to your Radia Database. Currently, the Radia Publisher supports the publishing of Windows Installer applications and operating system images.
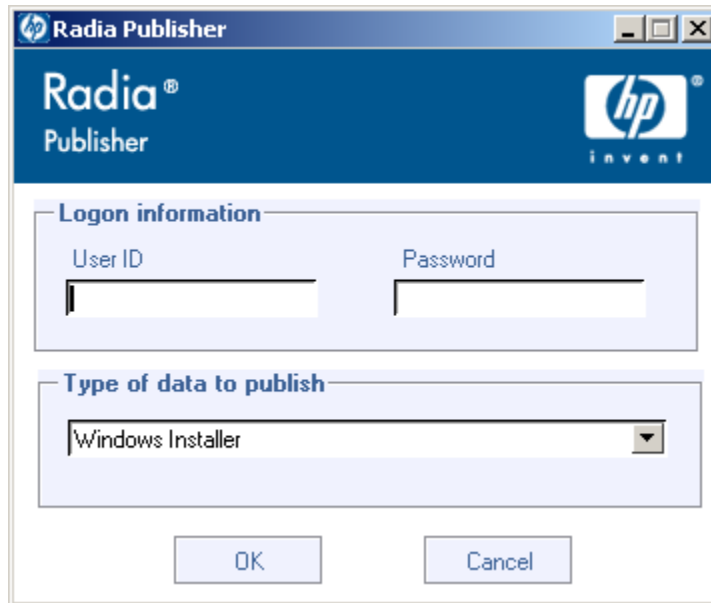

Figure 1.1 ~ The Radia Publisher login window.

About Packaging and Publishing with Radia

Packaging is the process of identifying resources, editing those resources' installation attributes, defining how they are to be installed, and saving the resources and installation instructions in a machine-readable file format. A package typically contains one or more files and configuration settings.

Publishing is the process of importing a package and its imbedded information to the Radia Database (AKA the Radia Configuration Server Database). A package must be published before its content can be distributed and deployed into your environment.

Radia offers many different packaging and publishing options through multiple tools. Make note of the different types of resource packages you will create and publish, then choose your Radia packaging and publishing products as required.

- If you're **packaging** Windows Installer files, use the **Radia Packager for Windows Installer**. (See the *Radia Extensions for Windows Installer Guide*).

- If you're **publishing** Windows Installer files, use the **Radia Publisher**. (See the *Radia Publisher Guide)*.

- To **package** and **publish** all other types of files and applications, use the **Radia Packager**. (See the *Radia Application Manager Guide* or the *Radia Software Manager Guide*).
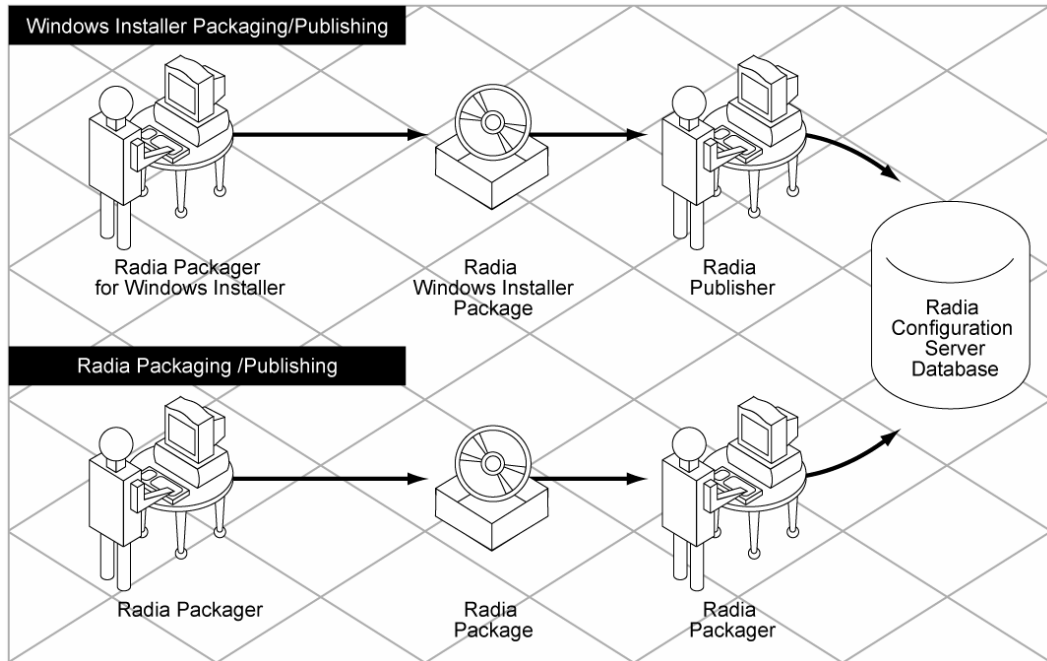
Figure 1.2 ~ Radia Packaging/Publishing options.

About the Radia Publisher

The Radia Publisher is an administrative tool that facilitates publishing Windows Installer applications and operating system images to your Radia Database. In only four steps, you can have your files ready for deployment to your environment.

Install the Radia Publisher to the computer where your administrator will have access to the necessary files for publishing.
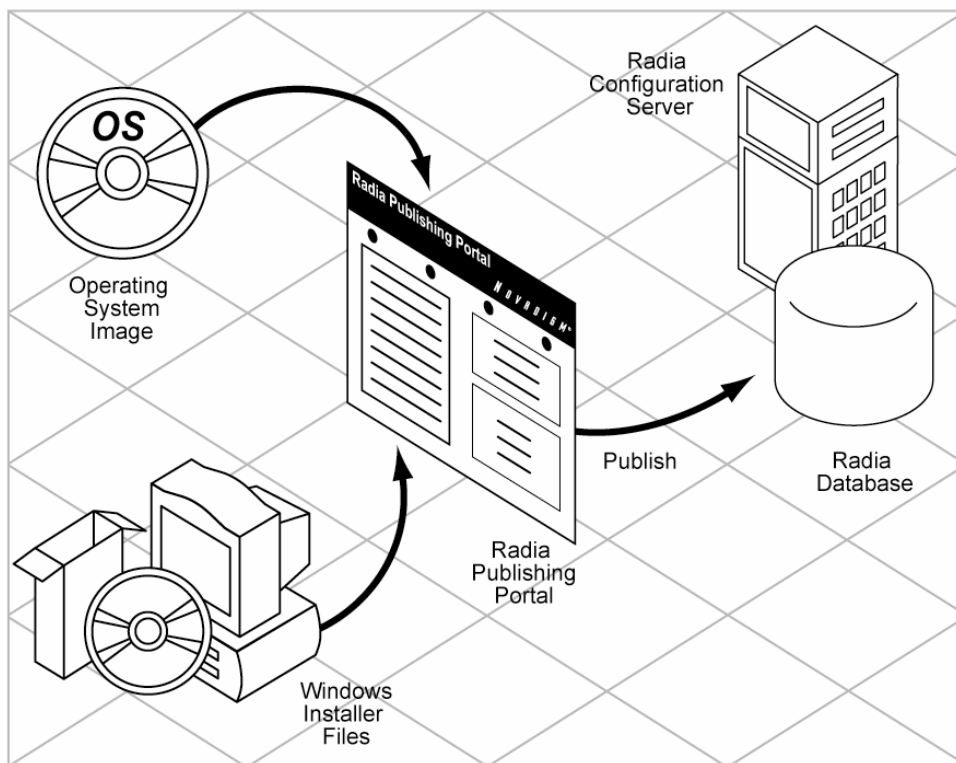

Figure 1.3 ~ Radia Publisher publishing to the Radia Database.

Publishing with the Radia Publisher

The Radia Publisher consists of four main steps for publishing files to the Radia Database. These are:

1. **Select** – Select the file or image to be published to the Radia Database.
2. **Edit** – Create the Administrative Installation Point, simulate the application's user interface, edit MSI features and properties, select any transforms, and enable Machine/User parameters (Windows Installer files only).
3. **Configure** – Provide the information for creating a Radia package or service that will be published to the database.
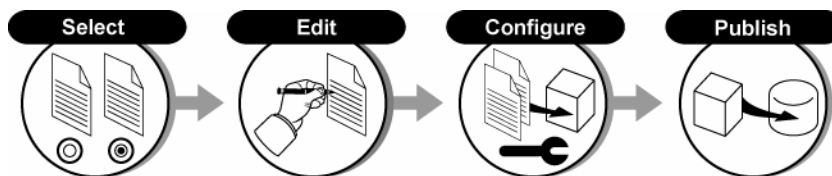4. **Publish** – Publish to the Radia Database.

Figure 1.4 ~ Publishing steps.

During each step you will select, edit, and configure the package to be published by providing the information requested at each window. *Chapter 3: Publishing Files with the Radia Publisher* starting on page 29 guides you through this process providing detailed explanations of each step in the publishing process.

At any point in the publishing process, you can see where you are by viewing the numbered steps at the top of the user interface. The current step is highlighted.
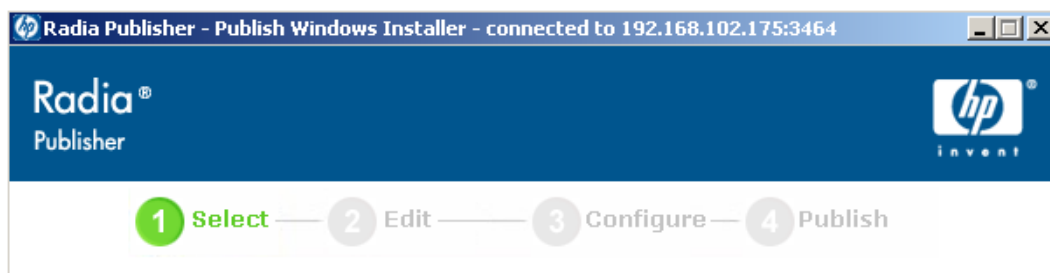

Figure 1.5 ~ Current publishing step is highlighted in the user interface.

Starting the Radia Publisher

The Radia Publisher should be run from the administrative computer where it was installed.

To start the Radia Publisher

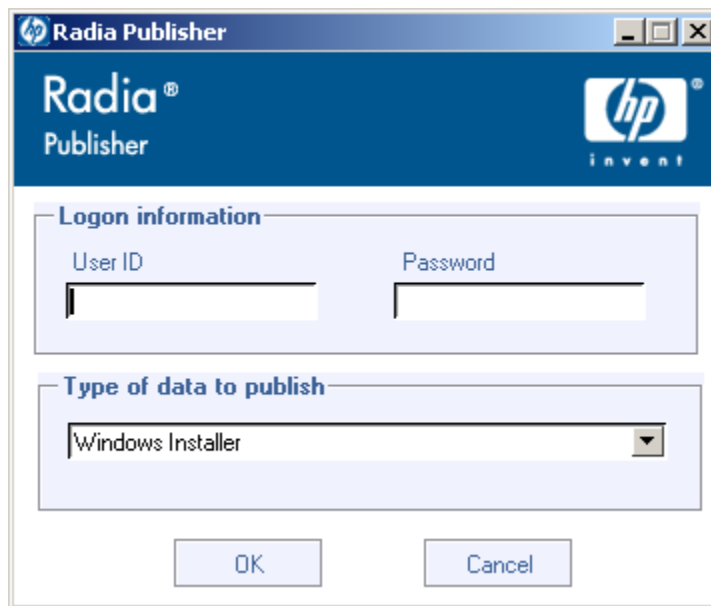1. Double-click the Radia Publisher icon on your desktop.

