Deployment Guide

Version 13

Prism Deploy® Guide published June, 2012

This publication could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. New Boundary Technologies may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time.

Copyright 2001-2012 by New Boundary Technologies, Inc. All rights reserved.

This manual, as well as the software described in it, may only be used or copied in accordance with the terms of the license agreement included with the Prism Suite installation and product.

Trademarks

The following trademarks apply to this volume:

NEW BOUNDARY TECHNOLOGIES, the New Boundary Technologies logo, Prism Suite, Prism Patch Manager, and Prism Asset Manager are registered trademarks of New Boundary Technologies, Inc.

LANOVATION, Smart Update, Prism Deploy, and the Prism Deploy logo are trademarks of New Boundary Technologies, Inc.

Express Inventory technologies are copyright Express Metrix, LLC 2012

Microsoft and Active Directory are registered trademarks of the Microsoft Corporation.

Windows, Windows NT, Windows 2000, Windows XP, Windows Vista, and Windows 7 are trademarks of the Microsoft Corporation.

All other products and companies are trademarks or registered trademarks of their respective companies.

Patent

Prism Suite is protected under US patent numbers 7,707,571, 7,568,018, and 6,564,369

Additional Notes

Unless otherwise noted, all names of companies, products, and persons contained herein are part of a completely fictitious scenario or scenarios and are designed solely to document the use of the product.

New Boundary Technologies, Inc.
1300 Godward Street N.E. Suite 3100
Minneapolis, MN 55413

Phone (toll free): 800-747-4487
Phone (local): 612-379-3805
Fax (local): 612-378-3818

URL: *www.newboundary.com*

6-13-2007

Table of Contents

Deploying Changes through the Console

Deploying Software with Prism

Prism Deploy® deploys software and administrative tasks to multiple computers simultaneously, no matter where they are in the world, no matter the task. Whether you need to roll out a software suite, or complete a mundane and otherwise manual change (for example, sending out system updates corporate-wide), Deploy automates it. You can also delete unapproved software from computers; even update security permissions on a registry key or file – all without ever visiting a workstation or server. The best part is, when Deploy installs software it does so more reliably than any other tool on the market. Post-rollout visits to fix problems are a thing of the past.

With Deploy, you can ensure computers have the software and patches they need. Instead of running a time-consuming inventory scan to determine which computers need an application, just build the package and give Deploy the profile of the computers that need it. Deploy takes care of the rest. If a computer's setup changes so it meets the profile, or if a new computer joins the network fitting the description, it automatically receives the software it needs, all without further intervention from you.

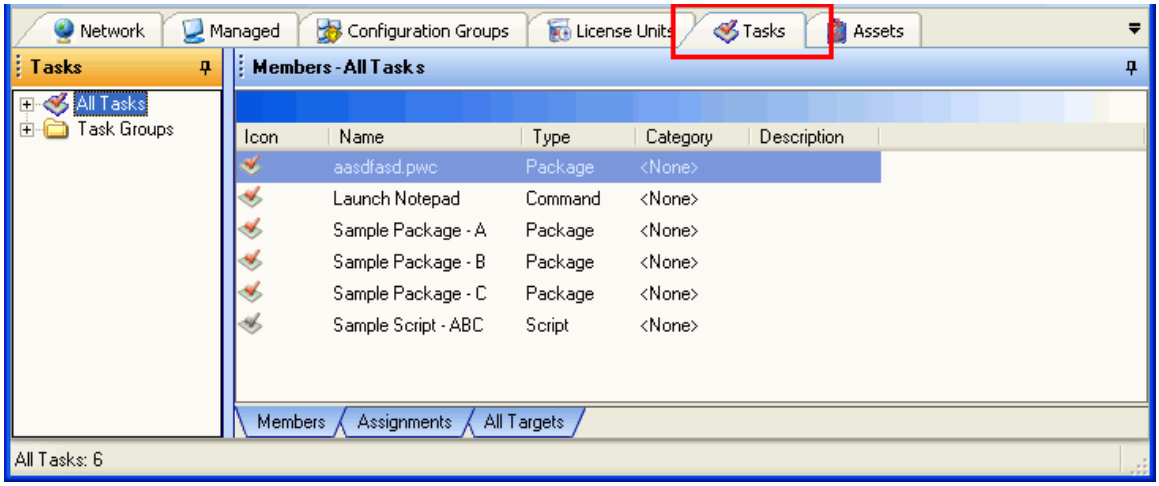
Overview of Channel Tasks

Tasks are the Prism files, scripts, and commands you want to install or execute on target computers. After setting up a Task in the Channel, drag that Task onto a computer, or a group of computers. For Task installation, the users do not have to be logged on or be aware of the changes.

There are four main types of Tasks:

- **Package**: The Task installs Prism files containing software, updates, deletions, or other system changes.
- **Command**: The Task runs a command or command file. This type of Task is very versatile, letting you run any type of third-party command that runs on the target computer—a command, an Internet address, or the name of a script file. To use a Prism command, put it in a Prism script and chose Script for the Task type.
- **Script**: The Task plays a Prism script (.PTS). A script is useful for installing several Prism files that need to be installed in sequence.
- **MSI:** The task installs a Windows Installer (MSI) package.

Tasks tab



The Tasks tab lists all of the Tasks available in the Channel, which are available for deploying changes to managed computers. From the tree view, you can quickly deploy Tasks by dragging and dropping or create new Tasks based on Prism file, scripts, or commands.

To help you maintain and quickly locate key Tasks, you can create groups of Tasks with similar or related purposes. After creating a group, simply drag Tasks into the group to populate it. Tasks are always listed in the Tasks branch, but now are also listed within one or more groups.

This tab is available in the tree view on the left side of the Console main window.

Right Click Options — All Tasks branch

- **Create Task**: Add a Task to the Channel and specify the file or command to run when the Task is installed.

Right Click Options — Task

These options are available when you right click on a Task:

- **Assign Task**: Select the targets that should receive this Task. Deploy lists all of the managed computers and groups in the Channel.
- **Add To Group**: Select the Task group to which you wish to add the selected Tasks. This is visible only if Task groups exist.
- **Delete**: Remove this Task from the Channel. This option does not uninstall changes that have already been made with this Task.
- **Rename**: Rename the Task.
- **Package Editor**: Available for Package Tasks only, this option opens the Package file in the Editor, if the Editor is installed on the same machine as the Console. This option is available for Package Tasks that are not marked as Distributed. Any Package, including Distributed Packages, can be edited using the **Edit** button on the Task Properties - Command tab.
- **Reports**: Select from a list of deployment reports.
- **Properties**: View the properties for the Task. (See Task General Properties.)

Note: These options are available any time you right click on a Task. For example, you can select from the same options by right-clicking on a Task on the Managed | All Tasks tab.

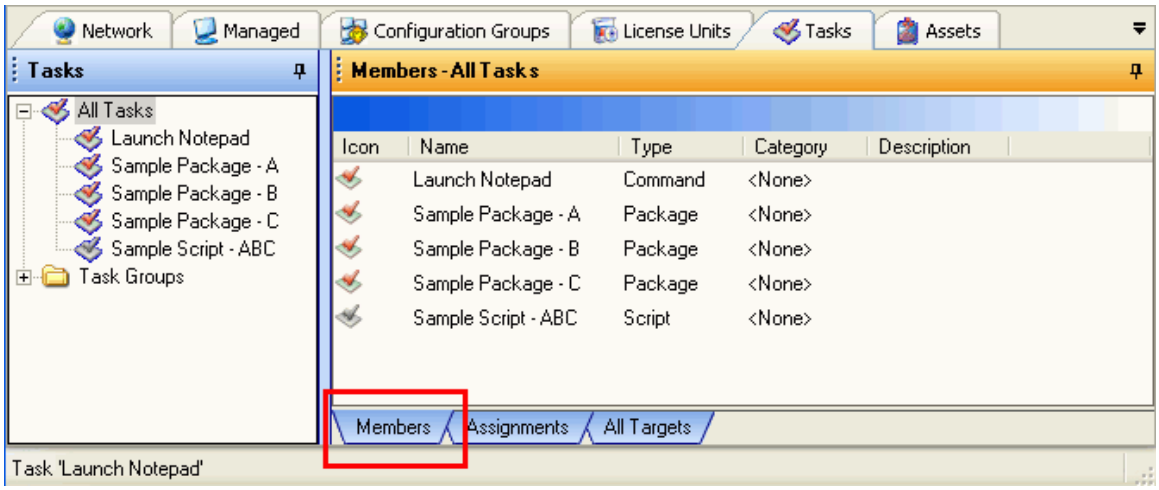
Right Click Options — Task Groups branch

- **Create Group**: Create a new group on the Tasks tab. After creating the group, populate it by dragging and dropping Tasks from the details pane onto the name of the group in the tree view. Use this feature to group Tasks with similar or related functions.

Right Click Options — Task Group

- **Create Task**: Add a Task to the Channel and specify the file or command to run when the Task is installed. The new task is automatically added to the selected Task Group.
- **Assign Task**: Select the targets that should receive this group of Tasks. Deploy lists all of the managed computers and groups in the Channel.
- **Create Group**: Create a new sub-group for the selected Task group.
- **Add To Group**: Select Tasks to add to the group.
- **Rename**: Rename the group.
- **Delete**: Delete the group from the Channel.

Members tab (Tasks)



The **Members** tab shows detailed information about the Tasks in the branch or group highlighted in the tree view.

This tab is available in the details pane when you have the Tasks tab selected in the tree view.

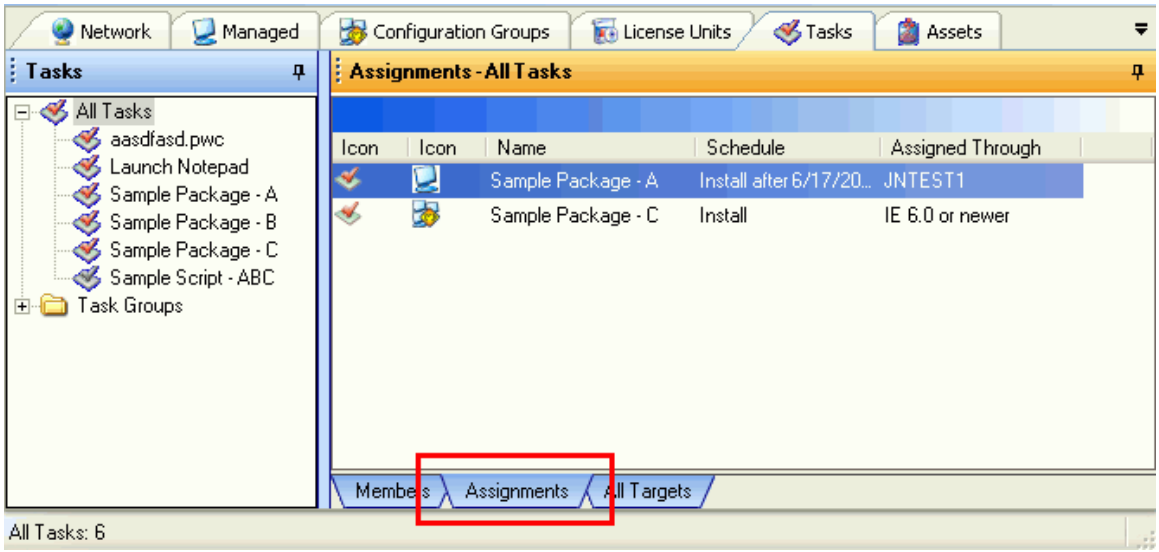
Right Click Options — Task

These options are available when you right click on a Task in the details pane:

- **Assign Task**: Select the targets that should receive this Task. Deploy lists all of the managed computers and groups in the Channel.
- **Add to Group**: Add the selected Tasks to a Task group.
- **Delete**: Remove this Task from the Channel. This option does not uninstall changes that have already been made with this Task.
- **Rename**: Rename the Task.
- **Package Editor**: Available for Package Tasks only, this option opens the Package file in the Editor, if the Editor is installed on the same machine as the Console. This option is available for Package Tasks that are not marked as Distributed. Any Package, including Distributed Packages, can be edited using the Edit button on the Task Properties - Command tab.

- **Reports**: Select from the Deployment reports in order to generate a report.
- **Properties**: View the properties for the Task. (See Viewing the Task General Properties.)

Assignments tab



The **Assignments** tab shows detailed information about assignments for the selection in the tree. This information includes the installation schedule, status, and the target the Task was assigned to originally.

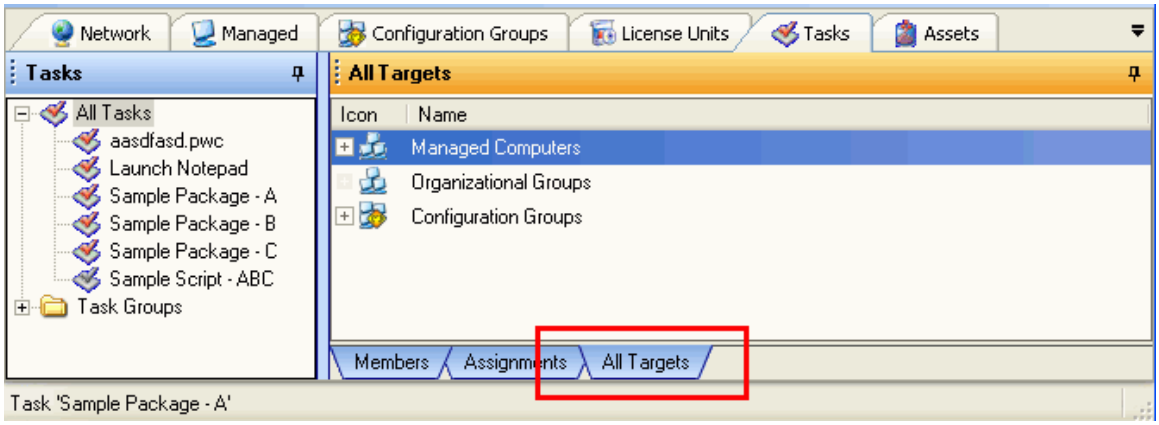
This tab is available in the details pane when you have the Tasks tab or ***Managed*** tab selected in the tree view.

Right Click Options — Task Assignment

These options are available when you right click on a Task Assignment in the details pane:

- **Schedule**: Change the schedule for the Task deployment.
- **Reinstall**: Reinstall the Task on the computers where it has already been installed. See Reinstalling or Repairing a Task.
- **Delete**: Unassign this Task from the current target. This option does not remove changes that have already been installed. It does stop this Task from being installed on selected targets in the future.

All Targets tab (Tasks)



This tab lists all of the managed computers and groups in the Channel. Use this list to quickly deploy a Task by dragging and dropping it from the tree view onto a target listed on this tab.

This tab is available in the details pane when you have the Tasks tab selected in the tree view.

Right Click Options — Computer

These options are available when you right click on a computer in the details pane:

- **Assign Task**: Assign a Task to the highlighted computer, and schedule installation.
- **Inventory Now**: If Asset Manager is installed, scan the highlighted computer or group to generate an inventory its hardware characteristics and the applications installed.
- **Request Poll**: Request an immediate poll of the client.
- **Delete**: Delete the computer from the channel.
- **Reports**: Select from the computer reports in order to generate a report.
- **Properties**: Display the computer's properties.

Right Click Options — Group

These options are available when you right click on a group in the details pane:

- **Assign Task**: Assign a Task to the highlighted group, and schedule installation.
- **Inventory Now**: If Asset Manager is installed, scan the highlighted group to generate an inventory its hardware characteristics and the applications installed.
- **Reports**: Select from the group reports in order to generate a report.
- **Properties**: Display the group's properties.

Creating Tasks

To add or set up a new Task in a Channel:

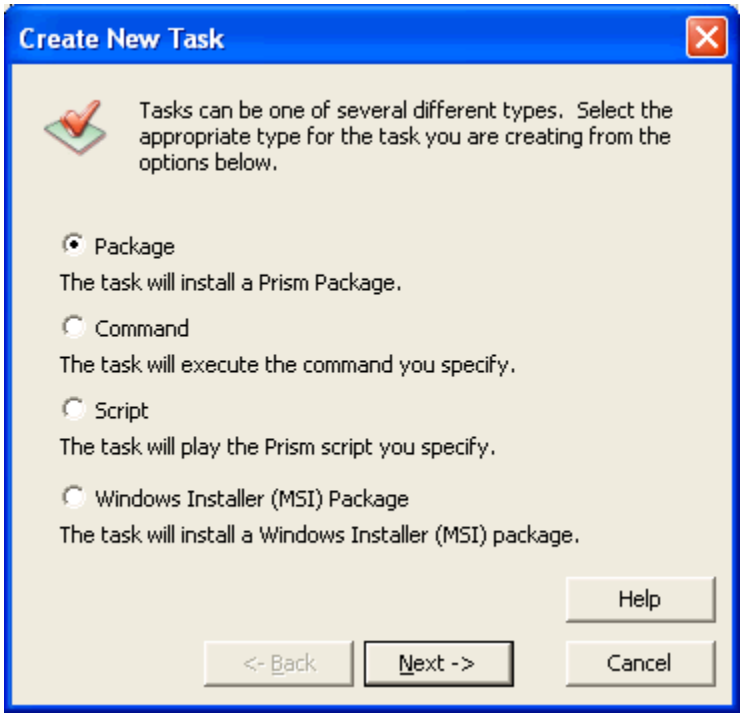
1. On the Tasks tab, press the  button in the toolbar. Alternately, you can select **Create Task** from the right-click menu on ***All Tasks*** or a Task Group, or the **Deployment menu**.
2. In the New Task dialog box, choose the appropriate type for the Task - Package, Command, Script, or MSI - you are creating. Then, click **Next**:
3. The next dialog box is specific to the type of Task (Package, Script, Command, or MSI) you selected. In this dialog box:
    - Enter the name or browse for the file to be used for the Task.

- Specify whether or not the Task is Distributed. This will determine how files associated with the Task are obtained by target computers.
- Depending on the Task type, you may need to provide additional information.

4. For Distributed Tasks, the next dialog will ask you to enter the names of files associated with the Task.
5. If you specified a location for a Task file that may not be appropriate for all target computers, Deploy displays a dialog box asking if you want to specify a different location.
6. In the New Task — Enter Name dialog box, type a descriptive name for the Task. Click **Finish**.

The Task is added to the Channel and can now be assigned to target computers or groups.

New Task - Select Type dialog

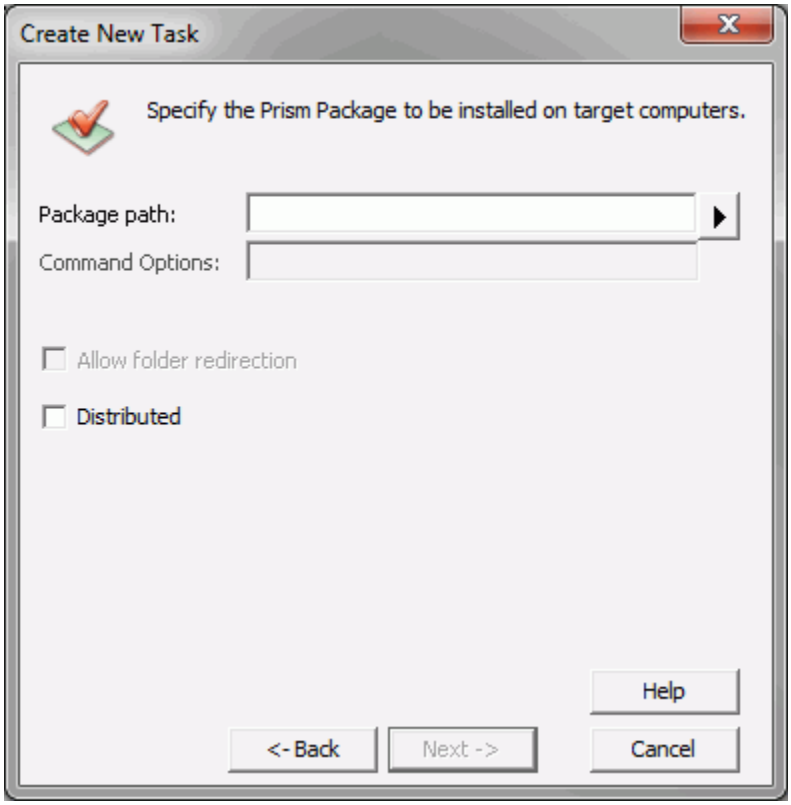


This dialog box opens when you add a new Task to the Channel. On this dialog box, choose the type of Task you want to create.

- **Package**: The Task installs a Prism file you created.
- **Command**: The Task runs the Command you type or plays a non-Prism script. This command can be an operating system hotfix, a URL to open a web page, a script, or any type of command that runs on the target computer.
- **Script**: The Task plays a Prism script (.PTS) you wrote. For example, a script is useful for installing several Prism files that need to be installed in sequence.

  To use Prism commands for a Task, they must be contained in a Prism script.
- **Windows Installer (MSI)**: The Task installs an Windows Installer (MSI) package.

New Package Task dialog



On this dialog, specify a Package Task.

This dialog appears as part of the Create Task wizard.

Package path

Enter the name and location for the Prism file to be installed by the Task. You can type the name directly into the text box or click  to browse for the file.

Command Options

This option is disabled for Package Tasks.

Allow folder redirection

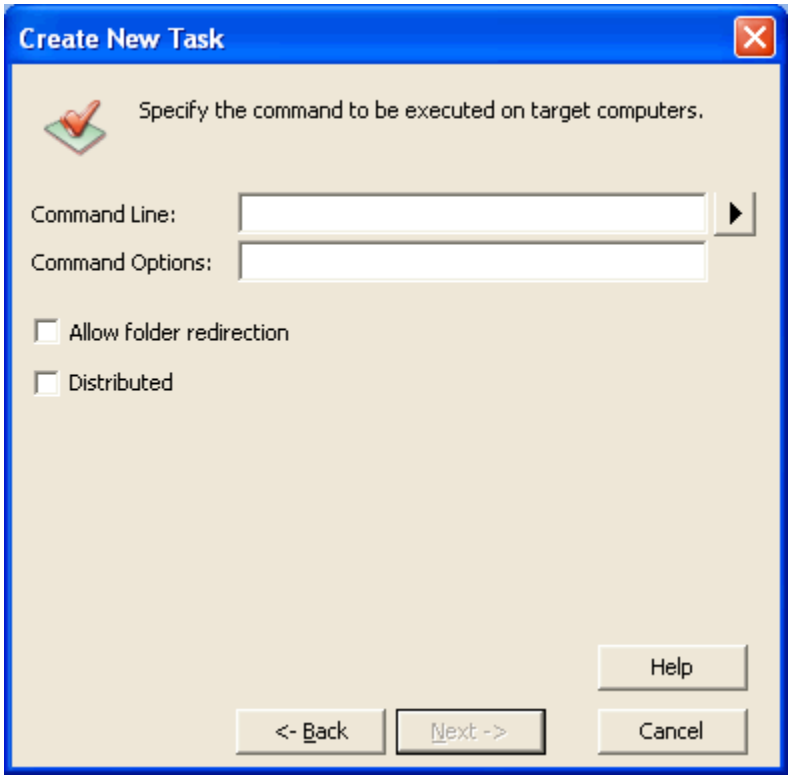
This option is disabled for Package Tasks.

Distributed

Indicates whether the Task's files should be Distributed:

- ☐ **Distributed**: With this option turned off, the Task becomes a Non-Distributed Task. The files required for a Non-Distributed Task must be stored in a location that is accessibly by all target computers, typically a network share.
- ☑ **Distributed**: With this option selected, the Task becomes a Distributed Task. The files and folders required for a Distributed Task are cached by the Prism server and synchronized from the server out to all Distribution Locations in the channel.

New Command Task dialog



On this dialog, specify a Command Task and its options.

This dialog appears as part of the Create Task wizard.

Command Line

Type the command to be executed by the Task or the name and location for the command file.

You can type the name directly into the text box or click  to browse for the file. For example, the command can be:

- A command that can be run on the target computer
- An internet address, such as a URL for a Web page that the Task will open (for example, http://www.yourcompany.com/mycommand.exe)
- A Windows command (for example, run mycommand.exe)
- A script file that is not a Prism script

Command Options

Enter options for the command in the Command Options text box.

Allow folder redirection

Specifies whether or not a command path should be redirected on 64-bit systems.

☑ **Allow folder redirection**: With this option selected, the command path will be automatically redirected on 64-bit systems to the 32-bit location for the command.

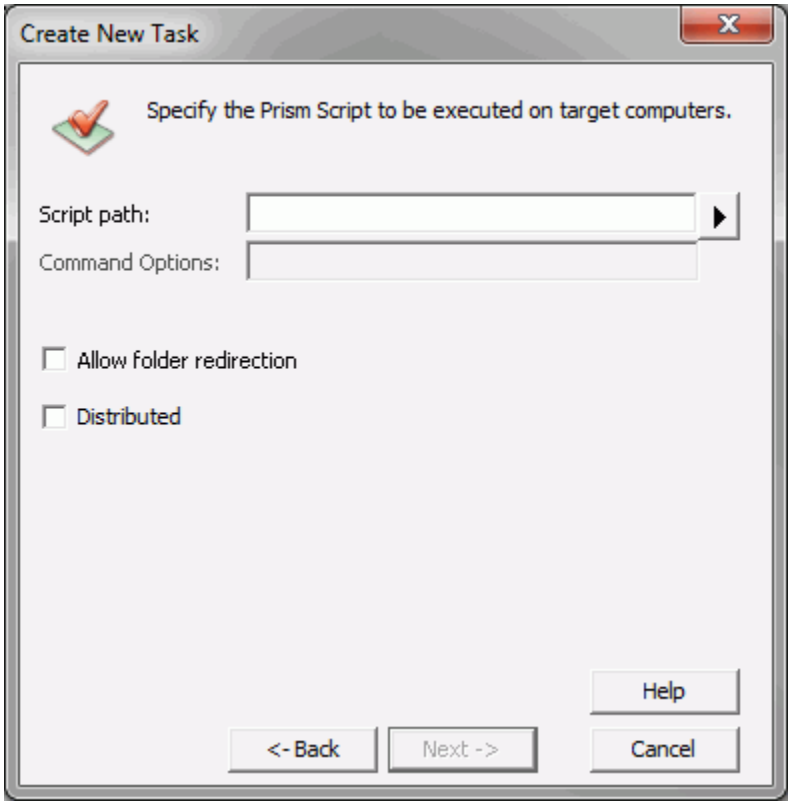
☐ **Allow folder redirection**: With this option cleared, the command path will be not be redirected on a 64-bit system. In this case, it is possible that on a 32-bit machine, the 32-bit version will be executed while on a 64-bit system, the 64-bit version will be executed.

Distributed

Indicates whether the Task's files should be Distributed:

- ☐ **Distributed**: With this option turned off, the Task becomes a Non-Distributed Task. The files required for a Non-Distributed Task must be stored in a location that is accessibly by all target computers, typically a network share.
- ☑ **Distributed**: With this option selected, the Task becomes a Distributed Task. The files and folders required for a Distributed Task are cached by the Prism server and synchronized from the server out to all Distribution Locations in the channel.

New Script Task dialog



On this dialog, specify a Script Task.

This dialog appears as part of the Create Task wizard.

Command Line

Enter the name and location for the Prism script to be played by the Task. You can type the file name directly into the text box or click  to browse for the file.

Command Options

This option is disabled for Script Tasks.

Allow folder redirection

Specifies whether or not a command path should be redirected on 64-bit systems.

- ☑ **Allow folder redirection**: With this option selected, the command path will be automatically redirected on 64-bit systems to the 32-bit location for the command.

☐ **Allow folder redirection**: With this option cleared, the command path will be not be redirected on a 64-bit system. In this case, it is possible that on a 32-bit machine, the 32-bit version will be executed while on a 64-bit system, the 64-bit version will be executed.

Distributed

Indicates whether the Task's files should be Distributed:

☐ **Distributed**: With this option turned off, the Task becomes a Non-Distributed Task. The files required for a Non-Distributed Task must be stored in a location that is accessibly by all target computers, typically a network share.

☑ **Distributed**: With this option selected, the Task becomes a Distributed Task. The files and folders required for a Distributed Task are cached by the Prism server and synchronized from the server out to all Distribution Locations in the channel.

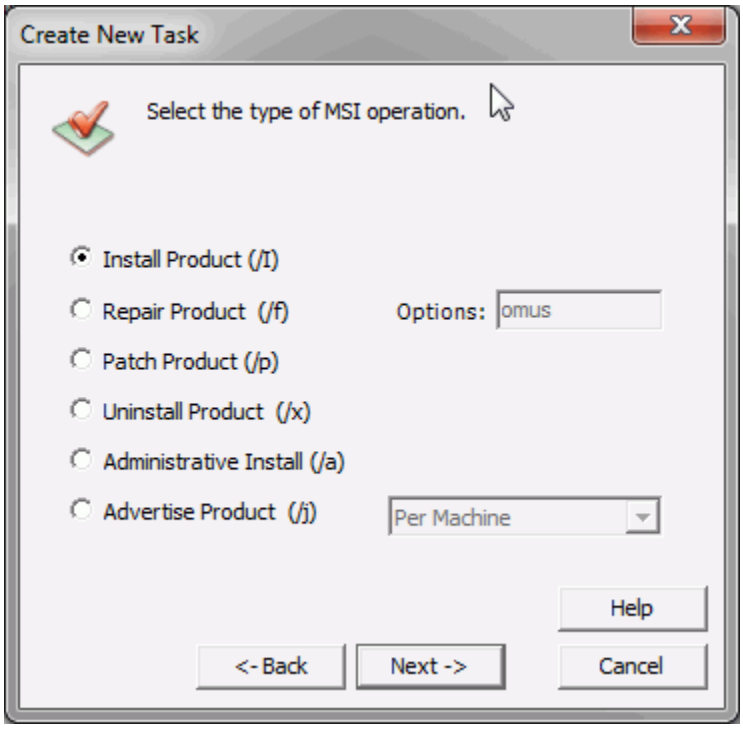
New Windows Installer (MSI) Task dialog

On this dialog and the subsequent dialog, specify a Windows Installer (MSI) Task. MSI Task creation involves 2 steps:

1. Select the type of MSI operation - for example, an installation;
2. Specify options for the selected operation - the MSI file and options such as logging and user interface.

These dialogs appears as part of the Create Task wizard.

Select MSI Operation Type

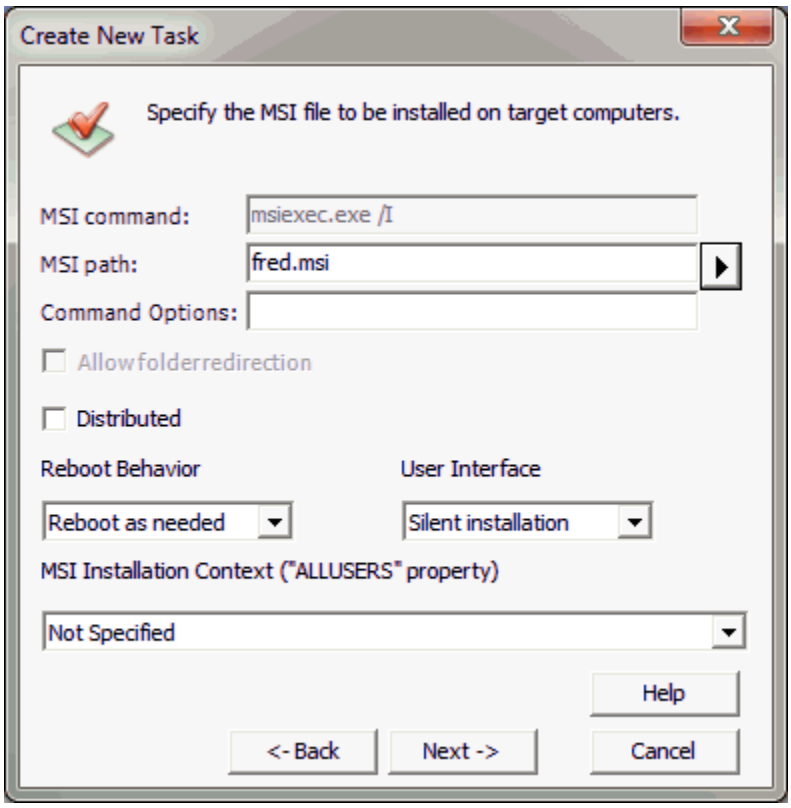


On the MSI operation type selection dialog, select the type of MSI Task to be created.

- Install: Install or configure a product.
- Repair: Repair a product. Default parameters are set to 'omus' but this can be overridden.
- Patch: Apply a patch.

- Uninstall: Uninstall a product.
- Administrative Install: Install a product on a network.
- Advertise Product: Advertise a product. This can be per machine (default) or per user.

Specify MSI File and Options



On this dialog, specify the options for the MSI Task.

MSI command

This read-only field displays the msiexec command corresponding to the selection in the previous dialog.

MSI path

Enter the path to the Windows Installer package. The command can be

- Microsoft Windows UNC (Universal Naming Convention). For example, `\\yourServer\yourShare\yourCommand.msi`.
- A mapped drive. For example, `F:\yourShare\yourCommand.msi`.
- An Internet address. For example, `http://www.yourcompany.com/yourCommand.msi`.

Command Options

Enter options and property values for the MSI in the Command Options text box. Refer to the Microsoft documentation for detailed information on the available command-line options and properties:

- Standard Installer Command-Line Options
- Command-Line Options

- Property Reference

Several command-line options are directly supported in the dialog; those do not need to be entered explicitly in Command Options. See Reboot Options, User Interface and MSI Installation Context sections below.

Note: The following Windows Installer command-line options are not supported: `/y, /z, /?, /h.`

Distributed

Indicates whether the Task's files should be Distributed:

☐ **Distributed**: With this option turned off, the Task becomes a Non-Distributed Task. The files required for a Non-Distributed Task must be stored in a location that is accessibly by all target computers, typically a network share.

☑ **Distributed**: With this option selected, the Task becomes a Distributed Task. The files and folders required for a Distributed Task are cached by the Prism server and synchronized from the server out to all Distribution Locations in the channel.

Reboot Options

This setting controls the reboot behavior. The selections correspond to Windows Installer `REBOOT` and `REBOOTPROMPT` property settings:

- **Reboot as needed** - Reboot occurs only as required by the installation.
- **Prompted reboot** - If the installation requires a reboot, the user will be prompted to reboot and select when to reboot. This is the same as Windows Installer property `REBOOTPROMPT=""` and Standard Installer command-line option `/promptedrestart`.
- **Never reboot** - Do not reboot, even if the installation requires it. This is the same as Windows Installer property `REBOOT=ReallySuppress` and Standard Installer command-line option `/norestart`.
- **Force reboot** - Always reboot after the installation has finished. This is the same as Windows Installer property `REBOOT=Force` and Standard Installer command-line option `/forcerestart`.

User Interface

This setting controls the feedback to the user. The selections correspond to Windows Installer `/q` command-line options:

- **Silent installation** - Hide all user interface elements of the installation. This is the same as Windows Installer option `/qn` and Standard Installer command-line option `/quiet`.
- **Show progress bar** - The installation will display a progress bar. This is the same as Windows Installer setting `/qb! REBOOTPROMPT=S` and Standard Installer command-line option `/passive`.
- **Show full interface** - Show the full user interface associated with the installation. This is the same as Windows Installer setting `/qf`.

MSI Installation Context

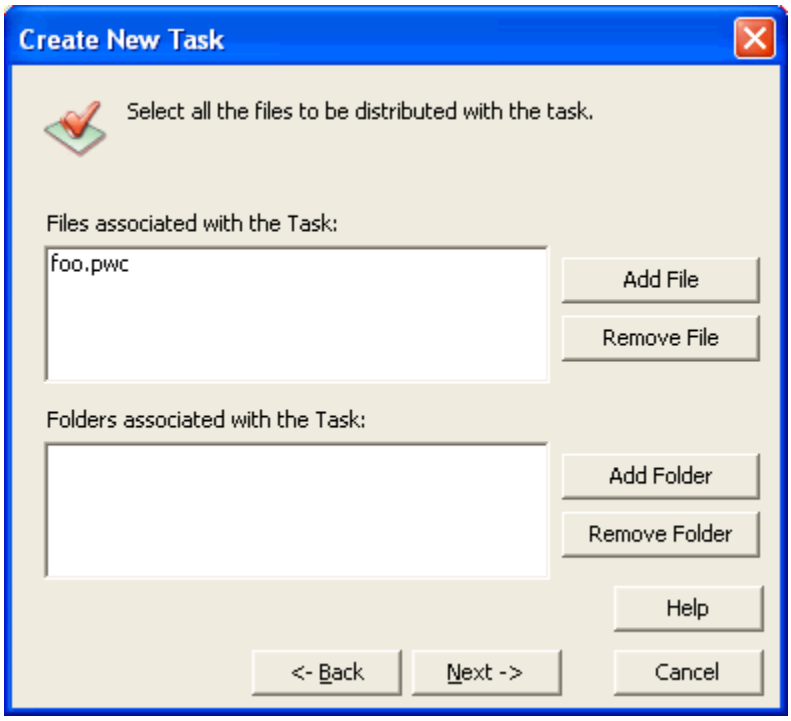
The **MSI Installation Context** drop down list specifies the installation context. The selections correspond to the Windows Installer `ALLUSERS` property settings. This option is only shown for MSI types Install and Uninstall.