Figure 1.6 ~ Radia Publisher logon window.

2. At the Radia Publisher logon window, enter your Radia administrator password and use the **Type of data to publish** drop-down list to select the type of file to publish. *Chapter 2: Installing the Radia Administrator Workstation* starting on page 19 explains in detail each step involved in publishing files to the Radia Database.

Summary

- Use the Radia Publisher to publish Windows Installer files and operating system images to the Radia Database.

- The Radia Publisher consists of four main steps: Select, Edit, Configure, and Publish.

- Start the Radia Publisher using the icon installed on your desktop.

2

Installing the Radia Administrator Workstation

At the end of this chapter, you will:

■ Understand the requirements for installing the Radia Administrator Workstation.

■ Be able to install the Radia Administrator Workstation.

The Radia Administrator Workstation

The Radia Publisher is one of the features available with the Radia Administrator Workstation suite of applications. This software is included on your Radia Infrastructure CD-ROM within the management_infrastructure directory.

The Radia Administrator Workstation includes the following features:

- Radia Packager
- Radia System Explorer
- Radia Client Explorer
- Radia Screen Painter
- Radia Publisher

When installing the Radia Administrator workstation, you have the option to install all Radia Administrator Workstation features or only those you require.

All Radia Administrator Workstation features are installed by default.

System Requirements

Radia Requirements

- Radia Database version 3.1 or later (Radia Configuration Server 4.5.1 or later). When publishing .msi files, two classes are required: MSIPROPS and MSIBASIC, both of which are available in this database version.
- The Radia Administrator Workstation must be installed prior to installing the Radia Publisher. The Radia Publisher uses shared components defined during the Radia Administrator installation to connect to your Radia Database.

Windows System Requirements

- Windows 9x, NT, 2000, Windows 2003 Server, and XP

Installing the Radia Administrator Workstation

To install the Radia Administrator Workstation

1. Insert your Radia Infrastructure CD-ROM and navigate to the appropriate operating system specific sub-directory within the administrator_workstation directory.

For example, to install the Windows version of the Radia Administrator Workstation, navigate to: \management_infrastructure\administrator_workstation\win32\

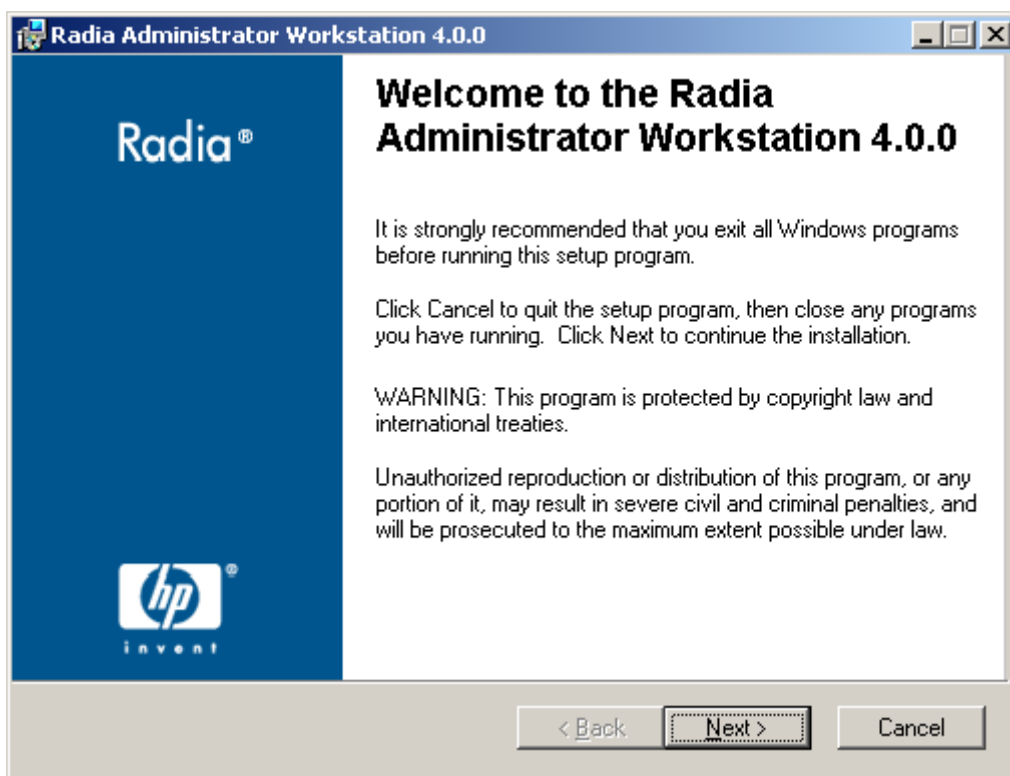**2.** Double-click **Setup.exe** to start the installation program.

Figure 2.1 ~ Radia Administrator Workstation welcome window.
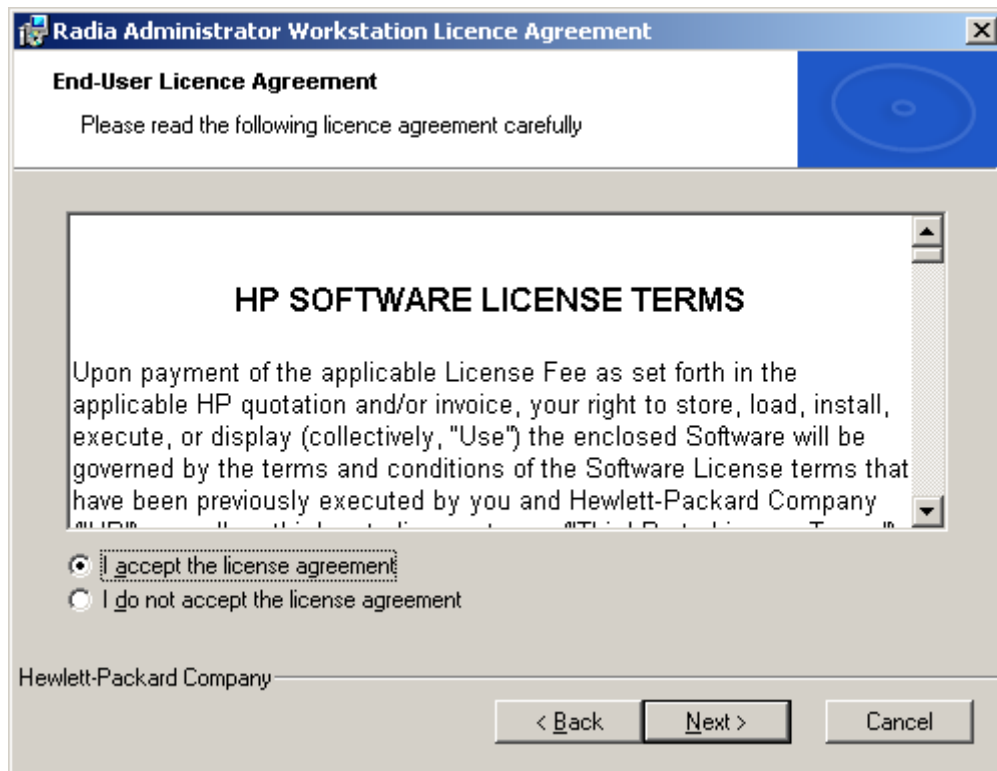
3. Click **Next**.

Figure 2.2 ~ HP Software License Terms window.

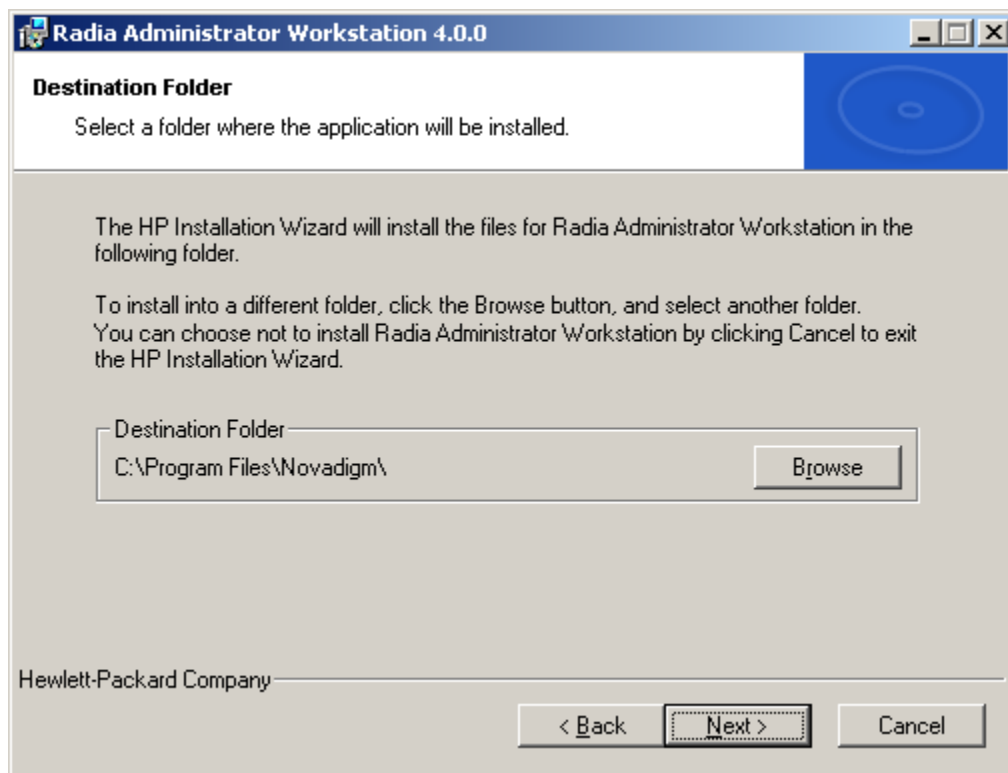**4.** Read and accept the HP Software License Terms and click **Next.**

Figure 2.3 ~ Select the directory in which to install the Radia Administrator Workstation.