- **Not Specified**: This option leaves the `ALLUSERS` property not set. If `ALLUSERS` is not set in the command options or otherwise set within the MSI package, the installer does a

per-user installation. To set `ALLUSERS=2`, select **Not Specified** and put `ALLUSERS=2` directly in the command options. The value ALLUSERS=2 enables the system to reset the value of **ALLUSERS** and the installation context, depending on the user's privileges and the version of Windows.

- **Available for all users of the computer**: Specifies per-machine installation context. This is the same as Windows Installer setting `ALLUSERS=1`.
- **Available only for the logged-on user**: Specifies a per-user installation context. This is the same as Windows Installer setting `ALLUSERS=""`.

New Task - Select Files dialog



This dialog lists the files and folders associated with a Distributed Task.

This dialog appears as part of the Create Task wizard for Distributed Tasks (that is, for Tasks whose Distributed checkbox is selected in the previous step).

The two lists display the files and folders associated with the **Distributed Task** that will be automatically synchronized out to all Distribution Locations, where they can subsequently be accessed by target computers. The file list includes the command file automatically. Additional files and folders can be added to the lists.

For example, if a Script Task is marked as Distributed and it references several files, these files may need to be added to this list so that computers in Distribution Locations can access them. This will cause them to be downloaded to the Distribution Location. When the script runs, those files will then be found in the Distribution Location's share.

Note To distribute a Script Task that references other files, the script must be edited by substituting `%DistShare%` as the path so that all references to those files can be resolved.

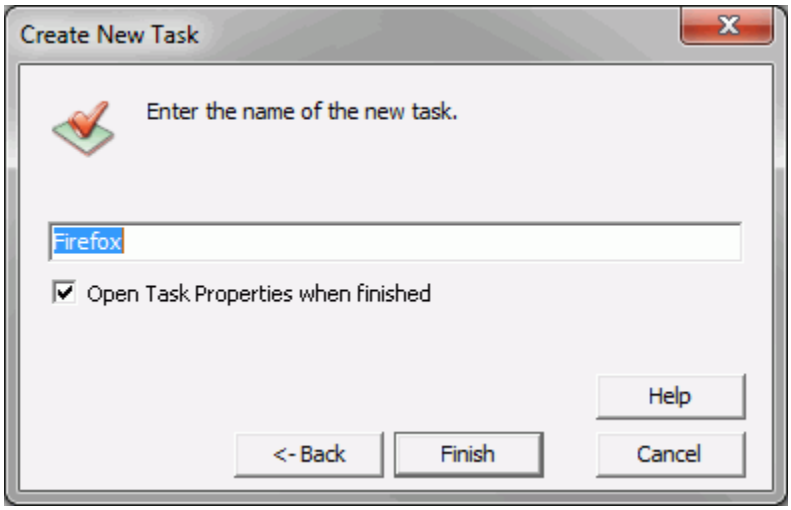
Manipulate the lists through the **Add** and **Remove** buttons (if the Task is not marked as **Distributed**, the lists are empty and the buttons are disabled):

- **Add File**: Opens a dialog that allows you to select files to be synchronized. The command file is automatically included in the list of files, so this does not need to be added separately.
- **Remove File**: Removes the selected file from the list.
- **Add Folder**: Opens a dialog that allows you to select folders to be synchronized. You can either browse to the folder's location or enter the folder name directly in the text box. Each folder - including all of its contained files and sub-folders - in the list will be synchronized out to the Distribution Locations.

> **Note** You cannot add the folder containing the command file to the list of folders to be synchronized. This can be inconvenient in cases where there is a single installation folder containing an installation command and multiple associated files and folders. A potential workaround is to create a batch file, at the same level as the installation folder, that invokes the installation command. In this case, the file list contains the batch file (instead of the installation command) and the folders list contains the single installation folder.

- **Remove Folder**: Removes the selected folder from the list.

New Task - Enter Name dialog



Type a descriptive name for the new Task. The name does not need to match the name of the Prism file or script that serves as the basis for the Task.

> **Note**: Task names must be unique.

Open Task Properties when finished

This checkbox controls whether not the Task Properties dialog opens automatically after pressing the **Finish** button.

☑ **Open Task Properties when finished**: Selecting this open causes the Task Properties dialog to open after **Finish** is pressed. This allows you to modify Execution options (Installation Scope and "Run as"), Uninstall settings, Prerequisites, and email Notifications options without having to open the Task Properties dialog at a later time. The Task is created when **Finish** is pressed, so cancelling the Task Properties dialog does not delete the new Task.

☐ **Open Task Properties when finished**: If this option is cleared, pressing **Finish** causes the Task to be created with default values for Execution options, Uninstall settings,

Prerequisites, and email notifications settings. The Task Properties for the new Task can be edited at a later time.

Distributing Task Files over the Network

Distributing Task Files

The files required for installing any Task must be available to the target computers in that channel. Prism Deploy provides two options to accomplish this for every Task: **Non-Distributed** and **Distributed**. When you create (or modify) a Task, you specify how you want its files distributed by checking or unchecking the **Distributed** checkbox:

☐ **Distributed**: With this option turned off, the Task becomes a Non-Distributed Task. The files required for a Non-Distributed Task must be stored in a location that is accessible by all target computers, typically a network share. The location of the file(s) required for the Task is relative to the target computers, and is part of the Task's definition.

☑ **Distributed**: With this option selected, the Task becomes a Distributed Task. The files and folders required for a Distributed Task are cached by the Prism server and synchronized from the server out to all Distribution Locations in the channel. A Distribution Location represents a group of computers in the channel that accesses files from their own common location. In contrast to Non-Distributed Tasks, the location of the files required for a Distributed Task does *not* need to be specified relative to the target computer. Instead, the location is specified within each Distribution Location.

Non-Distributed Tasks

A **Non-Distributed Task** is any Task whose **Distributed** checkbox is not checked. The files required for a Non-Distributed Task can be stored in any location that is available to all target computers, such as a file share. The location is specified for the Task during its creation or modification.

Note While this location does not need to be on the same computer as the Channel Server, the location specified *must reflect the location of the files from the target computers' point of view*.

The location can be in any of the following formats:

- Drive mapping (for example, F:\tasks\mycommand.exe)
- UNC path (for example, \\server1\tasks\mycommand.exe)
- Internet address (for example, http://www.yourcompany.com/mycommand.exe)

Deploy also supports use of environment variables in the location. For example, if you are providing a Task to users at different locations and each location has a server, use a variable for the home server - \\%server%.

Distributed Tasks

A **Distributed Task** is one whose associated files are copied into Distribution Locations. Distribution Locations represent a subset of computers that share a location-based characteristic, such as IP address range or domain name. Computers within each Distribution Location will access Task files from their unique share, rather than accessing them from a channel-wide centralized file share.

The files and folders required for a Distributed Task are cached by the Prism Server into a file cache and automatically synchronized out to all Distribution Locations in the channel. A target computer obtains a Distributed Task's files from the location specified by the Distribution Location it belongs to.

Making a Task **Distributed** requires the following actions:

1. Mark the Task's **Distributed** checkbox.
2. For all Tasks, specify any files and folders that need to be synchronized. For example, a Script Task that calls or runs other files will need those files to be specified.
3. Scripts must be edited to insert the %DistShare% variable into file references so that target computers can find the required files.

These steps can be applied during Task creation (when the command is specified) or later via Task Properties. When a user edits a Distributed task, any updated task files will automatically be synchronized out to the Distribution Locations.

Note If the Distributed flag is de-selected, the Task's files will be removed from the server file cache and will subsequently be removed from the Distribution Location.

Assigning Distributed Tasks

Distributed Tasks are assigned in exactly the same manner as non-distributed Tasks. Marking a Task as Distributed simply changes how target computers obtain the required files. It has no impact on *how* you assign the Task. Files associated with Distributed Tasks are synchronized out to Distribution Locations without user intervention - it occurs automatically and transparently.

Distribution Locations

A **Distribution Location** represents a subset of computers in the channel that share a location for storing Task files associated with Distributed Tasks. Computers within each Distribution Location will access Task files from their unique share, rather than accessing them from a channel-wide centralized file share.

Distribution Locations appear under the **Distribution Locations** node on the **Managed** tab.

A Distribution Location is a new type of group resembling a Configuration Group. Like Configuration Groups, membership in a Distribution Location is determined by evaluating rules. For example, a Distribution Location rule may define membership to include any computer whose IP address matches 192.168.1.* (that is, 192.168.1.0 to 192.168.1.255).

In contrast to Configuration Groups, each Distribution Location specifies its own local share from which each target computer in the group accesses files and folders associated with Distributed Tasks. Hence, those computers do not need to access files from a distant centralized location. Because members of the group have a common share where Task files are accessed, rules defining Distribution Locations are typically expressed in terms of a location-based characteristic such as IP address range, domain name, or computer name.

Distribution Locations are created on the Managed Tab by selecting the Distribution Location node and selecting Create Group via the context menu or the Edit menu.

Member Computers

Membership in a Distribution Location is determined automatically based on the rules in the group's definition. A computer can therefore change its membership over time, depending on how the rule is defined.

The member computers obtain information about their Distribution Location's share when they poll the server. If a member computer changes membership from one Distribution Location to another, it will obtain new Distribution Location information on its next poll and will subsequently access its new Distribution Location's share.

Distribution Agent

A **Distribution Agent** is a client within a Distribution Location that synchronizes files and folders associated with Distributed Tasks between the Prism Server and a share within the Distribution

Location. The Distribution Agent polls the server for any changes in the contents of that folder, and immediately downloads any changes. Member computers can then access these files from the Distribution Location's share.

Distribution Agents are configured through the Distribution Location's properties. Configuration includes selecting the Agent computer, specifying the share, and specifying access rights.

Self-distributed

In a Distribution Location that is **Self-distributed**, every member computer in the Distribution Location independently downloads files from the central server. These Distribution Locations have no common share, nor do they have a single, designated Distribution Agent. In a sense, each member in a Self-distributed Distribution Location acts as its own Distribution Agent.

The primary use of this type of Distribution Location is for collections of computers that are disconnected from the corporate domain and from each other. For example, the laptops of sales people might fall into this type of Distribution Location. Self-distributed Distribution Locations can be used whenever it is inconvenient or impossible for member computers to access a common share.

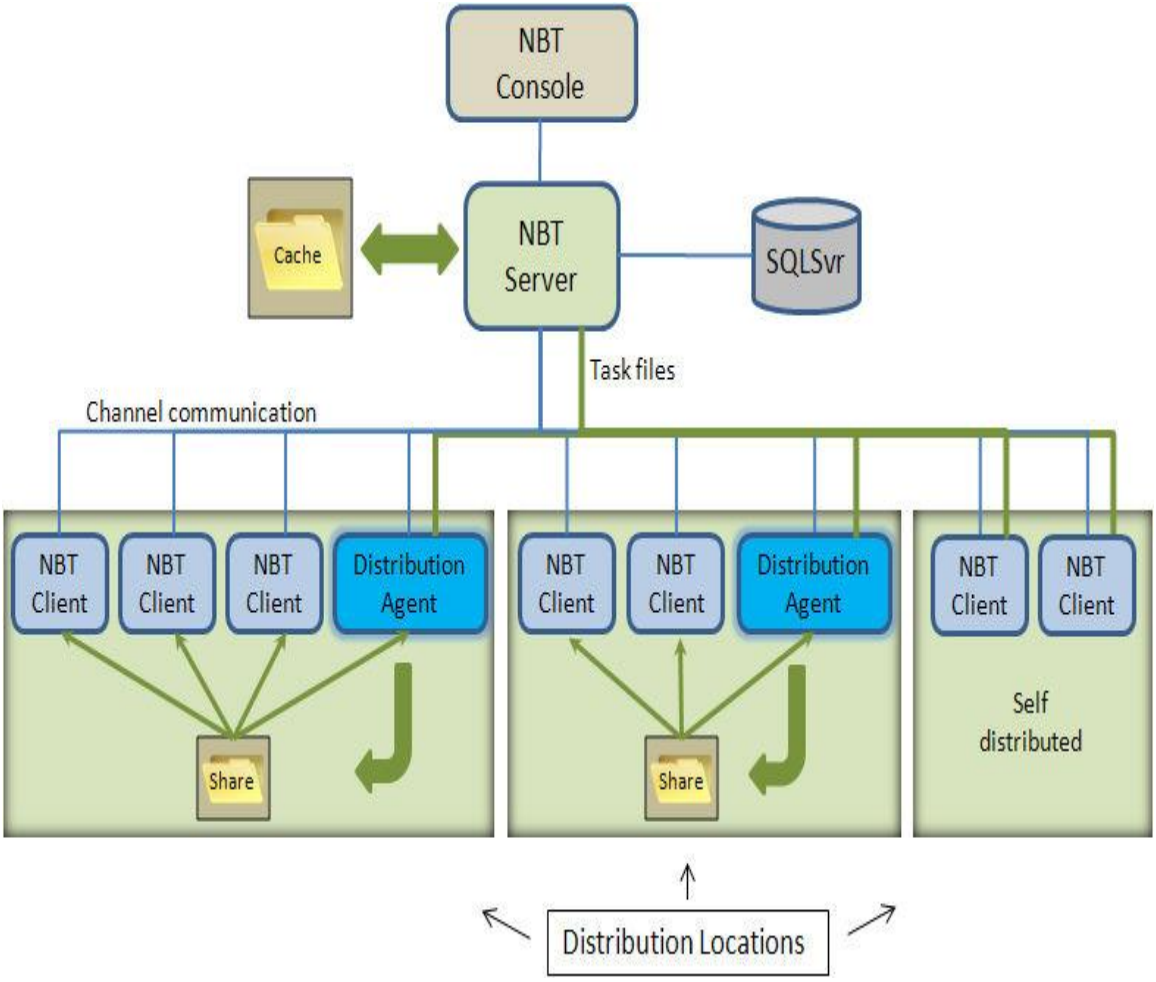
Self-distributed Distribution Locations are also configured through the Distribution Location's properties. While simpler and more appropriate in some circumstances, Self-distributed Distribution Locations impose a heavier load on the network relative to Distribution Locations that employ a single Distribution Agent.

File Transfer

The files and folders associated with Distributed Tasks are always transferred from the Server to a Distribution Location over *secure channel communication.*

Illustration

The figure below illustrates a channel with three Distribution Locations. Two of the Distribution Locations are configured with an Agent and a share. The third is Self-distributed. The two Agents and the two Self-Distributed clients communicate with the Server to keep their files synchronized with the Server's folder. The files are transmitted over channel communication, not via access to a centralized share. Each member computer accesses its local share, not the Server's folder, for files associated with Distributed Tasks.



Benefits of Distributed Tasks

Relative to Non-Distributed Tasks, Distributed Tasks require additional configuration as well as the creation and configuration of Distribution Locations. However, Distributed Tasks provide significant benefits for Deploy users. An installation using only Distributed Tasks provides the following benefits over installations using Non-Distributed Tasks:

1. **Configuration flexibility:** The channel does not require creation and maintenance of a centralized file share. The location specified when creating or modifying a Distributed Task does ***not*** need to reflect the location of the files from the target computers' point of view. This is because a target computer obtains the files from the location defined by its Distribution Location and not the location specified when the Task is created.
2. **Performance:** The Distributed Task's files are downloaded once for each Distribution Location instead of being downloaded by every target computer in the channel. This reduces network load because the downloads occur once per group of computers rather than once per computer.
3. **Security:** A Distributed Task's files are downloaded using secure channel communication.
4. **Internet deployment:** By using secure channel communication for files as well as all other client communication, using Distributed Tasks allows for secure internet installations of Prism Deploy.

Creating Distribution Locations

Follow these steps to create a Distribution Location:

1. On the **Managed** tab, select **Distribution Locations**.
2. On the **Edit** menu, or by right-clicking, select **Create Group...**
3. The Rule Expert opens, just like when creating a User-Defined Configuration Group. Create, name, and edit the rules in the same manner as if you were creating a Configuration Group. The value types available for Distribution Locations include IP address, computer name, computer domain, subnet mask, and others.

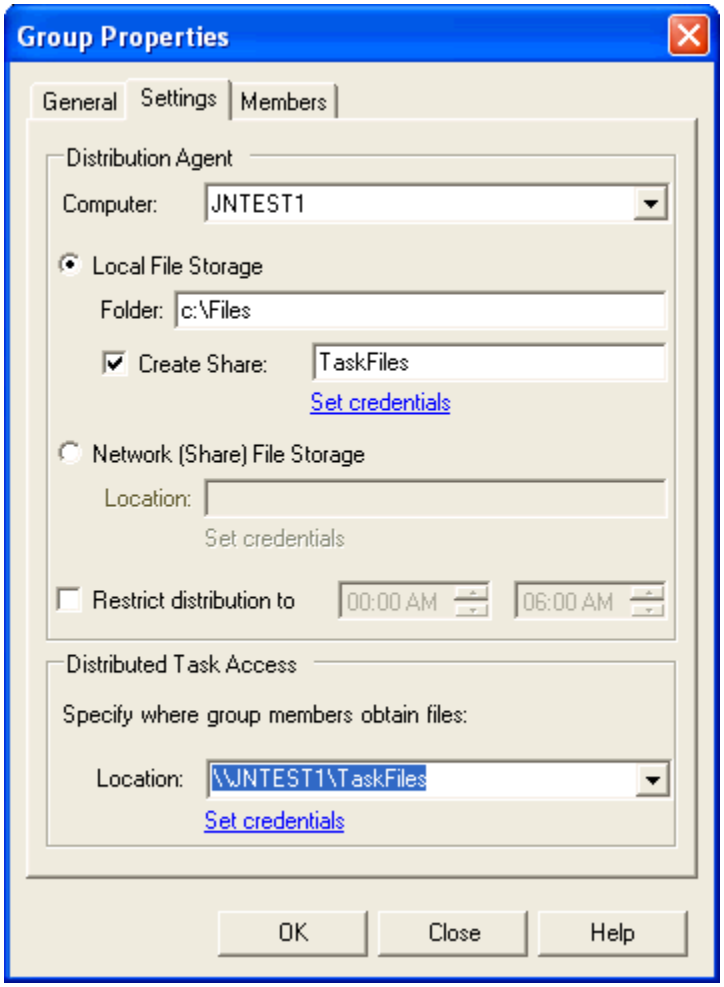
   It is recommended that you choose consistent value types for Distribution Locations so that each managed computer will belong to only one Distribution Location. For example, defining one Distribution Location based on computer names starting with “B” and another Distribution Location based on certain IP address range might cause a computer to belong to both groups.

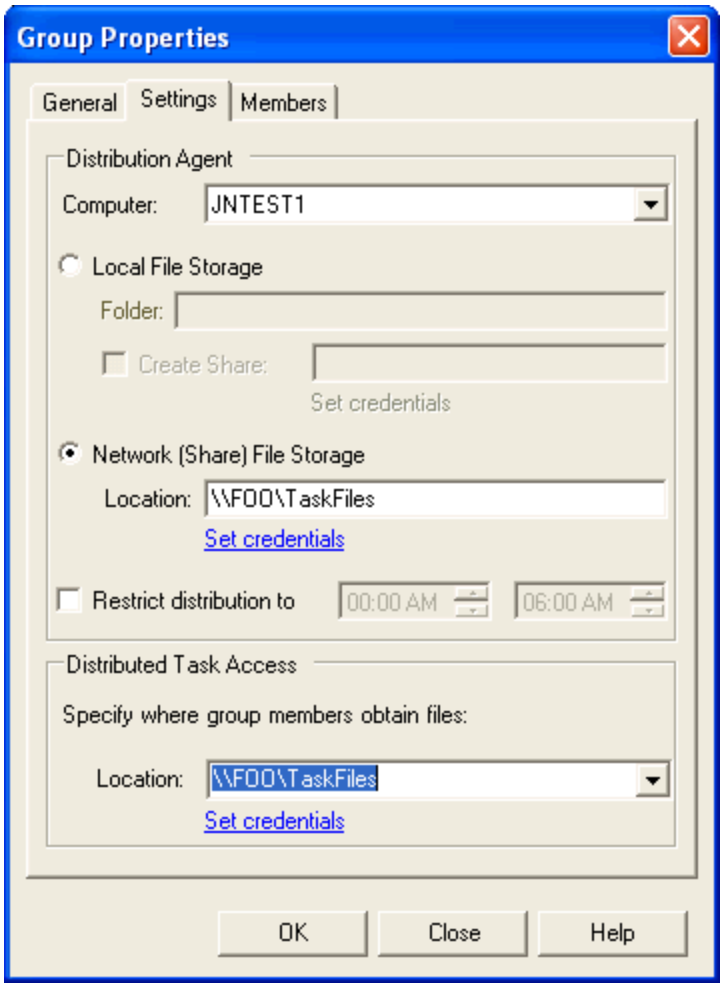
   When you are done defining and naming the rule, press **Finish**.
4. The Distribution Location Settings dialog opens. Use this dialog to configure the Distribution Location, including Distribution Agent selection and share specification

   Note Distribution Location Properties can be revised later through the Distribution Location's **Group Properties | Settings** tab.

5. Press **Finish** to complete creation of the Distribution Location group.

Distribution Location Settings





This dialog defines the parameters necessary for the Distribution Location to make Task files available to its member computers.

The **Distribution Location Settings** dialog appears as the last step when creating a new Distribution Location through the **Create Group** wizard and on the Distribution Location's **Group Properties | Settings** tab.

Distribution Agent

The Distribution Agent panel configures a polling, managed computer to will interact with the Prism Server on behalf of the Distribution Location. It will download all required files for Distributed Tasks from the Prism Server, and it will ensure that this share remains synchronized with the files cached on the Prism Server.

- **Computer**: This value identifies the **Distribution Agent** computer. This computer downloads task files from the Prism Server on behalf of computers in the Distribution Location. From the dropdown list, select a computer that is in the Distribution Location or that can write to the Distribution Location's share.

  Select <**self-distributed**> if you want every computer within the Distribution Location to download all task files rather than using a share. This option will generally be used in special cases such as "catch all" Distribution Locations. For example, it might be used for a Distribution Location containing computers that are not in any domain, such as laptops that are disconnected from the corporate network and do not have access to a common share. With <self-distributed> selected, most other settings in the dialog are disabled.

Notes The <**self-distributed**> option should be used with care, as it will increase the load on the server and network since all computers in that Distribution Location - not just a single Agent - will download all task files.

The drop-down list includes *all* computers in the channel. Use care when selecting to make sure the computer is suitable for the specific Distribution Location being configured.

- **Local File Storage**: With this option selected, Task files are stored locally on the Distribution Agent computer. Member computers will access the Distribution Agent computer to obtain the files.
- **Folder**: Specify a local path on the Distribution Agent computer where it should store downloaded files. If this folder does not exist, it will be created (if possible). Example, `C:\Files`.
- **Create Share**
  - ☐ With this option cleared, the Distribution Agent will not create a share. This would be the case, for example, if the Distribution Agent already has an existing share.
  - ☑ With this option selected, the Distribution Agent will create a share locally.
    - In the text box, enter the name of the share to be created (not including the computer's name). Example: `TaskFiles`.
    - **Set credentials**: Click this link to set authentication credentials for the Distribution Agent to create and write the files to the share.
- **Network (Share) File Storage**: With this option selected, Task files are stored on an existing network share (typically a server) that is accessible to all member computers. In this case, the Distribution Agent writes the files to the existing share and member computers access files from that share.

Note If the network share does not exist, you will need to create it since there is no option to create it automatically.

- **Location**: This specifies an existing share on the network where the Distribution Agent will store files (and, typically, where member computers will obtain the Task files). Include the full path to the share. Example: `\\FOO\TaskFiles`.
- **Set credentials**: Click this link to set authentication credentials for the Distribution Agent to write the files to the existing network share.
- **Restrict distribution to**: This setting defines when the synchronization can occur.
  - ☐ With this option cleared (the default setting), the Distribution Agent will synchronize Task files from the server on every poll.
  - ☑ With this option selected, the Distribution Agent will synchronize Task files only during the specific time window specified. For example, you could specify that files be synchronized only from 1:00 to 3:00 in the morning. Note that this does not impact the agent's normal polling frequency; it only impacts when the agent attempts to synchronize files.

Distributed Task Access

The **Distributed Task Access** panel defines how the other computers within the Distribution Location access the files downloaded by the Distribution Agent.

- **Location**: This specifies where the computers in the Distribution Location will obtain the Task files. Include the full path to the share. You can use the dropdown to select the default share location, depending on your selections in the Distribution Agent panel:
  - **Local File Storage selected and Create Share checked**: Default share is Distribution Agent Computer plus share name. Example: `\\JNTEST1\TaskFiles`.
  - **Local File Storage selected and Create Share unchecked**: Default value is just the Distribution Agent name (e.g., `\\JNTEST1`). To specify the share you must append the actual name of the existing share. Example: `\\JNTEST1\ExistingShare`.
  - **Network File Storage selected**: Default value is simply the value specified for the Network File Storage Location. Example: `\\FOO\TaskFiles`.
- **Set credentials**: Click this link to set authentication credentials to be used by members of the Distribution Location to read files from the network share.

Assigning and Scheduling Tasks for Installation

Deploy Tasks let you quickly and easily install software, run commands, and play scripts on target computers. After setting up a Task that contains the software, files, or changes you want to distribute, assign the Task to a computer or group, select the type of installation (once, repeating, reinstall, or uninstall), then schedule a time for the installation.

For example, drag a Task onto a target (computer or group) in the Channel and Deploy displays the Schedule Task dialog box to lead you through the options for installation. For immediate installation, you only need to drag the Task onto a target and accept the default value on each dialog box of the Scheduling Expert. When you want to schedule a future installation, Deploy lets you set a time, date, and reference point for the schedule. You can also select an option to "wake up a computer" to install a Task.

Assign a Task to a computer or group

Assigning a Task to a target is the first step in using Deploy to install Prism files on target computers, whether directly or through a group. Deploy offers several methods for assigning Tasks. Use any of the following:

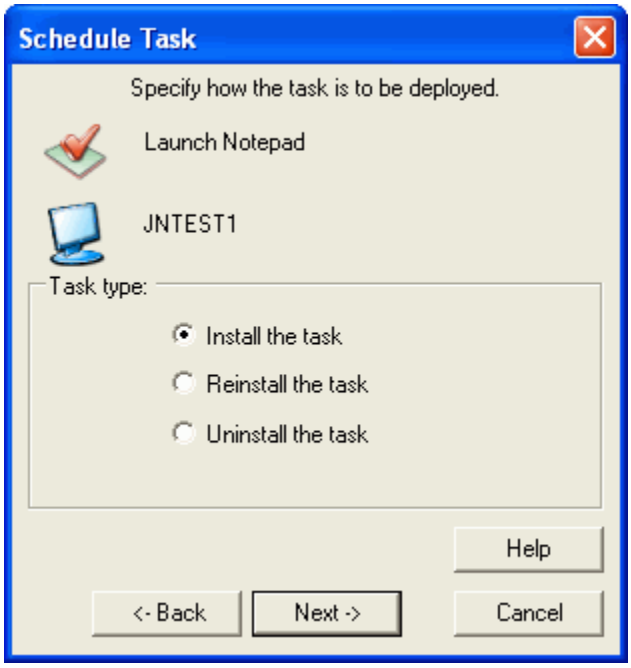
- Highlight the Task and drag it onto a computer or group in the details pane. The Schedule Task dialog box opens, allowing you to select deployment settings.
- Right-click on the name of a Task and choose **Assign Task** from the pop-up menu (or choose **Deployment** | **Assign Task**). The Assign Task dialog box opens to allow you to select targets, then the Schedule Task dialog box opens.
- Highlight the target (computer or computer group) and drag it onto a task in the details pane. The Schedule Task dialog box opens, allowing you to select deployment settings.
- Right-click on the name of a Computer or computer group and choose **Assign Task** from the pop-up menu (or choose **Deployment** | **Assign Task**). The Assign Task dialog box opens to allow you to select Tasks, then the Schedule Task dialog box opens.

Schedule Task installation

After you assign a Task to a computer or group, Deploy displays the Schedule Task dialog box and steps you through the process of setting the schedule and other options. The schedule options cover a variety of situations:

- Scheduling a Task for immediate installation. Accepting the default options will cause the Task to install immediately.
- Scheduling a Task to install at a specified time.
- Scheduling a Task to install based on user logon or system startup.
- Scheduling a Task to install hourly, daily, weekly, or monthly.
- Scheduling a Task to install when a computer enters a group.
- Assigning a Task to the results of a software inventory (if Asset Manager is installed).

Schedule Task: Task Type dialog



Use the **Schedule Task** dialog box to specify whether to install the Task once, install it repeatedly, reinstall the Task, or uninstall the Task. The **Schedule Task** dialog box opens when you assign a Task to a computer, group, or multiple targets, or when you schedule or reinstall a Task Assignment.

Choose an installation option

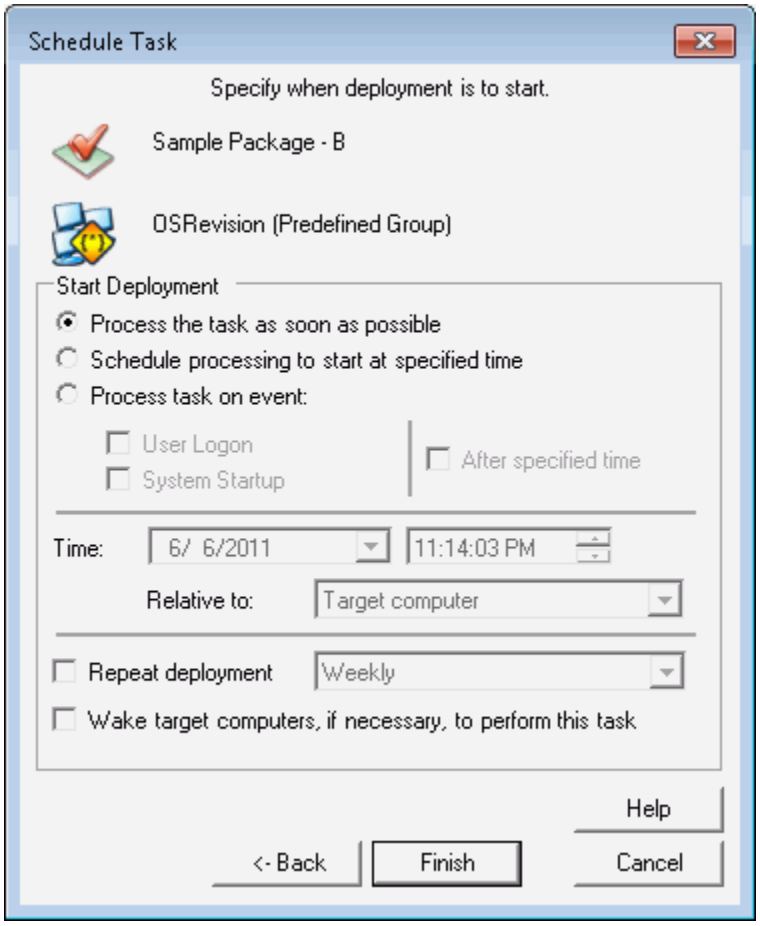
Use the **Task Type** options to install a new Task, schedule a recurring task, reinstall a Task that has already been installed, or uninstall a Task:

- **Install the task once**: Install or execute the Task on the identified target(s) one time.
- **Reinstall the task**: Repeat the installation of the Task on the identified target(s). Use this option to install an updated version of the Task or if the Task was not successfully installed or executed when you installed it earlier.
- **Uninstall the task**: Uninstall all of the changes made with the Task on the target computer(s). If the Task was installed multiple times, changes from each installation are removed.

  This option removes all changes made with a Prism file installed by this Task, even if the Prism file was installed more than once. To reverse a script or command, you must create a file that reverses the command(s).

Note: To uninstall or reverse a script or command, you must create a file that reverses the command(s).

Schedule Task: Start Deployment dialog



Use the **Schedule Task: Start Deployment** dialog box to indicate when to install or reinstall the Task - immediately or at a specific time. When you start the installation at a specific time, you also have the option of choosing a reference point for the start time.

The **Start Deployment** dialog box opens after you choose a schedule type on the Schedule Task dialog box while assigning a Task.

Start deployment

Choose from these deployment options:

- **Process the task as soon as possible**: The Console begins installing the Task as soon as you click **Finish**.

  The installation happens almost immediately on target computers that are available and have the Client Polling option set to one minute (one minute is the default value). If you are in a live or production environment, we strongly recommend that you create a test group and test the Task distribution with this group before you distribute it to a large number of computers.
- **Scheduling the processing to start at**: Install the Task at a specific date and time. After selecting this option, enter the date and time for the installation.

  For the installation to take place at the scheduled time, the target computer must be running and poll at least once before the scheduled installation time. For example, if you schedule a Task for installation on a laptop computer that is not connected to the

network, the Task is installed after the scheduled time, the next time the laptop connects to the network.

- **Process task on event**: Install the Task when a specific event occurs and, optionally, after a specified time. With this option selected, the Wake on LAN option is disabled.

  After selecting this option, select a check box for the type of event that will trigger the Task. Each time this trigger even occurs, the Task is installed or runs on the target computer.

  Options include:

  ☑**User Logon**: With this option selected, the Task is installed when a user logs on to the target computer. For example, create a Task to delete the contents of the C:\WINDOWS\TEMP folder. With this option, the TEMP folder is emptied each time a user logs in to the computer.

  ☑**System Startup**: With this option selected, the Task is installed when the target computer starts up. For example, create a Task to synchronize the time on the target computer with the time on the computer where the Channel Server is located. Then, schedule this Task to run at start up.

  Important: This option should only be used on Tasks that have the **Allow unattended installation** option checked on the Task Properties | Execution tab. This is because there is no user context during system startup, so the Task will not run or may not run correctly. In addition, packages containing both user and system sections should use both events to ensure the user section gets installed.

  If a Prism file is installed with this Task, it must have the **Unattended installation** option set through the **Advanced** button on the **File Properties | Requirements**.

  ☑**After specified time**: With this option selected, the Task only runs on the next event that occurs after the specified time.

Time

Specify the date and time for the installation to start. Use the **Relative to** fields to choose a reference point for the installation date and time:

- **This Computer:** The Task is installed at the scheduled time, using your computer as the reference point.

  For example, set the time to 11:00 p.m. in Denver. The Task is installed at 11:00 p.m. at headquarters in Denver and at 10:00 p.m. at the sales office in Los Angeles.

- **Target Computer:** The Task is installed at the scheduled time, using the target computer as the reference point.

  For example, you are in Seattle and you are installing software for the division in Atlanta. Select this option and set the schedule for 10:00 a.m. The Task is installed at 10:00 a.m. in Atlanta (7:00 a.m. your time).

- **Greenwich mean time:** The Task is installed at the scheduled time with an off-set, using GMT as the reference point. This option lets you set an absolute time.

  For example, set the schedule for 5:00 p.m. on April 1 Greenwich mean time. The Task is installed at:

5:00 p.m. on April 1 in London
12:00 noon on April 1 in New York
9:00 a.m. on April 1 in Los Angeles
1:00 a.m. on April 2 in Tokyo

Repeat Deployment

- ☑ **Repeat deployment**: Check this option to cause the deployment to repeat at a specific time or under specific conditions.
  - If you select **Process the task as soon as possible** or **Scheduling the processing to start at**, the deployment will repeat as specified:
    - **Hourly**, **Daily**, **Weekly**, or **Monthly**: The deployment will repeat at this time interval. For example, use this type of Task to schedule regular backups of the target computers.
    - **When a Computer Joins the Group**: The deployment will repeat whenever a computer joins or rejoins the group. This option is only available if the target of the assignment is group. For example, this option could be used to run a Package Task that removes a non-business application such as Smiley Central toolbar from members of a configuration group that detects the presence of the toolbar. If the end user reinstalls the application, their computer rejoins the configuration group, and the Task reinstalls.
  - If you select **Process task on Event**, the deployment will repeat every time the user logs in and/or the system starts up.
- ☐ **Repeat deployment**: With this option turned off, the deployment will not be repeated.

Wake on LAN

- ☑ **Wake target computers to perform this task**: With this option selected, if the target computer is turned off or shut down when the Task is scheduled for installation, the Channel attempts to wake up the computer and install the Task.