5. The Radia Administrator Workstation is installed to C:\Program Files\Novadigm, by default. To install the Radia Administrator Workstation to another directory, click **Browse** and manually select the new directory.
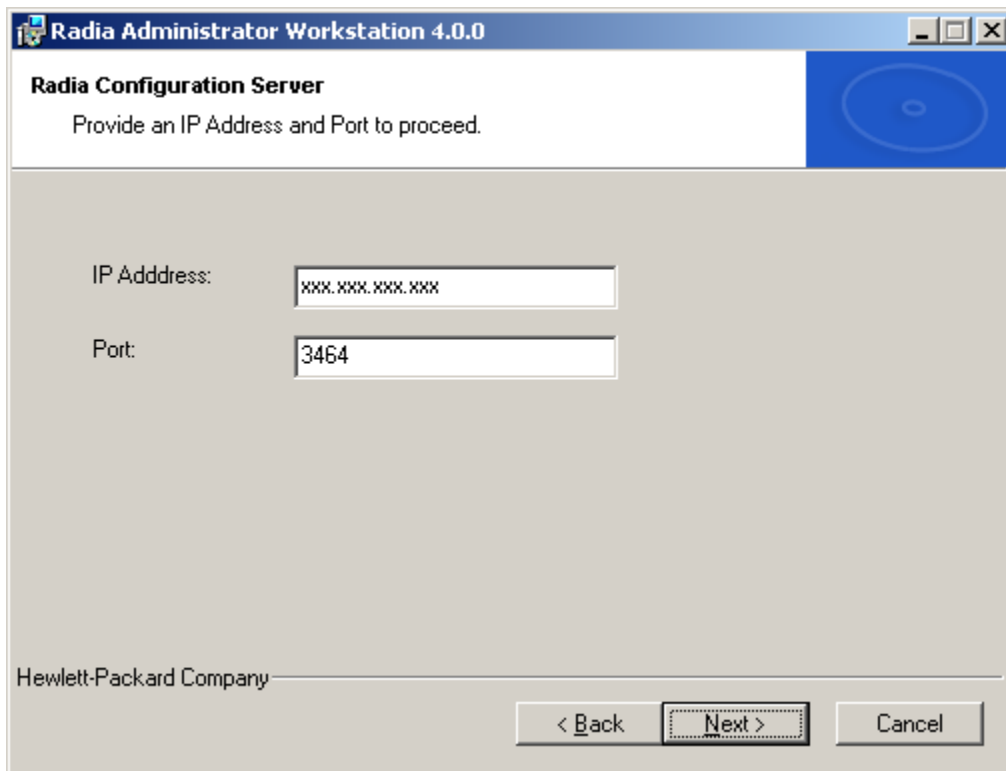
6. Click **Next**.

Figure 2.4 ~ Identify the Radia Configuration Server.

7. Type the IP address or Hostname, and port number of your Radia Configuration Server.
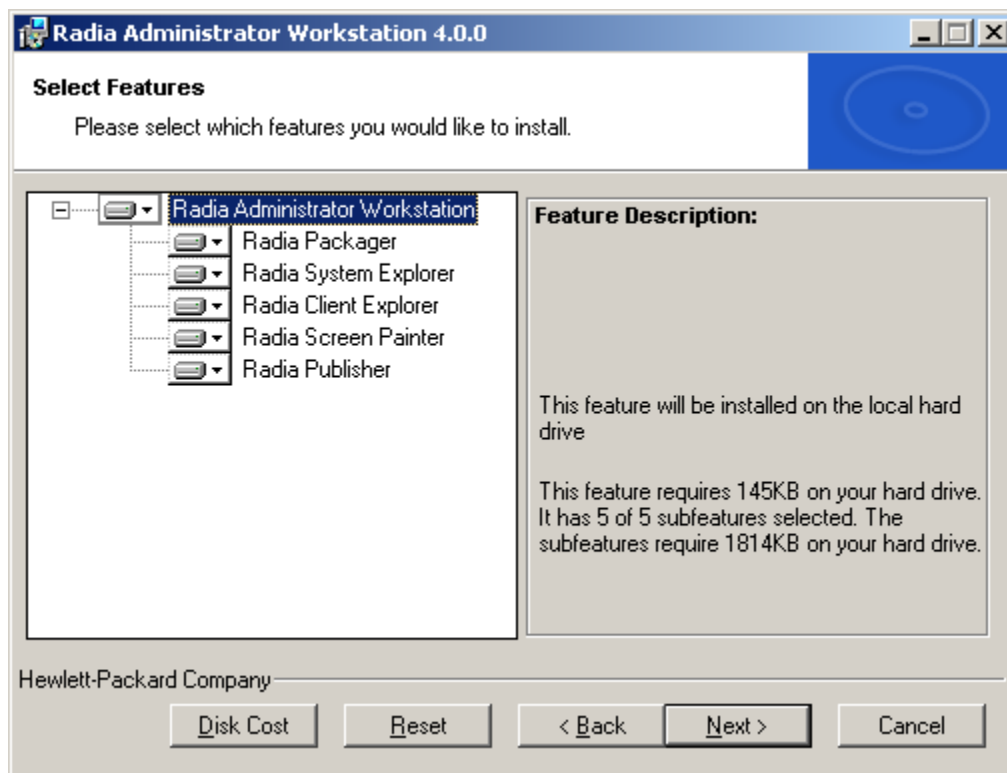
8. Click **Next**.

Figure 2.5 ~ Select Radia Administrator Workstation features.

9. Select which Radia Administrator Workstation features you would like to install. All features are installed by default.

To avoid installing an individual feature, click the down arrow to the left of the feature name and select **Entire Feature will be unavailable.**

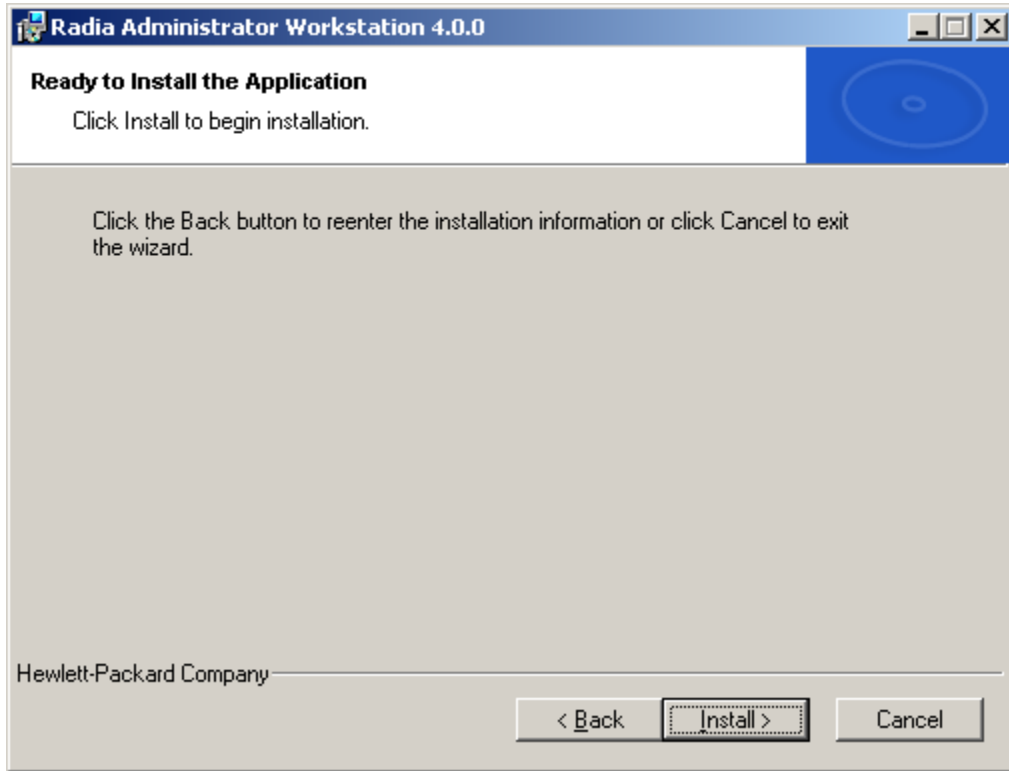**10.** When you have selected the Radia Administrator features to install, click **Next**.

Figure 2.6 ~ Ready to install window.

11. Click **Install** to begin the installation process.

When the installation is finished, click **Finish**.

Summary

- Radia Database version 3.1 or later is required to publish .msi files.
- Before you install the Radia Administrator Workstation, have your Radia Configuration Server IP address or Hostname, and port number ready.
- Install all or only some of the Radia Administrator Workstation features.

Publishing Files with the Radia Publisher

At the end of this chapter, you will:

■ Be familiar with the different types of files you can publish with the Radia Publisher.

■ Understand the processes involved for publishing the different file types.

Overview

The Radia Publisher supports the publication of Windows Installer files and operating system images to your Radia Database. The following sections explain the steps required for publishing each file type.

Additionally, the Radia Publisher allows for the publication of files in "batch mode." This lets you publish multiple .msi files to the Radia Database during one publishing session using a configuration file. Refer to the section *Publishing Multiple .msi Files (Batch Mode)*, on page 54 for more information.

Publishing Windows Installer Files

Windows Installer uses MSI files to distribute software packages to your operating system. The Radia Publisher uses the files to create a Radia package and service that are then published to the Radia Database. Once contained in the Radia Database, the software packages are ready for distribution to your environment.

To publish Windows Installer files

1. Start the Radia Publisher by double-clicking its icon on your desktop.
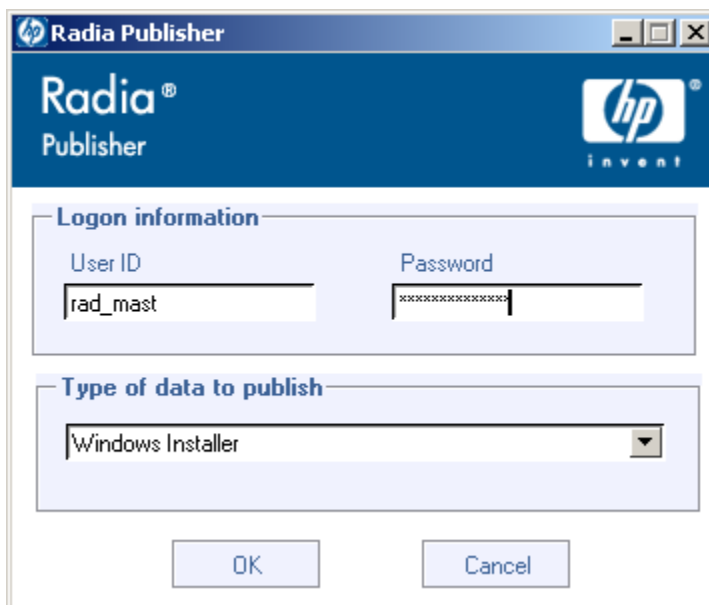

Figure 3.1 ~ Login information.