  For this option to work, the target computer must have the appropriate hardware and be correctly configured to support Wake on LAN via Magic Packets™.
- ☐ **Wake target computers to perform this task**: With this option turned off, if the target computer is turned off or shut down when the Channel attempts to install the Task, it is not installed at that time. The Task will be installed the next time the target computer is turned on and its Client contacts the Channel.

Scheduling an Immediate Task Installation

An immediate Task installation will occur immediately after the target computers receive the Task.

To install a Task immediately:

1. Assign the Task to a computer or group. (See Assigning and Scheduling Tasks for Installation.)
2. In the Schedule Task dialog box, select **Install the task** from the list of Task types: Click **Next**.
3. In the Schedule Task: Start Deployment dialog box, select **Process the task as soon as possible**.
4. Optional: If you want the deployment to repeat periodically:
   1. Check **Repeat deployment**.
   2. Select a frequency with which the deployment should occur:

        1. **Hourly, Daily, Weekly, or Monthly**. The Task will be repeated at this frequency based on the current time.
        2. **When a Computer Joins the Group** - The deployment will repeat whenever a computer joins or rejoins the group. This option is only available if the target of the assignment is group.
5. Optional: If you some computers may be powered off and you want to wake them so the Task can be installed, check **Wake target computers**.
6. Click **Finish** to deploy the Task.

The Channel installs the Task on the targets as soon as the target computers receive the Tasks. The results of the installation are displayed in the Deployment Reports.

Scheduling a Time-based Task Installation

You can schedule a Task installation to occur at a specific time and date. For example, where Task installation that may interfere with the work day you can set the installation to occur at a more convenient time.

To install a Task at a specific time and date:

1. Assign the Task to a computer or group. (See Assigning and Scheduling Tasks for Installation.)
2. In the Schedule Task dialog box, select **Install the task** from the list of Task types: Click **Next**.
3. In the Schedule Task: Start Deployment dialog box, select **Schedule processing to start at**.
4. Set the date, time, and time reference for the installation.

> **Note**: For the installation to take place at the scheduled time, the target computer must be connected to the Channel Server. For example, if you schedule a Task for installation on a laptop computer that is only connected to the network occasionally, the Task is installed after the scheduled time, the next time the laptop connects to the network.

5. Optional: If you want the Task deployment to repeat periodically:
    1. Check **Repeat deployment**.
    2. Select a frequency with which the deployment should occur:
        1. **Hourly, Daily, Weekly, or Monthly**. The Task will be repeated at this frequency based on the scheduled time.
        2. **When a Computer Joins the Group** - The deployment will repeat whenever a computer joins or rejoins the group. This option is only available if the target of the assignment is group.
6. Optional: If you are scheduling a Task to install after hours when computers may be powered off, check **Wake target computers**.
7. Click **Finish** to deploy the Task.

The Channel installs the Task on the targets at the time you set. The results of the installation are displayed in the Deployment Reports.

Scheduling an Event-triggered Task

An event-triggered Task is deployed when a specific event occurs. The events that can trigger a deployment are when the system starts up and when the user logs on.

To install a Task based on events:

1. Assign the Task to a computer or group. (See Assigning and Scheduling Tasks for Installation.)

2. In the Schedule Task dialog box, select **Install the task** from the list of Task types: Click **Next**.
3. In the Schedule Task: Start Deployment dialog box, select **Process task on event**.
4. Beneath **Process task on event**, check the events that should trigger deployment of the Task.
    1. **User Logon**: The task will deploy when the user logs on to the system.
    2. **System Startup**: The task will deploy when the system starts up.
5. Optional: If you want the deployment to repeat every time the selected events occur, check **Repeat deployment**.
6. Click **Finish** to deploy the Task.

The Channel installs the Task on the selected targets based on the events you specify. The results of the installation are displayed in the Deployment Reports.

Scheduling a Periodic Task

A periodic Task is deployed repeatedly at a set time interval: hourly, daily, weekly, or monthly.

To set a Task to install periodically:

1. Assign the Task to a computer or group. (See Assigning and Scheduling Tasks for Installation.)
2. In the Schedule Task dialog box, select **Install the Task** from the list of Task types: Click **Next**.
3. In the Schedule Task: Start Deployment dialog box, select when you want the deployment to start - as soon as possible or at a specified time..
4. Check **Repeat deployment**.
5. Select a frequency with which the deployment should occur: **Hourly, Daily, Weekly, or Monthly**. The Task will be repeated at this frequency based on the initial scheduled time.
6. Optional: If you want the Task to install when computers may be powered off, check **Wake target computers**.
7. Click **Finish** to deploy the Task.

The Task deploys at the time you set. After this initial installation, the Task is deployed repeatedly.

Assigning a Repeat Deployment Task Triggered by Group Entry

A task assignment can be configured to install to current members of a group and reinstall when a computer rejoins the group (or enters the group for the first time). For example, this option could be used to run a Package Task that removes a non-business application such as Smiley Central toolbar from members of a configuration group that detects the presence of the toolbar. If the end user reinstalls the application, their computer rejoins the configuration group, and the Task reinstalls.

Note: Repeat deployment task assignments based on group entry is only applicable where the target is a group.

To set a Task to install when a computer enters a group:

1. Assign the Task to a group. (See Assigning and Scheduling Tasks for Installation.)
2. In the Schedule Task dialog box, select **Install the Task** from the list of Task types: Click **Next**.
3. In the Schedule Task: Start Deployment dialog box, select when you want the initial deployment to start - as soon as possible or at a specified time.
4. Check the **Repeat deployment** option.
5. In the drop-down list, select **When a Computer Joins the Group**.

The initial Task deploys at the time you set to the current members of the group. Thereafter, it will occur whenever a computer joins or rejoins the group.

Copying Task Assignments

Task assignments for a Computer or group can be copied to another computer or group. This creates new assignments on the targets without having to set the deployment parameters via the Schedule Task dialog box. The new assignments will have the same settings as the original assignments.

To copy Task assignments from a computer or group to another target:

1. From the **Managed** or **Configuration Groups** tab, select the computer or group whose assignments you wish to copy.
2. Open the Assignments tab.
3. Select the assignments to be copied.
4. Drag them to the desired target.

All scheduling options defined on the original assignments apply to the new assignments..

Changing the Schedule for a Task

Changing the Schedule for a Task

You can reschedule or change the time and date for installing a Task at any time. The procedure for changing the schedule depends on the number of targets affected.

- Rescheduling the installation for all of the targets receiving the Task is similar to scheduling a new Task. For details, see Changing the Schedule Before a Task is Installed.
- Changing the schedule for only one target or a subset of the original recipients requires reassigning the Task to the selected target(s) and setting the appropriate time. For details, see Changing the Schedule for a Subset of the Assigned Targets.

Changing the Schedule Before a Task is Installed

The schedule for an assigned Task can be changed at any time. If you set a specific day and time for the installation, return to the Schedule Task dialog box to change the time or date. This procedure is outlined below.

To change the schedule for a Task:

1. Highlight the name of the Task assignment on the Assignments tab, either from the Managed tab or Tasks tab.
2. Open the Schedule Task dialog box by doing one of the following:
   - Right-click on the Task assignment and choose **Schedule** from the pop-up menu.
   - With Task assignment highlighted, choose **Deployment** | **Schedule**.
3. Complete the scheduling through the wizard, starting with the Schedule Task dialog box.

The Task is installed for all of the targets it is assigned to, according to the new schedule.

Changing the Schedule for a Subset of the Assigned Targets

After scheduling a Task for a group, you can install the Task ahead of time for only part of the original group. To schedule the installation at an earlier time for only some of the targets scheduled to receive it:

1. Re-assign the Task to the selected target(s) by doing one of the following:

- Drag the Task onto the target(s).
- Right-click on the Task assignment and select **Schedule** from the pop-up menu.

  Even though each target is already scheduled to receive the Task, you are assigning the Task as if it were a new Task. (See Assigning and Scheduling Tasks for Installation.)

2. Complete the scheduling through the wizard, starting with the Schedule Task dialog box.

When you look at the target in the Console main window, the Assignments tab shows both the original schedule for the Task, and another instance of the same Task with the new schedule. Even though the Task is scheduled twice, it is installed on the target only once, at the earliest scheduled time.

Reinstalling or Repairing a Task

Any number of circumstances can result in the need to reinstall a Task. For example, the installation failed or you decided to change the original Prism file. When a change is needed or an error interferes with the Task installation, you can reinstall the Task in just a few simple steps.

In most cases, the installation status for an individual target and the Deployment reports provide information about why the installation was unsuccessful.

Here are some sample situations and how to resolve them:

- A user deleted a critical file after the installation and it needs to be reinstalled. Reinstall the Task for this application as described in Reinstalling a Task.
- The Prism file installed with the Task asks the user to respond to a variable prompt. Originally, the installation was scheduled for midnight, when most users were logged off. Reinstall the Task during work hours as described in Reinstalling a Task.
- A power-outage in one building interfered with installation for a portion of the targets scheduled to receive the Task. When power returns, you want to reinstall the Task, but you do not have the computers identified by location. Reinstall the Task for this application as described in Reinstalling a Task for Computers that Did Not Receive It.
- After the Task was installed on the first 10 computers, you discovered an error in the Task itself. Repair and reinstall an updated Task as described in Reinstalling a Repaired Task.

Reinstalling a Task

A Task can be reinstalled for targets that received the Task earlier, but need the Task installed again. For example, if the installed software becomes corrupted, reinstall the Task to restore the software.

The Prism Console provides several ways to reinstall a Task. For example, you can use the Assign Task wizard (see Assigning and Scheduling Tasks for Installation), or you can select Task Assignments on the Assignments tab and use the Schedule or Reinstall menu options. The Reinstall menu option approach is discussed below.

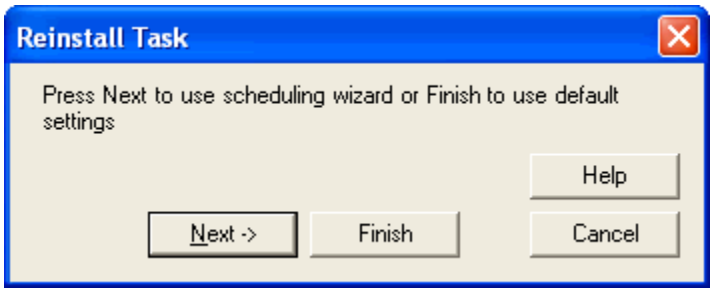
To reinstall one or more Tasks to one or more targets, follow these steps:

1. Select one of these three options - your choice depends on the number and type of targets as well as the number of Tasks:
    1. To reinstall to a specific computer or organizational group, open the Managed tab and select the desired computer or group.
    2. To reinstall to a specific configuration group, open the Configuration Groups tab and select the desired configuration group.

3. To reinstall to some or all the targets to which the Tasks are assigned, open the Tasks tab and select the desired Tasks (Task, Task group, or All Tasks).

2. Select the Assignments tab, and then select the assignments corresponding to the Tasks you want to reinstall.
3. Open the Reinstall Task dialog by either:
    1. Right-clicking on the name of the Task and choose **Reinstall** from the pop-up menu; or
    2. Selecting **Deployment** | **Reinstall** from the menu.
4. To reinstall immediately, click **Finish** and the Task will be reinstalled immediately. To customize settings, continue with the next step.
5. Click **Next** to customize reinstallation settings.
6. In the Schedule Task (Reinstall) dialog, select your reinstallation options.
7. Click **Next**.
8. In the Start Deployment dialog box, set a schedule for installing the Task.
9. Click **Finish** to reinstall the Task.

Reinstall Task dialog



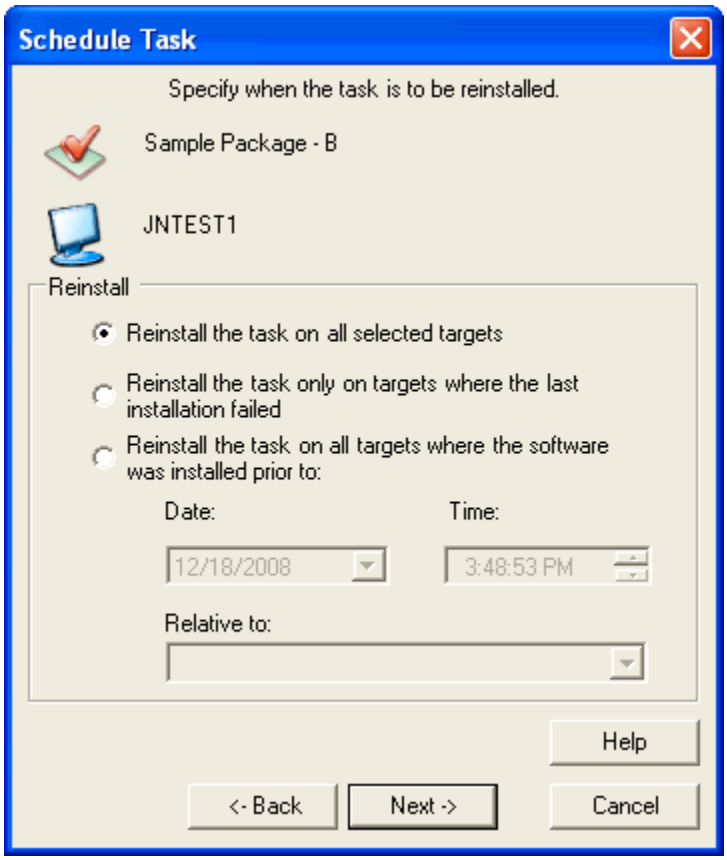
This dialog box opens when you select **Reinstall**, either by right-clicking on one or more Task assignments or selecting it from the **Deployment** menu.

Select **Next** to open the Schedule Task (Reinstall) dialog. This allows you to select specific settings for the reinstallation of the task(s), including the conditions for reinstallation (for example, that the reinstall should apply only on targets where the last installation failed) and when it should be applied.

Select **Finish** to reinstall immediately.

If you want to modify any of the above values, press **Next** instead of **Finish**.

Schedule Task (Reinstall) dialog



Use the **Schedule Task: Reinstall** dialog box to choose the type of reinstall to perform. The best choice depends on the number of targets receiving the task and on the situation requiring a reinstallation.

Choose from these options:

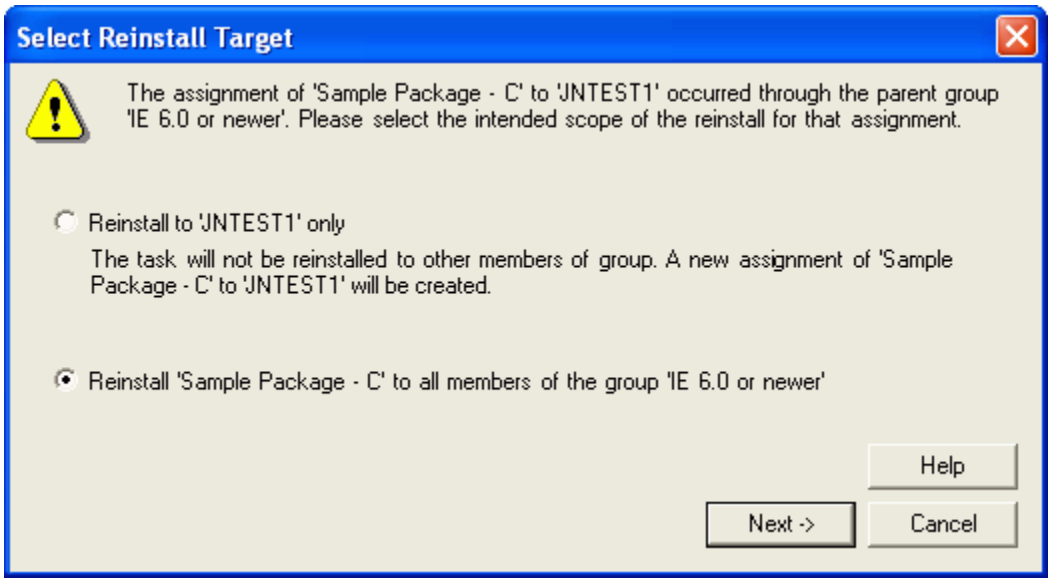
- **Reinstall the task on all selected targets**: Reinstall the Task on all of the targets highlighted in the Console main window. If a group is selected as the target, all targets in that group receive the Task.
- **Reinstall the task only on targets where the last installation failed**: Before reinstalling the Task, the Console identifies the targets where the same Task failed to install during the previous installation. If a group is selected as the target, the Task is reinstalled only for those members that failed to receive it the last time.
- **Reinstall the task were the software was installed prior to**: Before reinstalling the Task, Deploy identifies only the targets that received the same Task before the specified time.

  If a group is selected as the target, the Task is installed only for members of the group that received this Task before the time and date you enter. Targets that received the Task after the specified time do not receive the reinstall.

  After selecting this option, specify the reference time and date in the lower part of the dialog box. See Reinstalling a Repaired Task for an example.

Click **Next** to open the Start Deployment dialog, or **Back** to return to the previous dialog.

Select Reinstall Target dialog



This dialog box appears when you select one or more task assignments and choose the **Reinstall** menu item (right-clicking on the assignment or selecting the **Deployment** | **Reinstall** menu item), under the following conditions*:*

1. The Managed tab or Configuration Groups tab is selected;
2. A single computer is selected;
3. One or more of the selected assignments in the Assignments tab was assigned through a group.

When a single computer is selected, the Assignments tab displays all the task assignments for that computer. This includes direct assignments as well as assignments that occurred through a group to which that computer belongs. For the selected group assignments, this dialog allows you to choose whether the reinstall should apply to the selected computer only or all members of the assigned through groups. Each option has distinctly different consequences.

- **Reinstall to selected computer only**: This option restricts task reinstallation to the selected computer only. The tasks associated with group assignments will be reinstalled *only to the selected computer*. The reinstallation will not apply to other members of the assigned through groups.

Note This option will cause one new task assignment to be created for each selected group assignment.

- **Reinstall to all members of any Assigned Through groups**: This option will reinstall each task to the assigned through group defined within each group assignment. For each selected group assignment, the reinstall will apply to *all members of the assigned through group*, not just the selected computer.

Note These options pertain only to the group assignments; the choice you make has no impact on the direct assignments for that computer.

Press **Next** to proceed to the Reinstall Task dialog where scheduling parameters can be specified.

Reinstalling a Task for Computers that Did Not Receive It

If only part of a group to which a Task was assigned successfully received the Task, you can reinstall the Task for the entire group without identifying the individual computers that did not

receive it originally. Only the computers that did not receive it will reinstall the Task. To reinstall one or more Tasks to part of a group, follow these steps:

1. Select one of these two options - your choice depends on the type of group:
    1. To reinstall to a specific organizational group, open the Managed tab and select the desired group.
    2. To reinstall to a specific configuration group, open the Configuration Groups tab and select the desired configuration group.
2. Select the Assignments tab, and then select the assignment corresponding to the Task you want to reinstall.
3. Open the Reinstall Task dialog by either:
    1. Right-clicking on the name of the Task and choose **Reinstall** from the pop-up menu; or
    2. Selecting **Deployment** | **Reinstall** from the menu.
4. Click **Next**.
5. In the Schedule Task (Reinstall) dialog box, choose **Reinstall the task only on targets where the last installation failed**.
6. Click **Next**.
7. In the Start Deployment dialog box, set a schedule for installing the Task.
8. Click **Finish** to reinstall the Task.

Reinstalling a Repaired Task

Use the reinstallation option if you need to reinstall an updated or repaired Task based on the time the Task was repaired. For example, you scheduled a software update for all of the targets in the Channel. At 3:00 P.M., after half of the targets received the Task, you discovered an error and repaired it immediately. The targets that received the Task after 3:00 P.M. are fine. The targets that received the Task before 3:00 P.M. need the updated Task.

To reinstall the Task for the targets that did not receive the repaired Task:

1. As soon as you discover the problem, disable the Task. (See Temporarily Disabling Tasks.)
2. After repairing the Task, resume the installation by clearing the **Disable** check box for the Task on the Task Properties | General tab. The Channel resumes the installation, installing the repaired Task on the remaining targets.
3. Open the Tasks tab and select the desired Task.
4. Select the Assignments tab, and then select all of the Task's assignments.

    Note You can reassign it to all of the original targets, including those that later received the repaired Task.

5. Open the Reinstall Task dialog by either:
    1. Right-clicking on the name of the Task and choose **Reinstall** from the pop-up menu; or
    2. Selecting **Deployment** | **Reinstall** from the menu.
6. Click **Next**.
7. In the Schedule Task (Reinstall) dialog box, choose the **Reinstall the task on all targets where the software was installed prior to** option.
8. In the lower part of the dialog box, set the date, time, and reference point for the installation.

    In our example, set the time for 3:00 p.m.
9. Click **Next**.
10. In the Start Deployment dialog box, set a schedule for installing the Task.
11. If applicable, set the date, time, and reference point for the installation:
12. Click **Finish** to reinstall the Task.

The Task is reinstalled on all targets that originally received the Task before the set time, 3:00 P.M. in our example.

Unscheduling a Task

A scheduled Task can be removed from the schedule at any time. When a Task is unscheduled, the Task itself and all of its settings remain part of the Channel, but it is removed from the schedule and the record of this assignment is removed (for targets where it has already been installed).

To unschedule a Task or remove it from the schedule:

1. With the Assignments tab open in the details pane, highlight the name of the target originally scheduled to receive the Task.
2. Right-click and select **Delete** from the pop-up menu.
3. Deploy asks if you want to delete the assignment. Click **Yes**.

The Task is removed from the selected target and will not be installed for that target.

Note: Removing or deleting a Task from the schedule or from the Channel is not the same as uninstalling or removing the changes made on a target computer. To uninstall a Task or reverse the changes made on a computer, see Uninstalling Changes with Deploy.

Task Properties

General tab (Task Properties)

The **Task Properties** | **General** tab displays information about the Task currently highlighted in the main window.

This dialog box is displayed when you right-click on the name of one or more Tasks and choose **Properties** from the pop-up menu.

Other tabs on the Task Properties dialog include:

- Command tab - allows you to define the command
- Execution tab - allows you to define the account the Task should run under
- Uninstall tab - allows you to define how the Task is to be uninstalled
- Assignments tab - shows the current assignments of this Task
- Prerequisites tab - allows you to define pre-conditions that must be met for a Task to deploy
- Notifications tab - allows you to enable email notifications when deployments fail
- Security tab - allows you to see the Roles that have been secured from this Task.

If you select multiple Tasks and open the **Properties** dialog, only the **General**, **Execution**, **Prerequisites**, and **Notifications** tabs are visible.

General Fields

The information on the General tab includes:

-  Descriptive name of the Task. The secured task icon  , if present, indicates that the user is not allowed to modify the task or assign it.
- **Description**: Enter a description of the Task or the changes that it installs.
- **Category**: Lists the type of Task.
- **Installation Order**: This defines the relative installation order among tasks:
  - **System default**: Tasks set to this value will install in a system-defined order.
  - **Priority**: Any task with a Priority value set will be installed in priority order, where tasks with lower priority values will be installed before tasks with higher values. Setting these values allows you to control the installation order of a group of tasks.
    - Tasks with the same priority are installed in a system-defined order.
    - Tasks set to "System default" always execute after tasks with priority values.
- **Task ID** and **Numeric ID**: Internal identifiers.

Note These fields are disabled when multiple Tasks are selected.

Task Type

- **Package**: The Task installs a Prism file.
- **Command**: The Task is an Internet address or it runs a third-party command or script. The command can be any command that will run on the target computer or a script that is not a Prism script.

- **Script**: The Task plays a Prism script (.PTS). The script can install one or more Prism files and run Prism commands. (For information on Prism scripts and commands, see Scripting Prism Tasks.)
- **Windows Installer (MSI) package**: The Task installs a Windows Installer (MSI) package.

You can change the file name or the *Task Type* at any time. If you originally set up the Task as a Package and later need the complexity of a script, change the option on this tab to Script. Then go to the Command tab (in this dialog box) to change the file used to install the Task.

Important: If you also want the change to be installed on computers where the Task was installed before the change, you must reinstall the Task.

Note: The Task Type options are disabled when multiple Tasks are selected.

Disable

Indicates whether the Task is active or inactive. A disabled Task cannot be installed on target computers, even if it has been scheduled.

☐ **Disable**: With this option turned off, the Task is active and can be installed on targets.

☑ **Disable**: With this option selected, the Task is disabled and will not be installed on targets even if it has been scheduled, but it remains part of the Channel. Use this option to temporarily suspend deployment of the Task without having to change the existing schedule.

 When a Task is disabled, its icon is grayed in the Console main window.

This option is enabled even when multiple Tasks are selected. If only some of the Tasks are disabled, the checkbox will appear indeterminate (gray) indicating a mixture of values.

Changing a Task Location or Name

The name or location of the file installed by a Task can be changed at any time. This information is displayed on the Task Properties | Command tab.

For example:

- If you move a Prism file that is associated with a Task to a different computer for better performance, change the location for the file on this tab.
- If the Task becomes more complex and you need a Deploy script rather than a single Prism file, write the script, then change the name on this tab. The name of the Task can remain unchanged.

Change the Task File Name or Location

To change the name of the file used for a Task or its location:

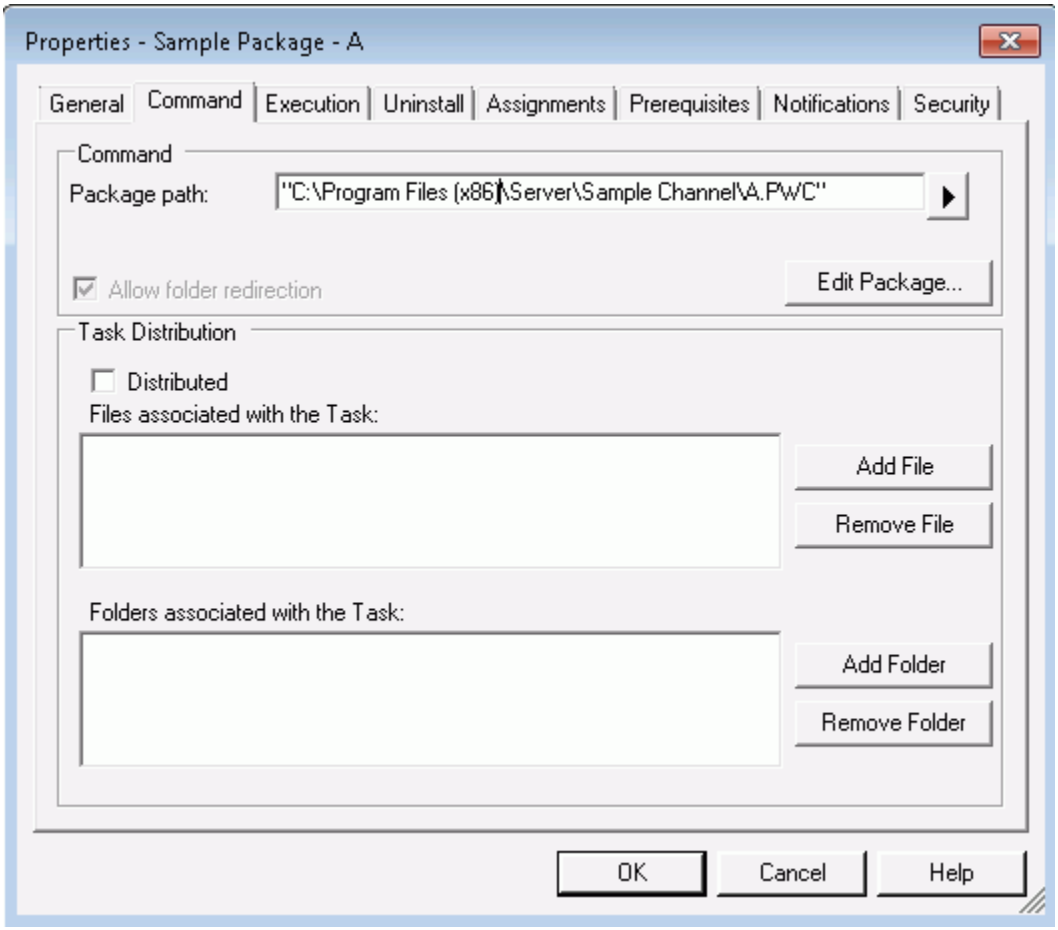
1. Right-click on the Task name in the tree view.
2. Select **Properties** from the pop-up menu.
3. On the Task Properties dialog, go to the Command tab.

   The text box near the top of the tab displays the location and name of the Prism file, Prism script file, or command file.

4. Type a new location or file name in the text box or click  to browse for the file.

The location or file name is updated when you click **OK**.

Command Tab (Task Properties)



Use the options on this tab to fine-tune how the command is run on a managed computer. This tab is displayed when you open the Task Properties and click on the Command tab.

The particular options available and/or enabled depend on the Task type.

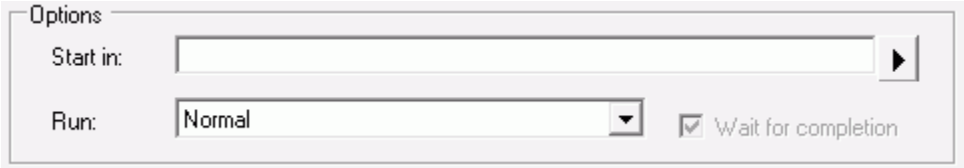
Command Panel

- **Package Path/Script Path/Command Line**: This text box displays the location and name of a file (Prism file, script file, command file). If the name or location contains spaces, enclose the entire path in quotation marks. You can enter the file name or path directly into the text box, or click  to browse for the file.
- **MSI command**: This text box displays the MSI command, including the path of the MSI file and main options.
- **Command Options**: This text box shows the command line options. Command line options are available for Commands and Windows Installer (MSI) Tasks only. For Windows Installer (MSI) Tasks, the initial set of options displayed depend on how the MSI Task was created.
- **Edit Package**: Click this button to edit the Package in the Editor.
- **Edit Script**: Click this button to edit the Script in a text editor such as Notepad.
- **Allow folder redirection**: This option applies only to Command and Script Tasks. It is disabled for Package Tasks because packages explicitly reference 32- and 64-bit folders.

☑ **Allow folder redirection**: With this option selected, the command path will automatically redirected on 64-bit systems to the 32-bit location for the command.

☐ **Allow folder redirection**: With this option cleared, the command path will be not be redirected on a 64-bit system. In this case, it is possible that on a 32-bit machine, the 32-bit version will be executed while on a 64-bit system, the 64-bit version will be executed.

Options Panel



These options are available only for Command Tasks:

- **Start in**: The working folder for the program launched by a command. This field can be left blank if the command does not require a working folder.
- **Run**: Choose an option from the list for controlling the appearance of the program's interface when a command is run:
  - **Normal:** The application window is displayed at the default size.
  - **Maximized**: The application window is maximized.
  - **Minimized**: The application window is minimized.
  - **Hidden**: The application window and any error messages are hidden from view. Use this option with caution. If the installation encounters an error, the application may need to be visible if it requires user input.
- **Wait for Completion**:

☑ **Wait for Completion**: With this option selected, the Client waits until the current command has finished running before beginning any other operation.

☐ **Wait for Completion**: With this option cleared, the Client may begin another operation before the current command has finished running.

Note This option is disabled if the Run As option (Execution tab) is set to **Run as this account**, which will force the client to wait until the current command has finished.

Task Distribution Panel

Distributed

Indicates whether the Task's files should be Distributed. All Task Types can be Distributed.

☐ **Distributed**: With this option turned off, the Task becomes a Non-Distributed Task. The files required for a Non-Distributed Task must be stored in a location that is available to all target computers, such as a file share. The location of the file(s) required for the Task is relative to the target computers.

☑ **Distributed**: With this option selected, the Task becomes a Distributed Task. The files and folders required for a Distributed Task are cached by the Prism server and synchronized from the server out to all Distribution Locations in the channel. In contrast to Non-Distributed Tasks, the location of the files and folders required for a Distributed Task do *not* need to be relative to the target computer. Instead, the location is specified by each target computer's Distribution Location.

Task Files and Folders

The two lists display the files and folders associated with the **Distributed Task** that will be automatically synchronized out to all Distribution Locations, where they can subsequently be accessed by target computers. The file list includes the command file automatically. Additional files and folders can be added to the lists.

For example, if a Script Task is marked as Distributed and it references several files, these files may need to be added to this list so that computers in Distribution Locations can access them. This will cause them to be downloaded to the Distribution Location. When the script runs, those files will then be found in the Distribution Location's share.

Note To distribute a Script Task that references other files, the script must be edited by substituting `%DistShare%` as the path so that all references to those files can be resolved.

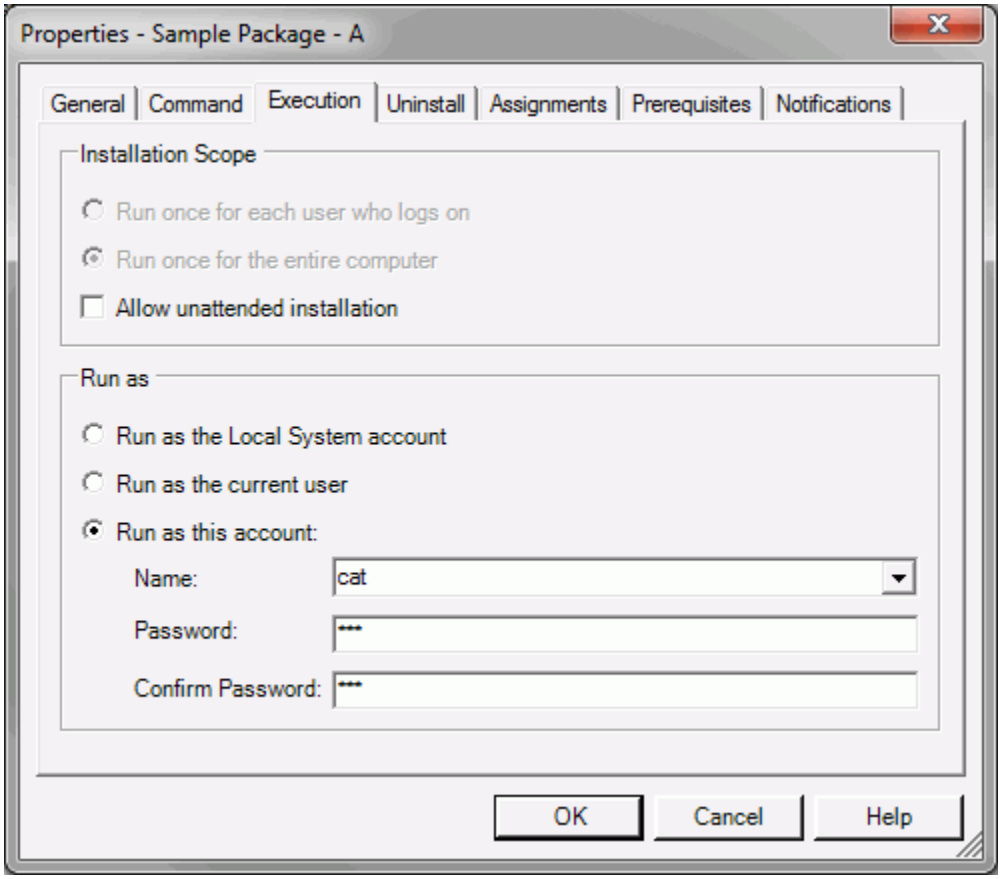
Manipulate the lists through the **Add** and **Remove** buttons (if the Task is not marked as **Distributed**, the lists are empty and the buttons are disabled):

- **Add File**: Opens a dialog that allows you to select files to be synchronized. The command file is automatically included in the list of files, so this does not need to be added separately.
- **Remove File**: Removes the selected file from the list.
- **Add Folder**: Opens a dialog that allows you to select folders to be synchronized. You can either browse to the folder's location or enter the folder name directly in the text box. Each folder - including all of its contained files and sub-folders - in the list will be synchronized out to the Distribution Locations.

  Note You cannot add the folder containing the command file to the list of folders to be synchronized. This can be inconvenient in cases where there is a single installation folder containing an installation command and multiple associated files and folders. A potential workaround is to create a batch file, at the same level as the installation folder, that invokes the installation command. In this case, the file list contains the batch file (instead of the installation command) and the folders list contains the single installation folder.

- **Remove Folder**: Removes the selected folder from the list.

Execution tab (Task Properties)



Use this tab to specify the users and account the Task is run under on the target computer.

This tab is displayed when you open the Task Properties and click on the Execution tab.

Note The options available on the Execution tab vary depending on the type of task and whether it is Distributed.

Installation Scope

- **Run once for each user who logs on**: When each user logs on to the target computer, the Task is run. Use this option for a command or script that includes settings that may be unique to each user. This option is available only for Commands and Scripts.
- **Run once for the entire computer**: The Task is run once, installing changes that apply to all users on the computer. This option is available only for Commands and Scripts.