2. In the **User ID** field, type your Radia administrator user ID.

3. In the **Password** field, type your Radia administrator password.

4. From the **Type of data to publish** drop-down list, select **Windows Installer**.
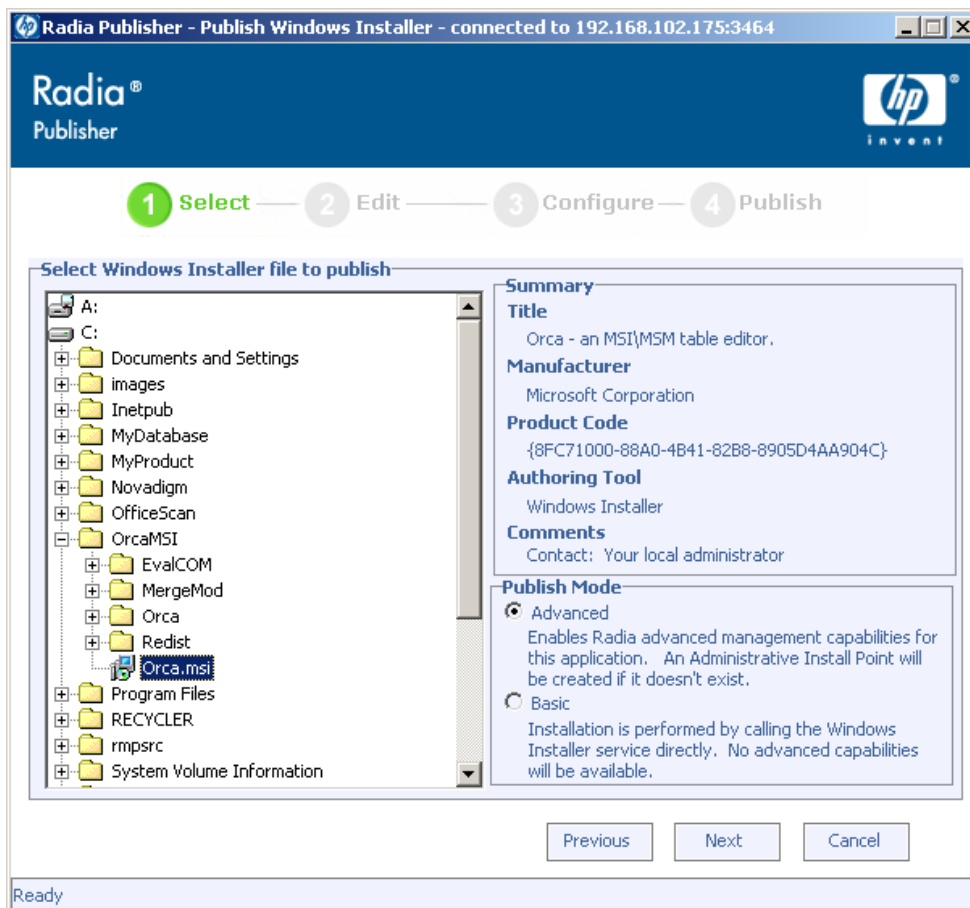
5. Click **OK**.

Figure 3.2 ~ Select the Windows Installer file to publish.

6. Use the Explorer window to find the Windows Installer file. The right-hand pane will display any information that is available for the MSI file you select.

7. In the **Publish Mode** section, select either **Basic** or **Advanced**.

If you select the **Advanced** publishing mode, you will be required to create an Administrative Installation Point in the next step as well as enable advanced management capabilities, including file feature and property management.

8. Click **Next**.

Publishing Modes: Advanced and Basic

After you select a file, and depending on which publishing mode you select, you will have different options available when you click **Next** in the **Select** window.

Advanced Publishing Mode

The advanced publishing mode offers seven options. Click on any link to configure that aspect of the advanced publishing mode.
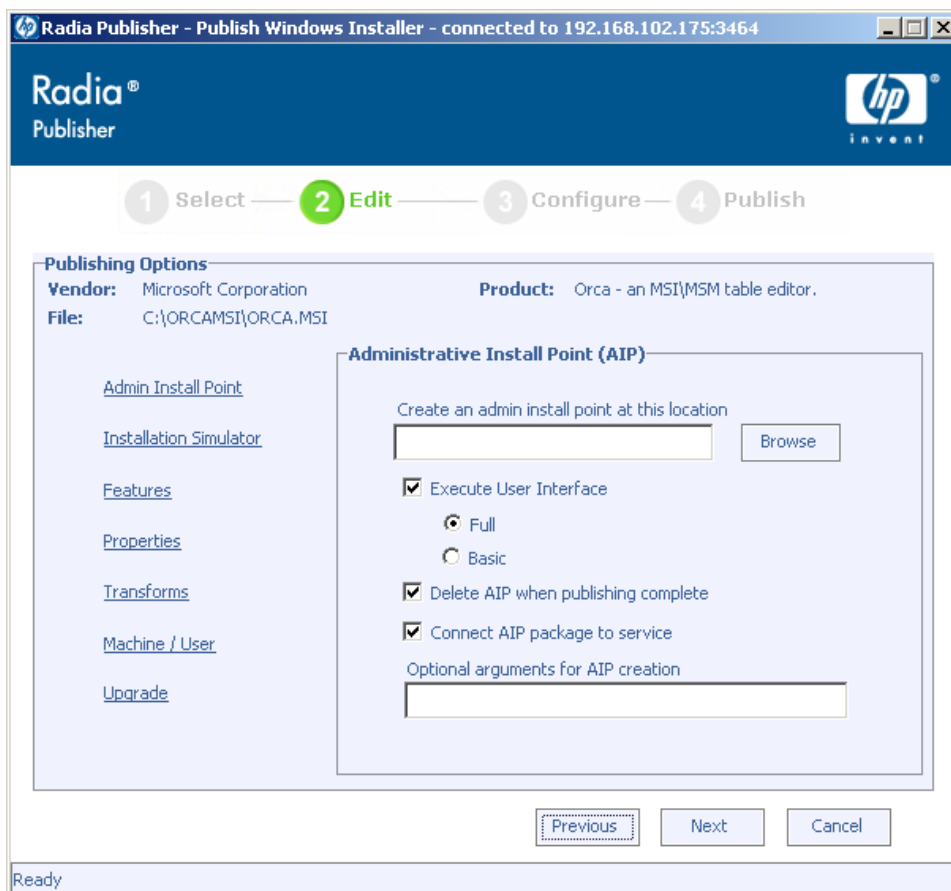
Figure 3.3 ~ Advanced Publishing Mode options.

■ **Admin Install Point**
Create an administrative installation point (AIP) by filling in the required fields.

> **Note**
>
> In order to connect the AIP package to the service, select the check box **Connect AIP package to service**. When this checkbox is selected, the AIP package will be connected to the service and will have its ZSTOP000 set to 1. This prevents the AIP package from being delivered to the desktop, but its connection to the service will guarantee its travel during Radia Distributed Configuration Server operations.

- **Installation Simulator**
  Run the application installation program.

- **Features**
  Configure the available MSI features.

- **Properties**
  Configure any of the available MSI properties.

- **Transforms**
  Reorder the application of any transform files associated with the Windows Installer file.

- **Machine / User**
  Define Radia Machine/User parameters.

- **Upgrade**
  Displays existing MSI packages within the Radia Database that are available for upgrade using the current Windows Installer package.

Basic Publishing Mode

The basic publishing mode offers four options. Click on any link to configure that aspect of the basic publishing mode.
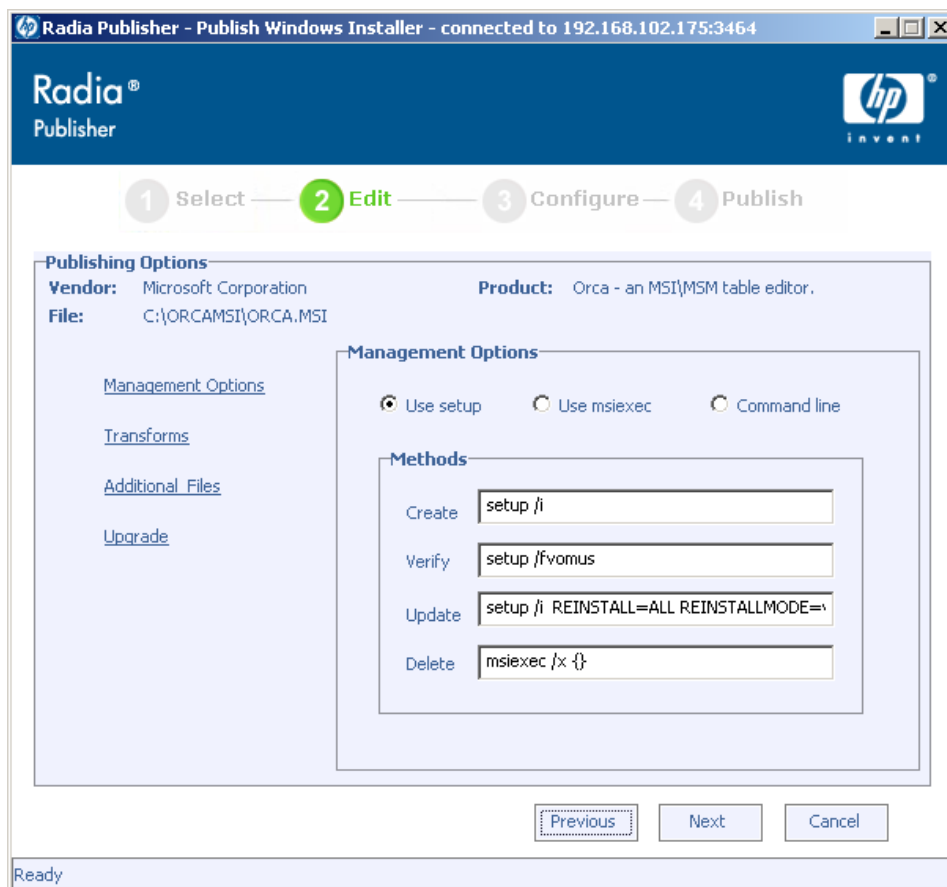

Figure 3.4 ~ Basic Publishing Mode options.

■ **Management Options**
Select which management option you would like to apply. The Management Option you select defines how the application will run once it is deployed.

  ● **Use setup**
  ● **Use msiexec**
  ● **Command line**

■ **Transforms**
Reorder the application of any transform files associated with the Windows Installer file.

■ **Additional Files**
Apply any available additional files.

■ **Upgrade**
Displays existing MSI packages within the Radia Database that are available for upgrade using the current Windows Installer package.

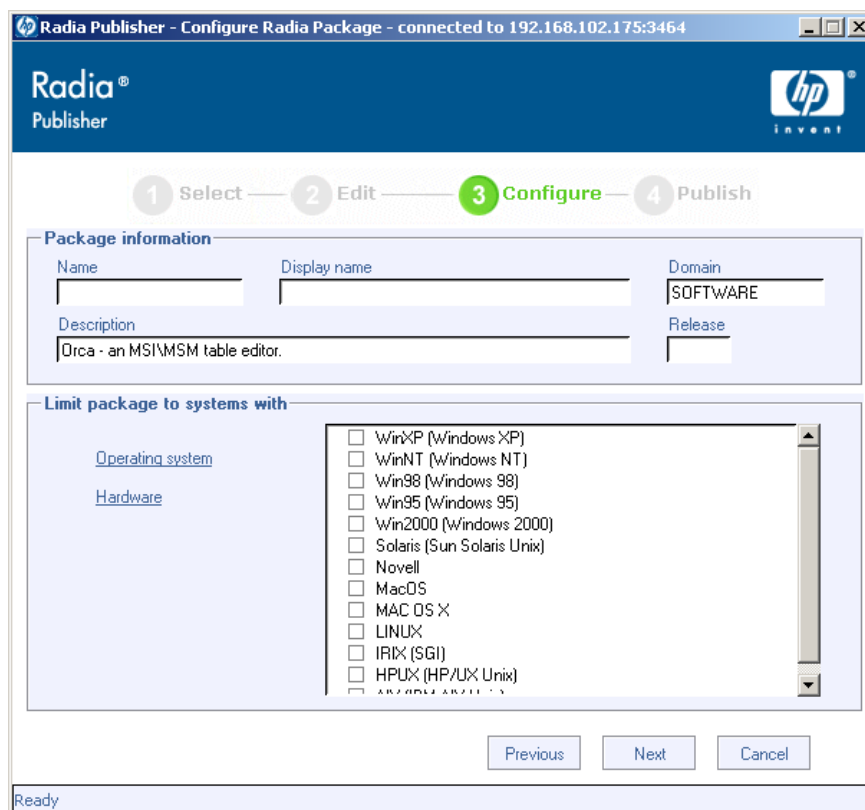When finished editing your publishing options, click **Next**.