☑**Allow unattended installation**: With this option selected, the Task can be installed even if nobody is logged on to the target computer. If no one is logged on to the computer and **Run as the current user** option is specified under **Run As** (see below), the Task is installed under the Local System account on the target computer.

If the Prism file is set up to prompt the user and no user is logged on, the file is installed without displaying the prompts. For example, if the Prism file chosen for the Task has a before prompt set up through the **File Properties | Messages** tab, this prompt will be ignored when the file is installed.

Use extreme care in defining other Task properties, including those that are part of the original Prism file or script. If there is not a user present when the Task is installed and user input is required, an error may occur.

☐ **Allow unattended installation**: With this option cleared, the Task will not run until a user logs in to the target computer.

If multiple tasks are selected, only the **Allow unattended installation** is enabled. If only some of the Tasks have this setting checked, the checkbox appears indeterminate (gray) indicating multiple settings.

Run As

In the **Run As** area of the tab, select the account on the target computer under which the Task is installed.

You can select multiple tasks and modify their Run As settings at one time. If the selected Tasks have different Run As settings, no Run As option will appear selected.

- **Run as the Local System account**: The Task is run under the Local System account, so that a user does not need to be logged on at the time of installation.

Important The Local System account does not usually have access to network resources.

- **Run as the current user**: The Task is installed under one of the following:
  - the account of the user logged on at the time of installation
  - the system account, if a) the machine is unattended and b) the **Allow unattended installation** option on this dialog box is selected.

    If the Task does not require special privileges, this option can be used.
- **Run as this account**: Specify the account under which the Task will be installed on the target computer. Enter the account user name and password in the text boxes. The account name can be listed in any of the following formats:
- **UPN (User Principal Name):** sue@YOURCOMPANY.COM
- **NT4 domain name\account name**: YourCompany\Sue
- **Local user name**: Administrator

For convenience, the Name dropdown list under "Run as this account" is pre-filled with all user credentials associated with all Tasks in the channel. You can also enter a new name and/or password manually.

For your security, the account information and password you specify is always encrypted.

Note For Command Tasks, this option forces the **Wait for completion** option to be checked on the Command tab.

Note For installing most Prism files, you do not need to worry about whether the user has permissions to install software on the target computer. However, you may need to specify an account with sufficient permissions in the following cases:

- installing a Prism file from a network volume to which the current user does not have access

- playing a Prism script from a network volume to which the current user does not have access

- running a command that requires a greater level of permission than that of the current user

Modifying Passwords for Multiple Tasks

If you have multiple tasks that use the same user credentials under "Run as this account", and you need to change the user's password, you can modify their password once for all Tasks instead of modifying each Task individually. To do this, multi-select those Tasks, open the **Properties** dialog, and select the Execution tab. Under "Run as this account", enter the new password.

Modifying Credentials for Multiple Tasks

If you have multiple tasks that use different Run As settings or use different user credentials under "Run as this account", and you want to change them all to a specific user's credentials, you can modify the credentials once for all Tasks instead of once per each Task. To do this, multi-select those Tasks, open the **Properties** dialog, and select the Execution tab. Select "Run as this account", select the desired user name from the dropdown list (or enter a new user name), and enter the password.

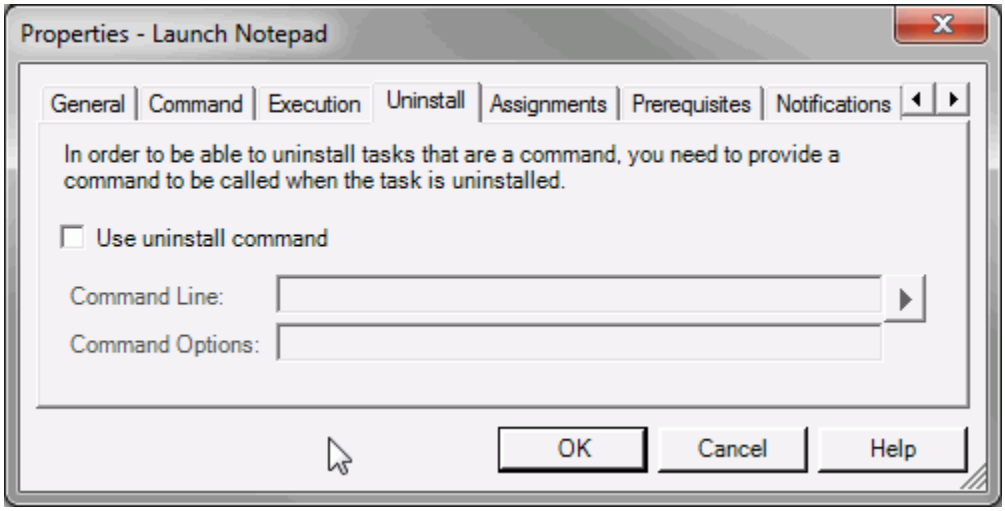
How the Task types respond to the Run As options

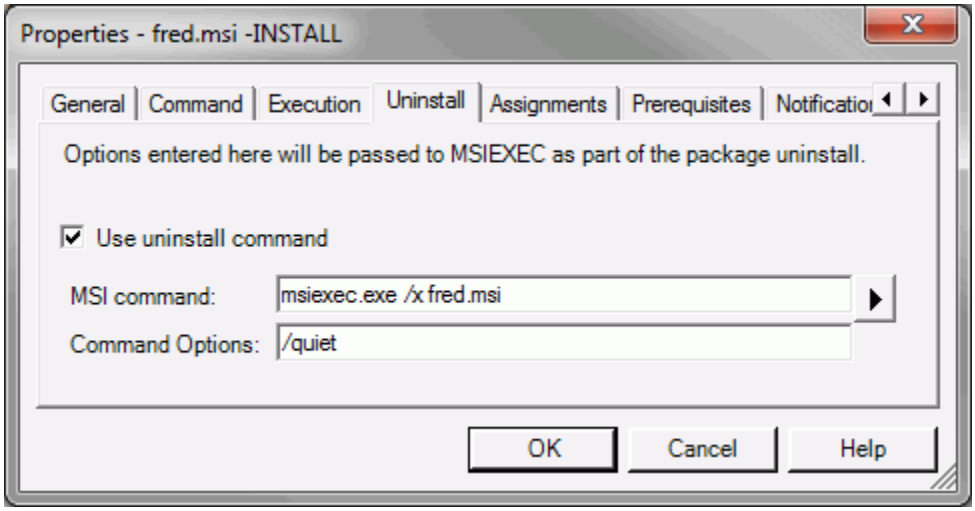
- **Package:** The account you specify is used to read the Prism file being installed.
- **Script:** The account you specify is the one under which the script file is opened. Any UNC-based files used by the script are accessed under this account, and any commands run by the script are run under this account.

  For example, you want to deploy the script \\server\volume\scriptfile.pts, but users on the target computers do not have access to this location. In this case, specify an account that does have access to this location.
- **Command:** The command run by the Task operates under the user account you specify.

  For example, if you specify **Run as current user** and the command opens Notepad, you may not have access to network folders or drives from within Notepad. However, running the same command under an administrator's account will give you access to network resources that are unavailable to the current user.
- **Windows Installer Package (MSI):** Windows Installer packages are always installed for the entire machine, using elevated privileges. The specified account will be used during the installation. If an .MSI file is located on the network, it should either be located on a null session share or the task should be run using an account that has access to the share.

Uninstall tab (Task Properties)





On this tab, specify how to uninstall or reverse the changes made when the Task was installed. This tab is displayed when you open the Task Properties and click on the Uninstall tab.

The options enabled depend on the Task Type:

- **Packages**: Uninstall functionality is built-in to Package Tasks, so all options on this tab are disabled. Deploy automatically uninstalls Packages when you select the Uninstall option on the Schedule Task dialog.
- **Commands**: All options are available for Command Tasks.
- **Scripts**: All options are available for Script Tasks.
- **Windows Installer Packages (MSI)**: All options are available for MSI Tasks. For MSI installation (/I) tasks, the MSI Command for uninstall is automatically set to use the /x option.

"Use Uninstall" Checkbox

This option is available for Script and Command Tasks only.

☑**Use uninstall script/command**: Select this option if you want to specify an uninstall command or script for uninstalling the Task. With this box selected, the Command or Script text box is active.

☐**Use uninstall script/command**: Clear this check box if the Task does not require an uninstall command or script.

Command Line

This text box displays the location and name of a command or script file used for uninstalling the Task. If the name or location contains spaces, enclose the entire path in quotation marks. You can type the name directly into the text box or click  to browse for the file.

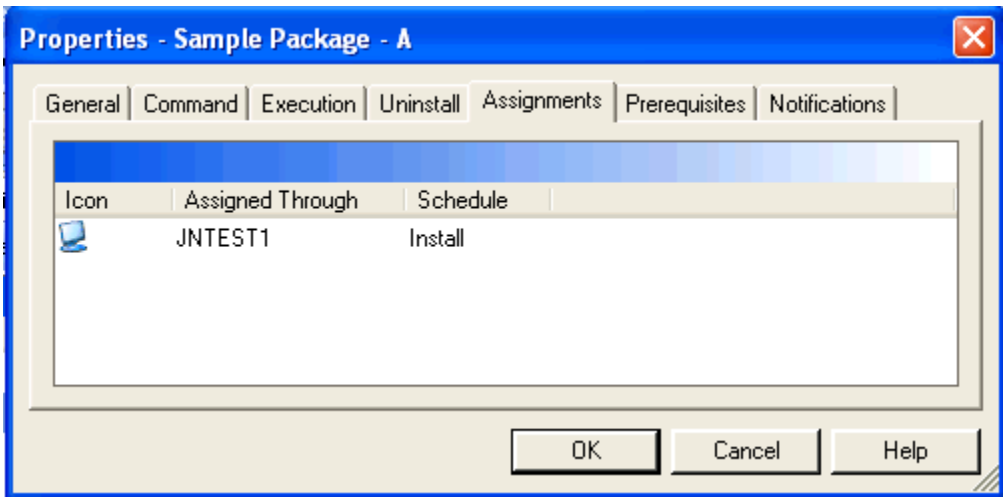
Note The Uninstall runs under the account and start in folder that were selected originally for the Task. If these settings are not appropriate for uninstalling the Task, create a separate Task for uninstalling. Then, within the uninstall Task, specify the settings required for uninstalling.

Command Line Options

This text box shows the command line options. The options must be appropriate for the type of Task and the specified command line. For MSI Tasks, the valid command line options are those available with Windows Installer 3.0 subject to the constraints defined for MSI Tasks.

Command line options are not available for Package Tasks.

Assignments tab (Task Properties)



The **Task Properties | Assignments** tab shows the targets that are scheduled to receive the current Task. On this tab, you can also assign or unassign the Task and change the Task schedule. This tab is displayed when you open the Task Properties and click on the Assignments tab.

The information displayed on this tab includes:

- **Assigned Through**: Name of the computer or group the Task is assigned to.
- **Schedule**: The date and time when the Task will be installed.

Update the Task schedule

1. Highlight the name of one or more targets to which the Task is assigned.
2. Right-click and select **Schedule**. The Schedule Task dialog box opens.
3. Continue using the scheduling wizard to change the settings.

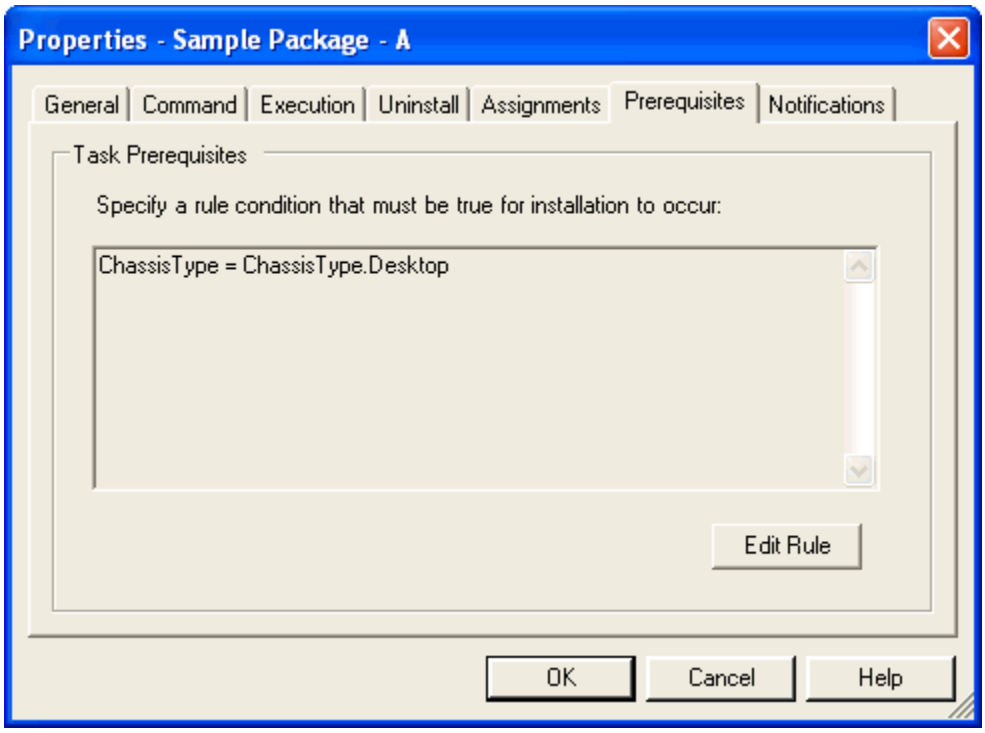
   You can choose to install, reinstall, or uninstall the Task.

4. Click **Finish**.

Reinstall the Task

1. Highlight the name of one or more targets to which the Task is assigned.
2. Right-click and select **Reinstall**. The Reinstall Task dialog box opens.
3. Click **Finish** to reinstall immediately, or **Next** to select scheduling options.

Prerequisites tab (Task Properties)



The **Task Properties | Prerequisites** tab allows you to define conditions that must be satisfied for the task to be executed.

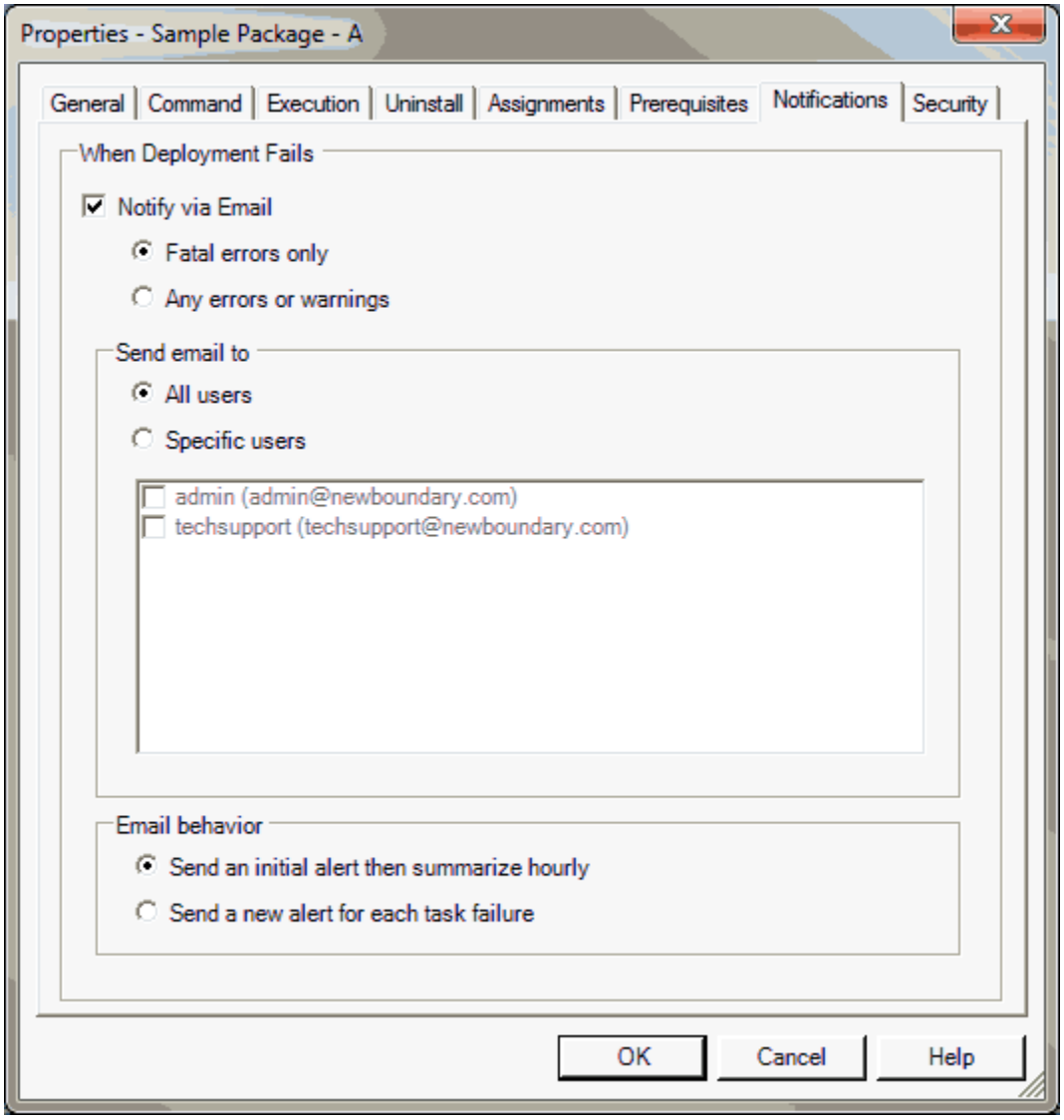
This tab is displayed when you open the Task Properties and click on the Prerequisites tab.

This tab displays the current rule text, if any. Press the **Edit Rule** button to open the Rule Editor dialog to edit the rule.

The client evaluates the rule before the Task is scheduled to run, then takes the following actions:

- If the rule is satisfied, the Task is executed.
- If the rule is not satisfied, the Task will fail. It will not evaluate the condition again until the Task is reinstalled or assigned again.

Notifications tab (Task Properties)



The **Task Properties | Notifications** tab allows you to configure deployment alerts. This allows you to receive an email when this task fails to deploy.

This tab is displayed when you open the Task Properties and click on the Notifications tab.

You can choose to receive emails when a a task fails:

☐**Notify via Email**: With this option turned off, no notification emails will be sent when the deployment fails.