Figure 3.5 ~ Package information window.

1. Use the **Package Information** section to enter the Radia package information.

2. Use the **Limit package to systems with** section to limit the package to any specific operating system or hardware. Click on any link to display the configurable options.

3. Click **Next**.

Figure 3.6 ~ Service Information window.

4. Select whether you want to create a new service (**Create new**), use an existing service (**Use existing**), or skip creating a service (**No service**) at this time.

> **Note**
>
> If you want to create a package only, select **No service**. This is useful if, for example, you have a single service, but want to create multiple packages and later connect them to the existing service using the Radia System Explorer.

5. Enter the appropriate information in the service description text boxes. If you are using an existing service, make sure you enter the service name correctly.

6. In the **Assignment type** section, select whether the service is mandatory or optional. By default, **Mandatory** is selected, which will distribute this service to all available subscribers. Optional services are only available if you are using the Radia Software Manager client. See the *Radia Application Manager* and *Radia Software Manager Guides* for more information about mandatory versus optional services.

7. In the **Management type** section, indicate how the application will be managed after it is deployed.

Management Type and Assignment Type correspond to the ZSVCMO service attribute value (See the *Radia Application Manager* and *Radia Software Manager Guides* for more information about ZSVCMO). This field can contain two values, depending on what you select for each type. *Table 3.1 ~ ZSVCMO Variables* below, displays the possible attribute values.

Table 3.1 ~ ZSVCMO Variables			
Assignment Type	**Management Type**	**Resulting ZSVCMO Value**	**Explanation**
Mandatory	Automatic	M	Service is deployed to all subscribers and managed by Radia.
Mandatory	Manual	MO	Service is deployed to all subscribers. It is not managed by Radia after deployment.
Optional	Automatic	OM	Service is deployed to only those subscribers that accept it. Service is managed by Radia.
Optional	Manual	O	Service is deployed to only those subscribers that accept it. It is not managed by Radia after deployment.

8. In the **Report on the following events** section, select each check box next to the events you would like to record. See the *Radia Application Manager* and *Radia Software Manager Guides* for more information.

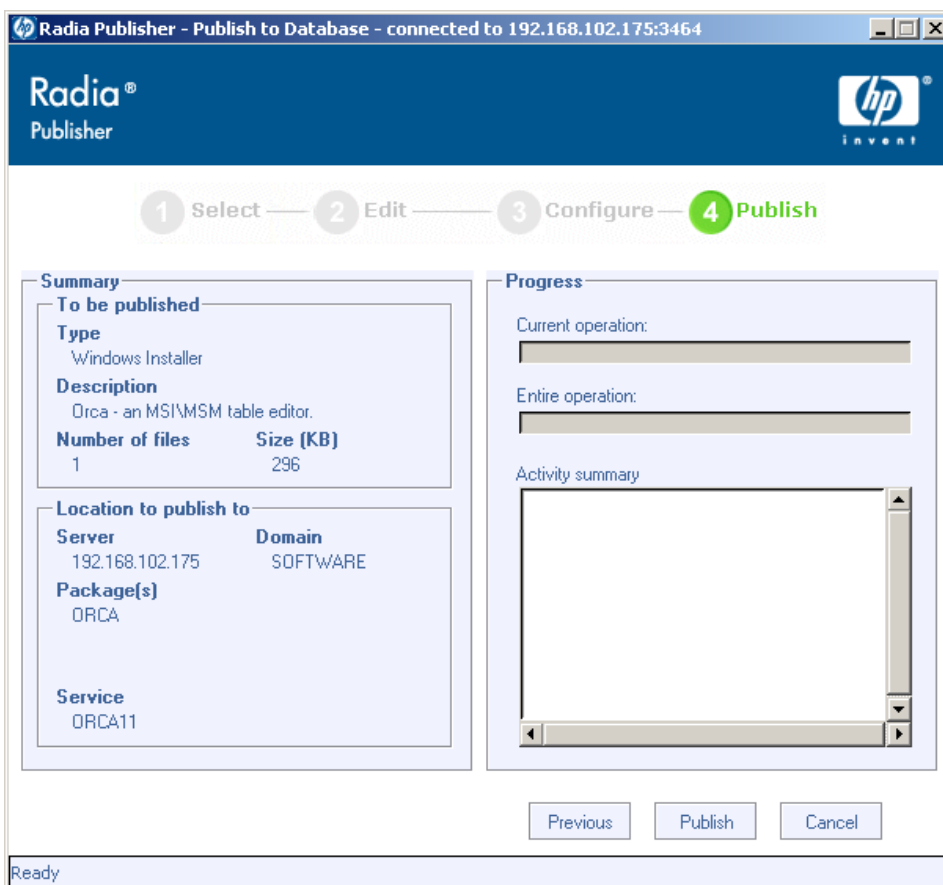**9.** Click **Next**.

The **Publish** window opens.

Figure 3.7 ~ Publish window.

10. Review the **Summary** section to verify the package and service information you provided during the previous steps. When you are satisfied, click **Publish**.

11. Click **Finish** when the publishing process is finished to exit the Radia Publisher. Use the Radia System Explorer to view your operating system package or service.
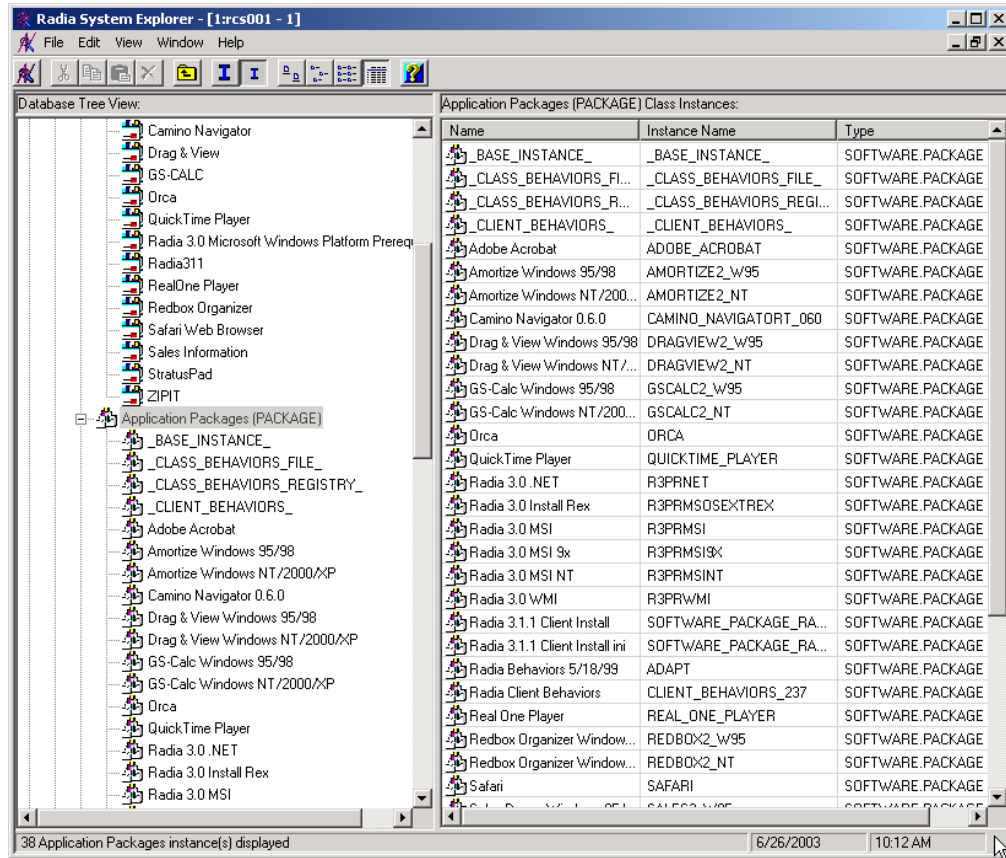
Figure 3.8 ~ New Radia package and service viewed in the Radia System Explorer.

The service is now ready for distribution to your enterprise.

Upgrading an Existing Package using an .MSP File

If you selected a Microsoft patch file (.msp) for publishing in step 6 on page 31 you will be presented with only the **Upgrade** option at the Publishing Options window. The Radia Publisher displays all available packages from the Radia Database that can be upgraded with the selected file. Select the corresponding package to be upgraded from the list on the right-hand side of the window.
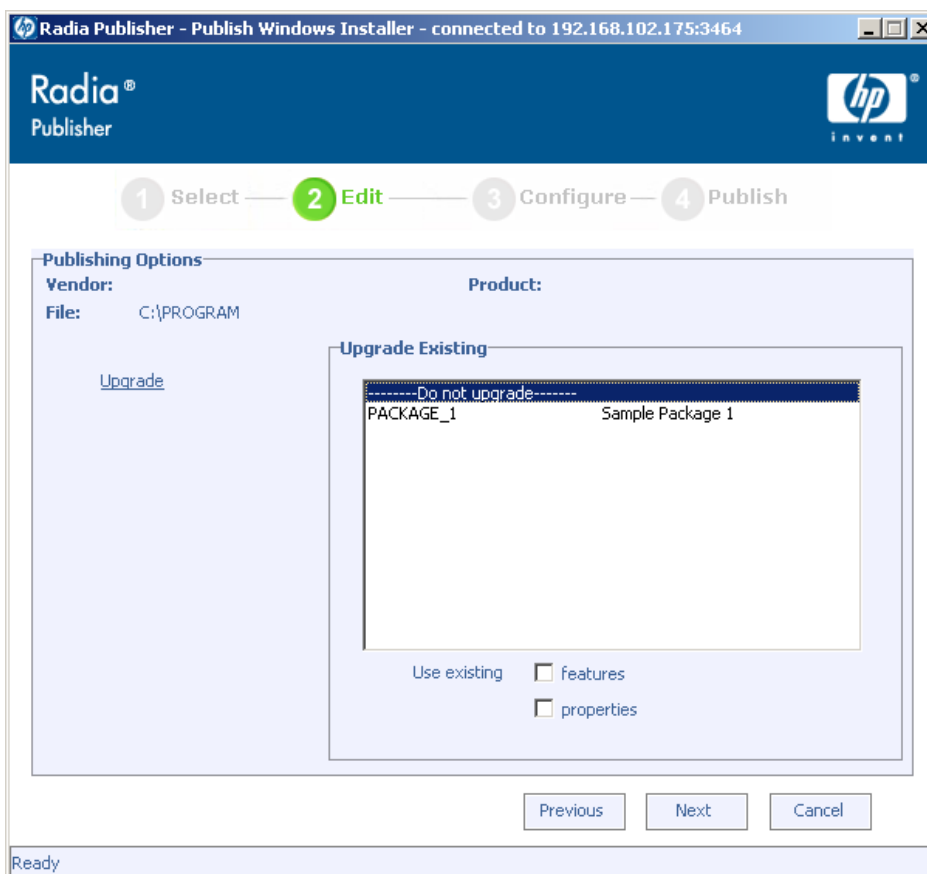

Figure 3.9 ~ Only the Upgrade option is available when using patch files (.msp).

Click either of the check boxes in the **Use existing** section at the bottom of the window in order to use the existing package's features or properties. Select both check boxes to use both the existing package's features and properties.

You can also upgrade an existing package using another Windows Installer file. The Advanced Publishing Mode also presents you with the Upgrade option when publishing a Windows Installer file (.msi).

> **Note**
>
> The Radia Publisher will create a new package for the upgrade. Once the new upgrade package is published to the Radia Database, manually connect it to the existing package using the Radia System Explorer.

Publishing Operating System Images

The following operating system image files are supported by the Radia Publisher:

■   Operating System image files (.IMG)

■   Power Quest image files (.PQI)

When an image file is published, a Radia package is added to the OS domain in the Radia Database. If you decide to create a service during the publishing steps, a Radia service is added to the ZSERVICE class of the OS domain.

In addition to image files, you can also publish Sysprep.inf files. When Sysprep.inf files are published to your database, they are prefixed with SPRP_ and stored in the SYSPREP class of the OS domain.

Note
Sysprep.inf files should be published in an uncompressed state.

To publish operating system images
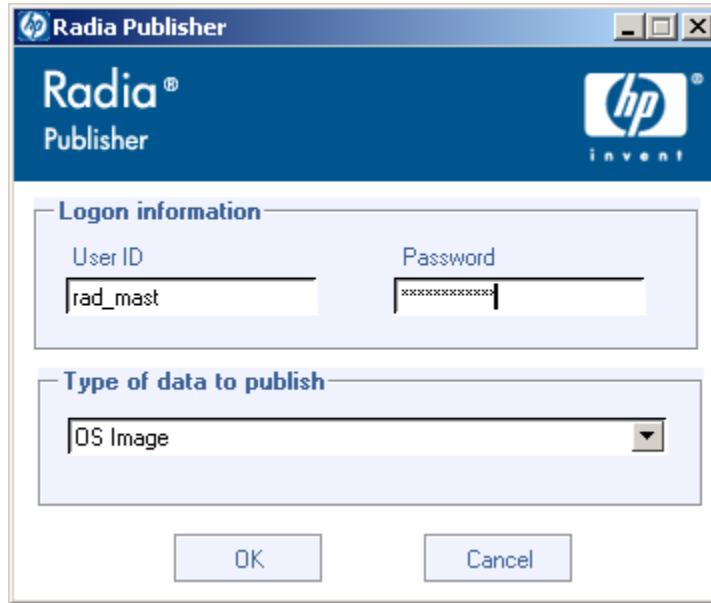
1.   Start the Radia Publisher.

Figure 3.10 ~ Login information.

2. In the **User ID** field, type your Radia administrator user ID.

3. In the **Password** field, type your Radia administrator password.

4. Whether you are publishing an OS image or a Sysprep.inf file, select **OS Image** from the **Type of Data to Publish** drop-down list.
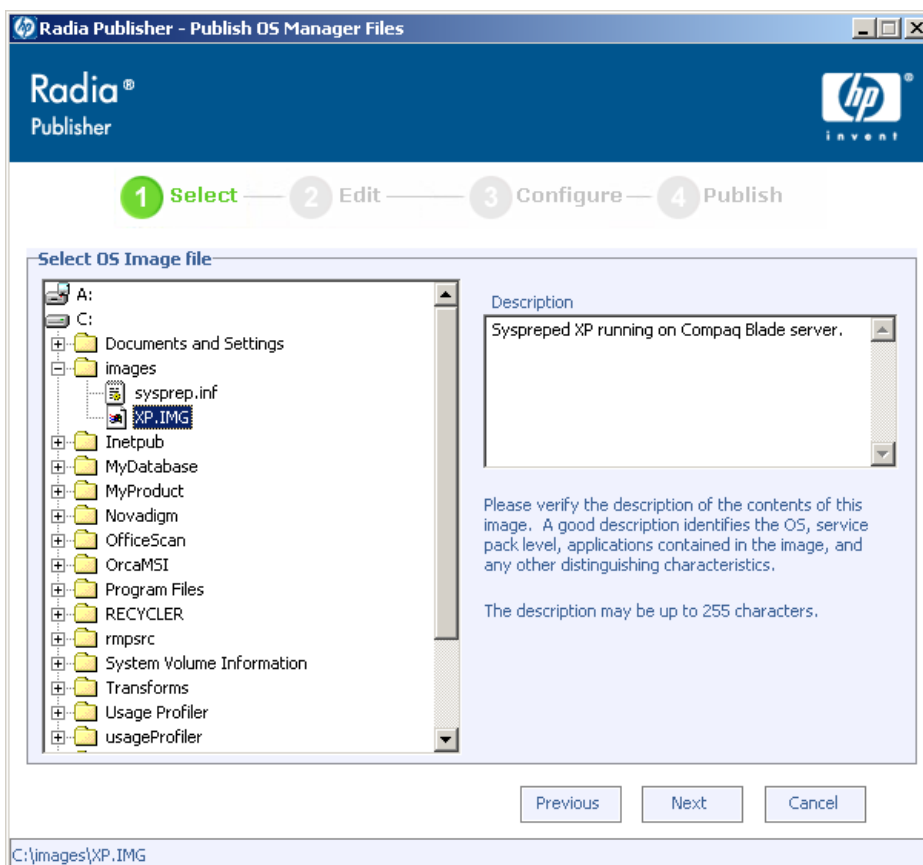
5. Click **OK**.

The **Select** window opens.

Figure 3.11 ~ Select the operating system image file to publish.

6. Use the **Select** window to find to select the file you want to publish.

> **Note about Sysprep.inf files**
>
> If you select a Sysprep.inf file, you will be prompted to enter a SYSPREP instance name. When you click **Next**, you will skip directly to the final step.
>
> SYSPREP files are published to the SYSPREP class in the OS domain of the Radia Database. Use the Radia System Explorer to view your published instances.

7. Use the **Description** window to verify the file before you continue. You can also add information to the description if you choose.

8. Click **Next**.

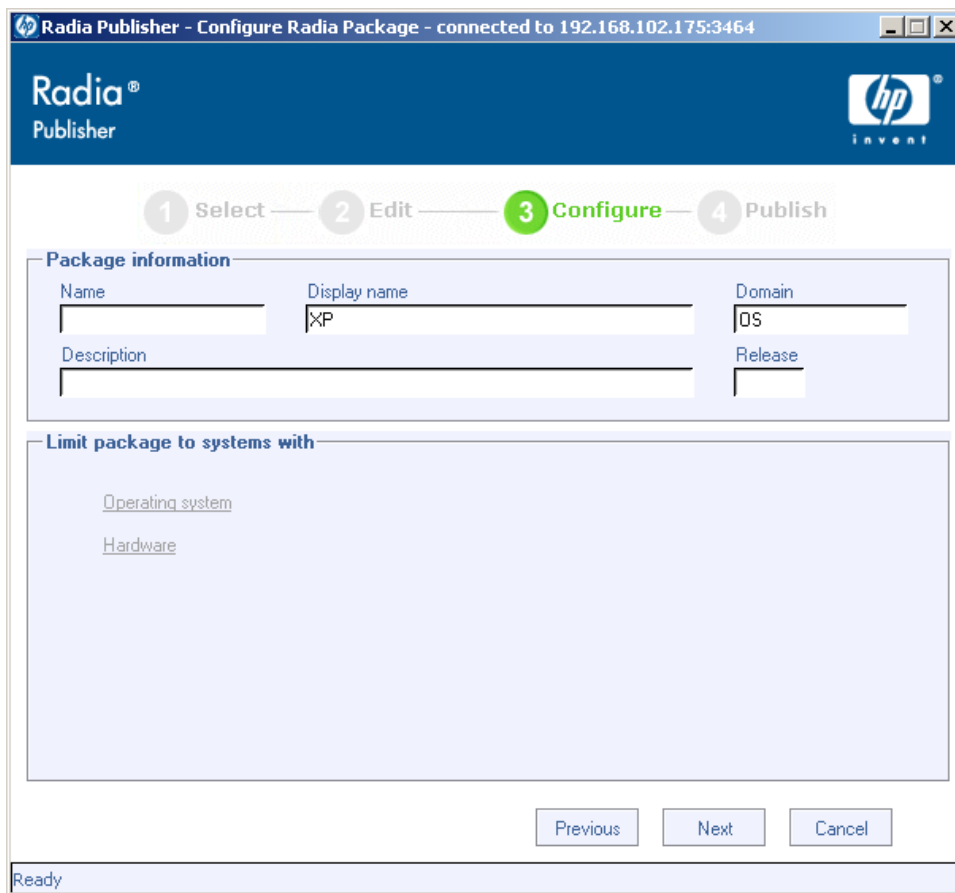Otherwise, the **Package Information** window opens.

Figure 3.12 ~ OS image Package information.

9. Use the **Package information** section to enter the Radia package information. Note that the **Limit package to systems with** section is not available when publishing operating system images.

10. Click **Next**.

The **Service Information** window opens.

Figure 3.13 ~ OS image service information.

11. Select whether you want to create a new service (**Create new**), use an existing service (**Use existing**), or skip creating a service (**No service**) at this time.

> **Note**
>
> If you want to create a package only, select **No service**. This is useful if, for example, you have a single service, but want to create multiple packages and later connect them to the existing service using the Radia System Explorer.

12. Enter the appropriate information in the service description text boxes. If you are using an existing service, make sure you enter the service name correctly.

13. In the **Assignment type** section, select whether the service is mandatory or optional. By default, **Mandatory** is selected, which will distribute this service to all available subscribers. Optional services are only available if you are using the Radia Software Manager client. See the *Radia Application Manager* and *Radia Software Manager Guides* for more information about mandatory vs. optional services.

14. In the **Management type** section, indicate how the application will be managed after it is deployed.

Management type and Assignment type correspond to the ZSVCMO service attribute value (See the *Radia Application Manager* and *Radia Software Manager Guides* for more information about ZSVCMO). This field can contain two values, dependant upon what you select for each type. *Table 3.2 ~ ZSVCMO Variables* below, displays the possible attribute values.

Table 3.2 ~ ZSVCMO Variables			
Assignment Type	**Management Type**	**Resulting ZSVCMO Value**	**Explanation**
Mandatory	Automatic	M	Service is deployed to all subscribers and managed by Radia.
Mandatory	Manual	MO	Service is deployed to all subscribers. It is not managed by Radia after deployment.
Optional	Automatic	OM	Service is deployed to only those subscribers that accept it. Service is managed by Radia.
Optional	Manual	O	Service is deployed to only those subscribers that accept it. It is not managed by Radia after deployment.

15. In the **Report on the following events** section, select each check box next to the events you would like to record. See the *Radia Application Manager* and *Radia Software Manager Guides* for more information

16. Click **Next**.
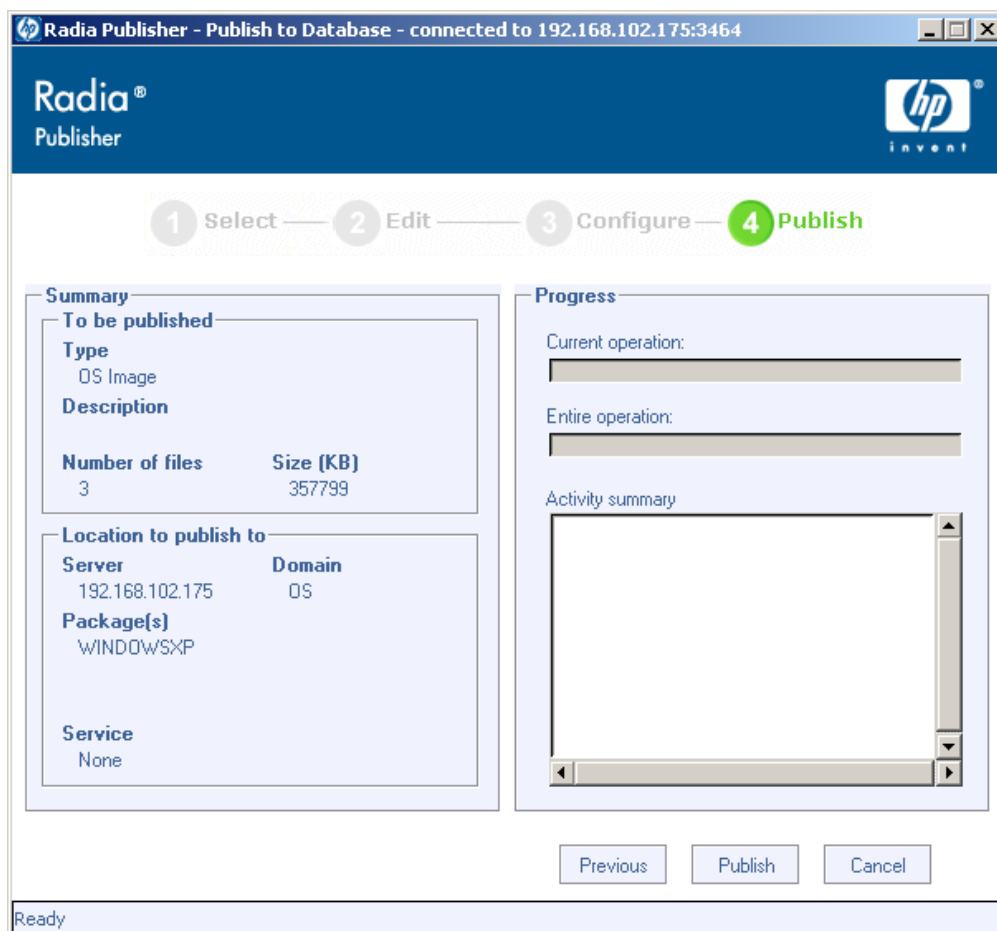
The **Publish** window opens.

Figure 3.14 ~ OS image publishing session complete.

17. Review the **Summary** section to verify the package and service information you provided during the previous steps. When you are satisfied, click **Publish**.

18. Click **Finish** when the publishing process is complete to exit the Radia Publisher.

Use the Radia System Explorer to view your operating system package or service in the OS domain.

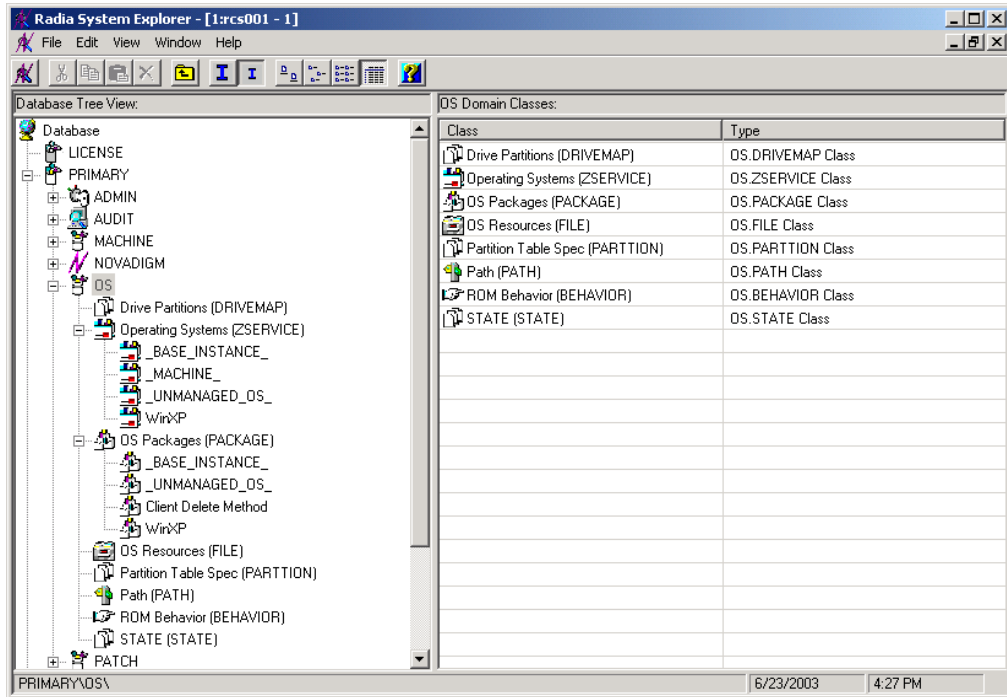| Note |
| :--- |
| Remember, SYSPREP files are published to the SYSPREP class in the OS domain of the Radia Database. Use the Radia System Explorer to view your published instances. |

Figure 3.15 ~ OS package and service are visible using the Radia System Explorer.

The service is now ready for distribution into your enterprise.

Command Line Publishing

The Radia Publisher supports a non-graphical publishing mode, substituting any values entered during the GUI publisher with values predefined in a configuration file. The configuration file allows you to predefine any package and service values and store them in a text file. The Radia Publisher can then be run from the command line with the parameter **–cfg** designating which configuration file to use during the publishing session.

An example command line to run the Radia Publisher with a configuration file called **test.cfg**, would look like the following:

```
hide nvdkit nvdtk.exe/main.tcl pubport.tkd -cfg ↵
    "c:\program files\novadigm\test.cfg"
```

> **Note**
>
> Make sure to use a fully qualified filename when designating the configuration file.

The parameters of the configuration file are defined in *Table 3.3 ~ Configuration File Parameters* below.

Table 3.3 ~ Configuration File Parameters	
Parameter	**Description**
package	Defines the PACKAGE class instance name.
pkgname	Specifies the friendly name of the PACKAGE class instance (NAME attribute).
pkgdesc	Specifies a description of the PACKAGE class instance (DESCRIPT attribute).
service	Defines the name of the ZSERVICE class instance that will be optionally created.
svcname	Specifies the friendly name of the ZSERVICE instance (NAME attribute). The ZSVCNAME attribute is also set from this value.
svcdesc	Specifies a description of the ZSERVICE class instance (DESCRIPT attribute).
addtosvc	Controls whether or not a service instance is created for this session. 1 = create service 0 = don't create
path	Defines the Radia Database path to the file and domain to which the package will be published, for example, PRIMARY.SOFTWARE.

Table 3.3 ~ Configuration File Parameters

Parameter	Description

attr CLASS {body}	Additional instance attribute values to be added during promote. Note that the Radia Publisher only supports PACKAGE and ZSERVICE as values for CLASS. Any attribute to these classes can be passed in, even those not supported by the user interface. Examples:
	
	    attr ZSERVICE {
	AUTHOR {John J. User}
	                URL {http://www.nowhere.com}
	                OWNER {Microsoft Corp}
	                 }
	
	    attr PACKAGE {
	                RELEASE {1.0}
	                 }
	
	Values should be enclosed in braces.

msi {body}	This section specifies MSI user interface settings. Allowable settings are:
	
	aiplocation — This specifies where the AIP is to be created (if necessary).
	publishmode — Advanced or Basic.
	execui — How to execute the MSI's user interface during AIP creation. Allowable values are Full, Basic, or None.
	deleteaip — Whether or not to delete the AIP after publishing.
	aiparguments — optional arguments for AIP creation.
	enableperuser — Whether to disable per user installation of this package (N) or any combination of the following: S=shortcuts, R=registry, E=private exe, D=private dll. Specifying "SD", for example, will turn on per user installation for shortcuts and private dlls.
	
	An example msi section would resemble the following:
	msi {
	        aiplocation {c:\testaip}
	        publishmode advanced
	        execui none
	        deleteaip n
	        aiparguments ""
	        enableperuser sre
	}
	   Note: Values are case-sensitive.

A sample configuration file is displayed in *Figure 3.16 ~ Sample configuration file* on page 53.

```
pkgname      MYTESTPKG
package      CJMTEST
pkgdesc      {This is a test package}
path         PRIMARY.SOFTWARE

addtosvc     1
service      CJMTESTS
svcname      MYTESTSVC
svcdesc      {This is a test service}

attr PACKAGE {
        RELEASE 1.0
        BEHAVIOR
}

attr ZSERVICE {
        AUTHOR        me
        URL           {HTTP://WWW.NOWHERE.COM}
        OWNER         {Microsoft Corp.}
        PRICE         {5}
}
msi {
        aiplocation    {g:¥aip}
        publishmode    advanced
        deleteaip      n
        aiparguments   {}
        enableperuser  n
```

Figure 3.16 ~ Sample configuration file.

Defining which File to Publish on the Command Line

In addition to defining a configuration file, you can define the file to be published on the command line. Use the –**file** parameter to define which file to use in the publishing session.

```
hide nvdkit nvdtk.exe/main.tcl pubport.tkd –cfg "c:\program ↵ files\novadigm\test.cfg" –file
    "c:\program files\Novadigm\testfile"
```

Publishing Multiple .msi Files (Batch Mode)

The Radia Publisher allows for the publishing of multiple .msi files to the Radia Database during one publishing session. This is achieved by calling the Radia Publisher from the command line and designating a location using the **–batchmsi** parameter. The Radia Publisher then scans this location and all subdirectories for .msi files as well as any configuration files or transform (.mst) files.

> **Note**
>
> Unlike publishing single .msi files using the command line, when publishing batches of .msi files with the Radia Publisher, the command line parameter **–cfg** cannot be used to designate a configuration file. Instead, at least one configuration file must be included within the .msi file directory location.

Each directory should contain the components of a Radia Package that will be published to the Radia Database, including at least one .msi file as well as one configuration file and any applicable transform files.

The files contained in the designated location must follow a few simple rules:

- If more than one .msi file exists in a directory, an error will be recorded and the publishing session scan will continue.
- If more than one configuration file exists in a directory, an error will be recorded and the publishing session scan will continue.
- If desired, only one configuration file can be used for all .msi files found. When only one configuration file is discovered in the root directory location, the Radia Publisher will apply that configuration to all .msi files it finds thereafter. To define a unique name to each package then, use symbolic substitution, as described later in this chapter.
- In order for the scan to be successful, at least one configuration file must be present at the root directory location.

A sample command line to run the Radia Publisher and designate the location to scan would look like the following:

```
hide nvdkit nvdtk.exe/main.tcl pubport.tkd –batchmsi "c:\msifileslocation"
```

This example will cause the Radia Publisher to scan the directory "c:\msifileslocation" and all subdirectories looking for .msi files as well as any configuration files or transforms.

When the publishing section is finished, verify the session by referring to the **publisher.log** file located in your log directory.

Using Multiple Configuration Files

Only one configuration file is required for all .msi files you are publishing as long as you are using symbolic substitution to assign unique package names, as discussed later in this chapter.

However, if you would like to apply a separate configuration file to each .msi, then a separate configuration file must reside within the directory of the .msi to which it will be applied.

Any configuration file found within a directory will be applied to the .msi file found within that directory and any subsequent .msi files found in each subdirectory, unless another configuration file is found.

For instance, the **Acrobat Reader 6.msi** file, as displayed in *Figure 3.18 ~ Configuration files within the directory structure* on page 56, will use the configuration file **global.cfg**, located in the root location directory, **/Adobe MSI Apps to Publish** (as seen in *Figure 3.17 ~ Root location directory containing a configuration file* below) since no configuration file is included in the **/Acrobat Reader 6.0** directory.
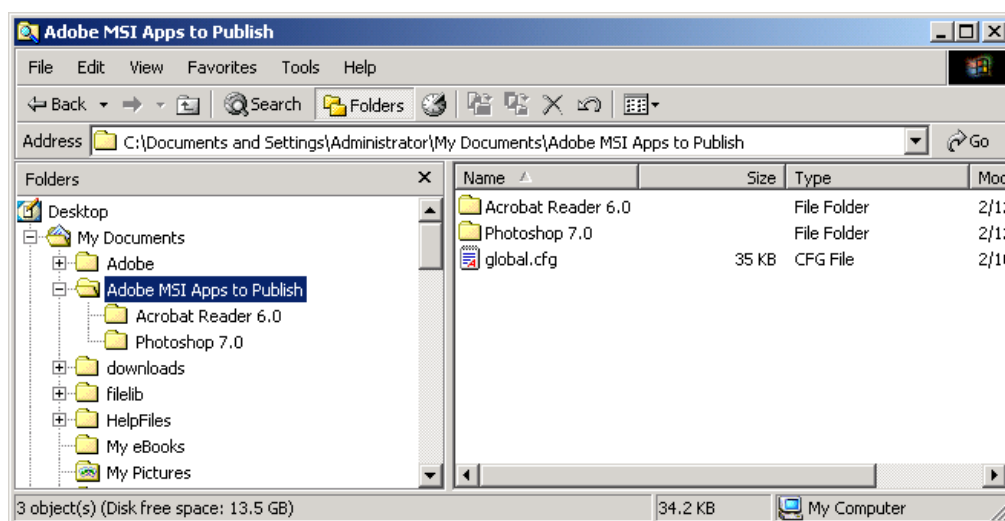
Figure 3.17 ~ Root location directory containing a configuration file.
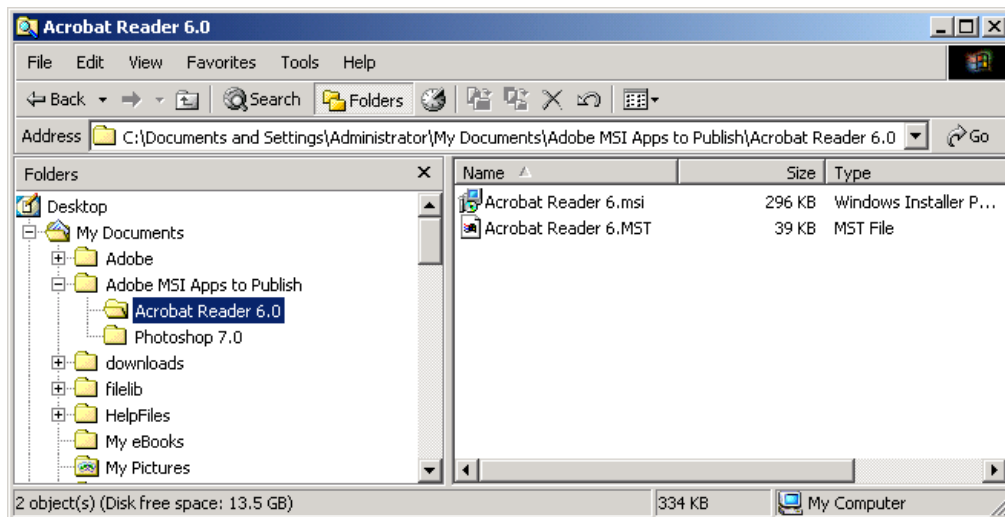
Figure 3.18 ~ Configuration files within the directory structure.

Including Transform (.mst) Files

You can apply any number of transform files with your .msi files by including those transforms in the same directory with the .msi file to which they will be applied. Transform files are only applied to the .msi file with which they share a directory.

Figure 3.19 ~ Transform files included in the directory structure on page 57, displays two transform files, both of which will be included with the **Photoshop 7.msi** package upon publication.
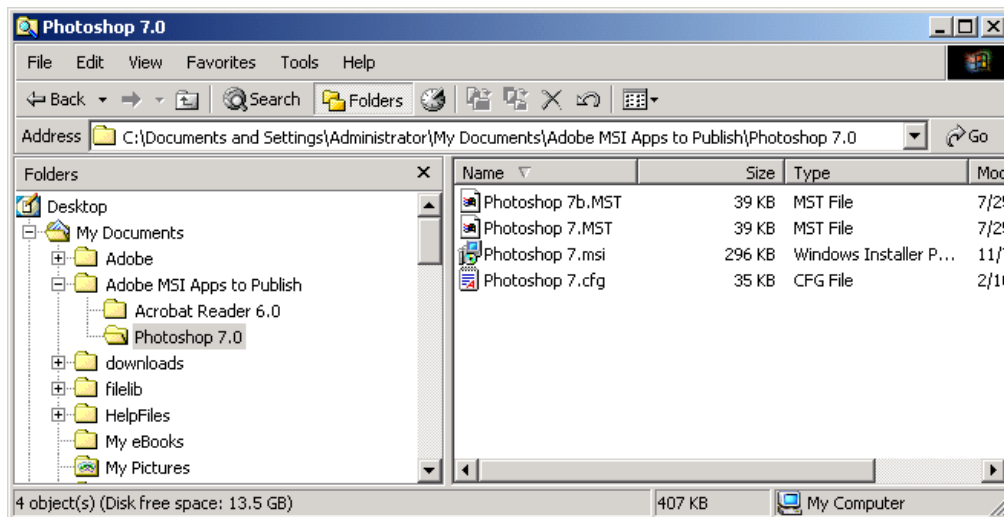
Figure 3.19 ~ Transform files included in the directory structure.

Customizing Configuration File Values with Symbolic Substitution

If you are publishing similar .msi files and would like to use only one configuration file, you can direct the Radia Publisher to assign unique values using symbolic substitution. For instance, create unique package names by setting the **pkgname** variable within the configuration file to %MSIPROP.VARNAME% or another symbolic variable as found within your .msi file. When used, the Radia Publisher will search the current individual .msi file it is publishing and substitute the pkgname value with the corresponding value found within the .msi file, based on the symbolic value you supplied.

A sample configuration file displaying the use of symbolic substitution is displayed in *Figure 3.20 ~ Sample configuration file displaying symbolic substitution* on page 59.

```
# -----------------------------------------------------------------------------
#       Radia Publisher - Sample imbedded MSI Properties Configuration File
# -----------------------------------------------------------------------------


#------------------------------------------------------------------------------
#    Package related variables
#------------------------------------------------------------------------------
pkgname               {"%MSIPROP.RADIA_PKG_DISPLAY_NAME%"}
package               {"%MSIPROP.RADIA_PKG_INSTANCE_NAME%"}
pkgdesc               {"%MSIPROP.RADIA_PKG_DESCRIPTION%"}
path                  {"%MSIPROP.RADIA_PKG_PUBLISHPATH%"}    # PRIMARY.SOFTWARE


#------------------------------------------------------------------------------
#    Service related variables
#------------------------------------------------------------------------------
addtosvc              {%MSIPROP.RADIA_ADDTOSVC%}             # 0 = no, 1 = yes
service               {"%MSIPROP.RADIA_SVC_INSTANCE_NAME%"}
svcname               {"%MSIPROP.RADIA_SVC_DISPLAY_NAME%"}
svcdesc               {"%MSIPROP.RADIA_SVC_DESCRIPTION%"}

attr PACKAGE {
        RELEASE       {"%MSIPROP.RADIA_SVC_AUTHOR%"}         # 1.3.5.7
}


attr ZSERVICE {
        AUTHOR        {"%MSIPROP.RADIA_SVC_AUTHOR%"}
        URL           {"%MSIPROP.RADIA_SVC_URL%"}
        OWNER         {"%MSIPROP.RADIA_SVC_OWNER%"}
        PRICE         {"%MSIPROP.RADIA_SVC_PRICE%"}
        ZSVCMO        {"%MSIPROP.RADIA_SVC_ZSVCMO%"}         # (i.e., M,O,MO,OM)
        ZCREATE       {"%MSIPROP.RADIA_SVC_ZCREATE%"}        # (i.e., radiamsi /I /qu)

msi {
        aiplocation   {"%MSIPROP.RADIA_AIPLOCATION"%}        # c:\temp\RadiaAIP\
        publishmode   {"%MSIPROP.RADIA_PUBLISHMODE"%}        # advanced, basic
        execui        {"%MSIPROP.RADIA_EXECUI"%}             # full, basic, none
        deleteaip     {"%MSIPROP.RADIA_DELETEAIP"%}          # yes, no
        aiparguments  {"%MSIPROP.RADIA_AIPARGS"%}            #
        enableperuser {"%MSIPROP.RADIA_ENABLEPERUSER"%}      # s,r,e, and d (i.e., sre)
        addfiles      {"%MSIPROP.RADIA_ADDFILES"%}           # comma separated list
        features      {"%MSIPROP.RADIA_FEATURES"%}           # comma separated list
        transform     {"%MSIPROP.RADIA_TRANSFORM"%}          # comma separated list
        mgttype       {"%MSIPROP.RADIA_MGTTYPE"%}            # manual, automatic
}
```

Figure 3.20 ~ Sample configuration file displaying symbolic substitution.

Notice in the example above every value for each package and service created will be generated from the associated .msi file property table.

There are two primary MSI substitution value sources.

- **%MSISUMM_*summary_variable_name*%** - extracted from the Summary information associated with the MSI file.

- **%MSIPROP_*property_name*%** - extracted from the Property table associated with the MSI file.

Embedded MSI Properties and Radia Extensions for Windows Installer User-Defined Properties

Radia Extensions for Windows Installer's Package Wizard, Advanced option, User Defined Properties, allows you to automatically add a set of properties and their default values to every .msi package created using the Radia Advanced Publisher. These properties can be manually added to any package whether created by Wise, InstallShield, or another vendor.

A default set of condition testing can also be included in each package to ensure that these values have been updated by the administrator during the packaging process. This prevents package condition test failure.

Refer to the *Radia Extensions for Windows Installer Guide* for more information.

Summary

■ Use the Radia Publisher to publish operating system images or Windows Installer files to your Radia Database.

■ Radia Packages and Services are visible in the Radia domain by using the Radia System Explorer.

■ Operating system services and packages are stored in the OS domain of the Radia Database.

■ The Radia Publisher can publish batches of .msi files.

■ Use symbolic substitution within your configuration file to avoid using multiple configuration files when publishing multiple .msi files.

■ Transform files to be included while publishing a batch of .msi files should reside within the same directory as the .msi file to which they apply.

Figures

Tables

Procedures

Index