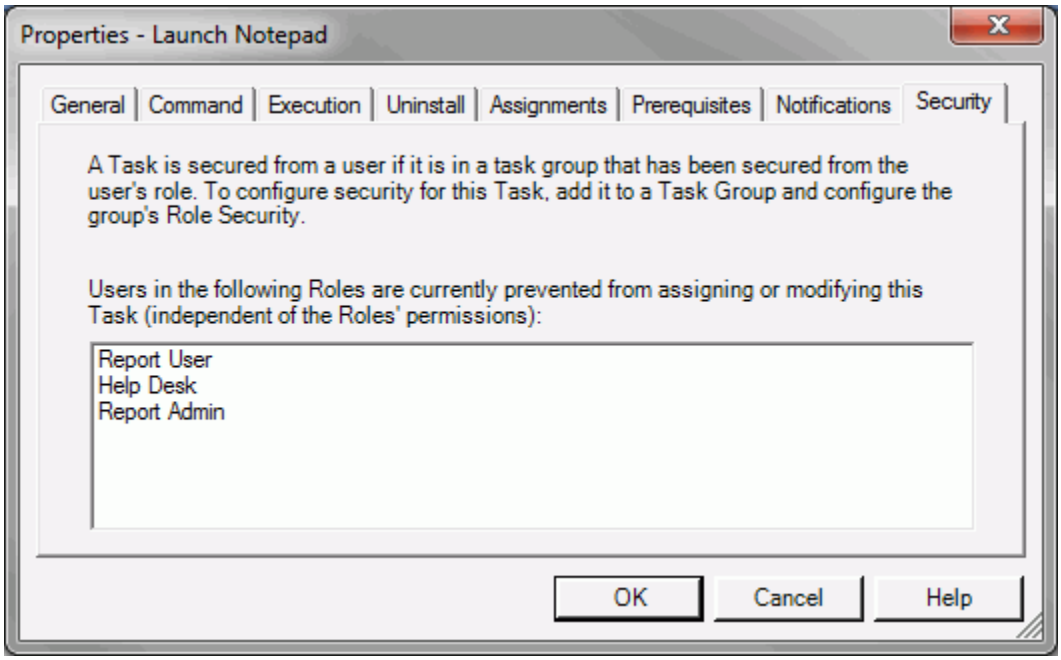
☑**Notify via Email**: With this option selected, notifications will be sent when deployments fail.

You can choose:

- To receive notifications for all errors or just fatal errors.
- To send failure emails to all users or you can select specific users.
- To select whether users receive a summary email (hourly), or an email for each task failure. Sending summary emails helps reduce the likelihood of users receiving a flood of emails.

Note Enabling email notifications requires configuration within your SQL Server database. Please refer to the New Boundary Technologies support site at http://www.nbtnet.newboundary.com/supportkb/ for instructions on how to configure email for your specific environment.

Security tab (Task Properties)



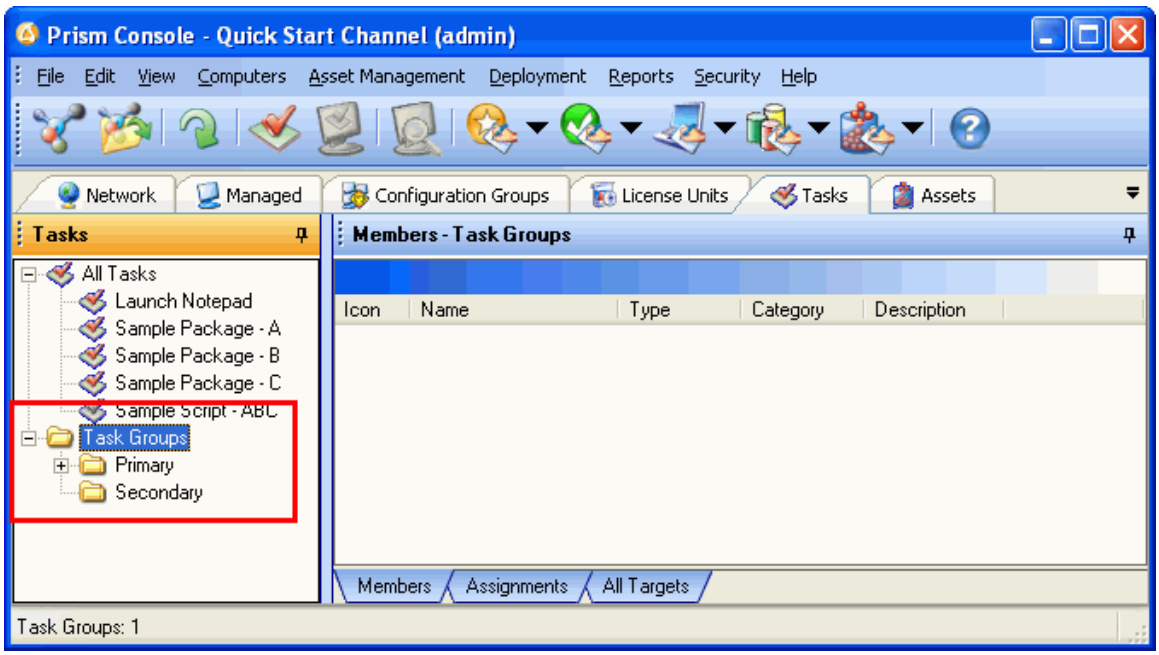
This tab displays the Roles that are currently prevented from modifying the Task due to access control. Each role in the list has been secured from one or more task groups of which this task is a current member. Users in these roles will not be able to assign or modify this task even if they have the requisite task permissions.

To configure security for a task:

1. Add the task to a task group.
2. Open the task group's Properties.
3. Select the Security tab.
4. Un-check any roles that should be prevented from modifying the tasks in the group.

Task Groups

Grouping Tasks



Task groups help you keep Tasks organized based on their function or other similarity. Task groups also you to secure tasks so that specific users (specific roles) cannot modify or assign them.

Task groups are created and populated by you, so they can reflect any grouping that you find useful. You can assign a Task group to a computer or computer group. This has the same effect of assigning the individual Tasks in the group to each target.

To create a Task group:

1. Right-click on the **Task Groups** branch on the Tasks tab in the tree view.
2. Choose **Create Group** from the pop-up menu.
3. On the **Group Name** dialog, enter a name for the group. Click **OK**.
4. With the new group displayed on the Tasks tab, populate the group by clicking and dragging Tasks from the list of Tasks in the details pane onto the group.

Controlling the Installation Order of Tasks

You can control the installation order of tasks in task groups by setting the Priority values on each individual Task. Tasks with lower priority values will be installed before tasks with higher values. The priority settings on a Task are channel-wide, hence impact the relative execution order of that Task within any Task Group to which the Task belongs.

To specify an installation order:

1. Select the Task.
2. Open the Task Properties dialog either by right-clicking on the tasks or via the **Edit** menu.
3. Select **Priority** and then select a priority value for that Task.
4. Press **OK**.
5. Repeat 1-4, giving each Task a different priority value.

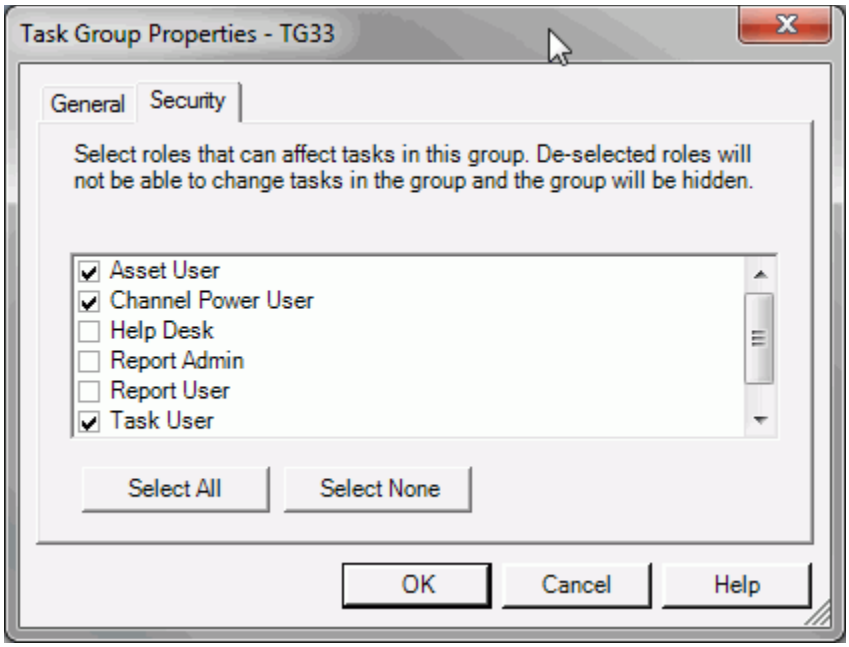
Tasks whose installation order is **System default** will install after Tasks that have **Priority** values. Tasks with the same **Priority** value will be installed in a system-defined order.

General tab (Task Group Properties)

The **Task Group Properties** dialog displays information about the selected task group. This dialog appears when you select a Task Group and choose **Properties** from either the **Edit** menu or the right-click menu.

The **General** tab allows you to change the name of the Task Group. Task Group names must be unique.

Security tab (Task Group Properties)



The Security tab displays the roles that are allowed to assign and manipulate the group's tasks. If a role is *not* selected for a Task Group, a user in that role:

- Will not be able to see the Task Group;
- Will not be able to assign, unassign, or edit Tasks in that Task Group.

The user will still be able to see the secured Tasks; but they will not be able to assign them or edit them. For that user, such tasks appear with a special secured Task icon: .

A user in a role that *is* selected can see the Task Group, which will appear with a special icon indicating that the task group is secured from one or more roles.

Note: This tab is visible only if the channel is secured and the user has permission to view security information.

Task Security

Task security offers two ways of restricting a user's ability to modify tasks. Both approaches are based on configuring the user's role:

1. **Restricting actions that can be performed**: Users must have the appropriate permission to perform a specific action on a task. This is done by specifying task permissions on roles.
2. **Restricting Access to specific tasks**: Users can be restricted from modifying specific tasks, independent of their permissions. This is done by controlling a role's access to tasks in a specific task group.

Task Permissions

The actions a user can perform on a task are controlled by manipulating the role's task-related permissions. These permissions include:

- Task Assignment permissions - these permissions allow the user to assign tasks to computer and groups and edit assignments;
- Task Group permissions - these permissions allow the user to create and delete task groups and modify group membership;
- Task Management - these permissions allow the user to create, delete, and edit tasks.

For example, a Help desk role might have permission to assign tasks but not create or delete tasks.

Task-related permissions for a role are specified on the Permissions tab of the **Role Properties** dialog.

Controlling Access to Tasks

By default, a role's task-related permissions apply to all tasks. For example, a role with **Assign Tasks** permissions can assign any task.

To limit which tasks a user can manipulate, a role can be *restricted* such that certain tasks cannot be assigned or edited by that role regardless of the role's permissions. This can be useful in situations where individual administrators manage independent subsets of tasks in the channel - for example, some administrators may not be allowed to install critical tasks.

Access is controlled by securing specific **task groups** from a role. When a task group is secured from a role, users in that role cannot see the task group and cannot assign tasks in that group, independent of their permissions and even if those tasks are in other task groups that the user *can* manipulate.

When a task group has been secured from a role, any task added to that task group becomes secured from the role. The tasks secured from a role can change over time since the group's membership can change.

Although secured tasks cannot be assigned (either directly or indirectly) by the associated role, those tasks can be seen and are visible in reports.

Securing Access to Tasks

Securing access to tasks can be done in two equivalent ways:

1. **Secure a Role**: First, select a role by opening the Manage Channel Security dialog, and press **Properties**. On the Secured Task Groups tab of the **Role Properties** dialog, add task groups that are to be secured from users in this role. Users in this role will not be able to modify tasks in those task groups.

2. **Secure a Task Group**: First, select a Task Group and open its **Properties**. Select the Security tab, which lists all available roles. Un-check roles that are to be secured from this task group. Users in these roles will not be able to modify tasks in this task group.

These approaches are equivalent. For example, if you select a task group and un-check a role, the task group will appear in the list of secured groups under the Secured Task Groups tab of the role's **Properties**.

Viewing a Task's Secured Roles

To view the roles that are currently secured from assigning or modifying a specific task, open the task's **Properties** and select the Security tab.

Disabling or Deleting Targets and Tasks

Temporarily Disabling Tasks

Tasks can be temporarily taken off line while you diagnose a problem. This feature lets you disengage the Task from the Channel without extensive editing. For example, if you have discovered a problem with one of the Prism files, disable the Task while you diagnose the problem. When the problem has been fixed, activate the Task once more without having to re-assign it or duplicate complex scheduling options.

You can disable single or multiple Tasks. In each case, highlight the name of one or more Tasks, then set the options on the **Properties** dialog box. Disable the selected objects with the **Disable** check box:

☐ **Disable**: With this option turned off, the Task is active. For example, the Task can be scheduled and installed. This option is the default.

☑ **Disable**: With this option selected on the Task Properties | General tab, the Task will not be installed. If the scheduled installation time passes while the Task is disabled and you clear this check box, you will need to re-install the Task.

Temporarily Stopping the Installation of a Task

If you want to prevent the installation of a Task without dismantling your Task assignments and schedule, you can temporarily disable it. To disable a Task:

1. Highlight the Task name in the tree view or details pane.
2. Open the **Task Properties** dialog box by doing one of the following:
   - Select **Edit** | **Properties**.
   - Right-click on the Task name and choose **Properties** from the pop-up menu.
3. In the Task Properties | General tab, select the **Disabled** check box.

 The Task icon is grayed to indicate that it is disabled.

The Task cannot be installed while the **Disabled** option is selected. You can reactivate the Task at any time by clearing the **Disabled** check box on the Task Properties | General tab.

Temporarily Preventing Installation on a Computer

If there is a problem with an individual computer account or a group of computers, you can temporarily take these computers offline while you diagnose and solve the problem. While a computer is disabled, it will not receive any Tasks. When the problem is fixed, you can quickly enable the computer(s) and continue with scheduled Task installations. To disable an individual computer or a group of computers:

1. Highlight the computer name in the details pane.
2. Open the **Properties** dialog box by doing one of the following:
   - Select **Edit** | **Properties**.
   - Right-click on the computer name and choose **Properties** from the pop-up menu.
3. In the **Computer Properties | General Tab**, select the **Disabled** check box.

The computer icon  changes to indicate that it is disabled.

Tasks will not be installed for the selected computer(s) while the **Disabled** option is selected. You can reactivate the computer at any time by clearing the **Disabled** check box on the Computer Properties | General tab.

Unscheduling a Task

A scheduled Task can be removed from the schedule at any time. When a Task is unscheduled, the Task itself and all of its settings remain part of the Channel, but it is removed from the schedule and the record of this assignment is removed (for targets where it has already been installed).

To unschedule a Task or remove it from the schedule:

1. With the Assignments tab open in the details pane, highlight the name of the target originally scheduled to receive the Task.
2. Right-click and select **Delete** from the pop-up menu.
3. Deploy asks if you want to delete the assignment. Click **Yes**.

The Task is removed from the selected target and will not be installed for that target.

Note: Removing or deleting a Task from the schedule or from the Channel is not the same as uninstalling or removing the changes made on a target computer. To uninstall a Task or reverse the changes made on a computer, see Uninstalling Changes with Deploy.

Permanently Removing a Task from a Channel

Deploy lets you temporarily disable (or temporarily halt) or permanently remove a Task. For example, you have a Task for installing an application that your company no longer supports. Delete it permanently from the Channel.

Delete a Task

To remove a Task that you no longer plan to use, do one of the following:

- Right-click on the Task name in the tree view and select **Delete** from the pop-up menu.
- Highlight the name of the Task in the tree view and select **Edit** | **Delete**.

The Task is permanently removed from the Channel. The original Prism file or script file remains unchanged.

Important: Deleting a Task is **not** the same as uninstalling or rolling back the changes on target computers. Deleting a Task from the Channel only removes the Task name and any other settings applied to the Task within the Channel. It does not change the original Prism file, the software installed on a computer, or other changes that were installed on computers by the Task.

See Uninstalling Changes with Deploy for information on uninstalling a Task.

Note: If you delete a scheduled Task from the Channel before it is installed, it will not be installed. To keep the Task intact, but stop its installation, disable or unschedule it instead.

Uninstalling Changes on Target Computers

Uninstalling Changes with Deploy

Deploy lets you remove Tasks and reverse or roll back the changes made by a Task that has been executed. When you roll back a Task, all of the changes made to the target computer by the Task are removed. The simplest Tasks to roll back are those Tasks that use Prism files. If the Task uses a command, write a command to reverse the changes before rolling back the Task.

Uninstalling a Prism File

Uninstalling a Package or MSI Task from the Console is very similar to scheduling it. The difference is that you choose the Uninstall option rather than the Install option on the **Schedule Task** dialog box:

1. In the Console main window, assign the Task to the target(s) where the Task was installed.

   See Assigning and Scheduling Tasks for Installation.
2. In the Schedule Task dialog box, select the **Uninstall the Task** option. Press **Next**.
3. In the Start Deployment panel, set a time for uninstalling the Task. Press **Finish**.

   The Task previously installed with the Task is uninstalled at the specified time.

For more detailed information on how Packages are uninstalled, see the *Packaging Guide* or Packaging section of the online help.

For MSI Tasks, the uninstall uses the Windows Installer "/x" option to uninstall the MSI file.

Uninstalling a Script or Command

Uninstalling a Script or Command requires setting up uninstall information for the Task before it is uninstalled.

First Step: Set Up Uninstall Information

To set up the Command or Script that can be used to later uninstall or reverse the changes made with a Task:

1. With the Tasks tab selected in the tree view, highlight the name of the Task
2. Open the Properties dialog box for that Task by doing one of the following.
   - Choose **Edit** | **Properties**.
   - Right-click on the Task name and choose **Properties** from the pop-up menu.
3. Go to the Task Properties | Uninstall tab.
4. Select the **Use uninstall script** option.
5. Type the uninstall command or file name in the text box or click  to browse to the file.

   The path or location must reflect the location of the file from the target computer's point of view.
6. Click **OK**. The Task can be uninstalled by following the steps below.

Second Step: Uninstall the Script or Command Task

1. In the Console main window, assign the Task to the target(s) where the Task was installed.

   For information on assigning the Task, see Assigning and Scheduling Tasks for Installation.
2. In the Schedule Task dialog box, select the **Uninstall the Task** option.
3. In the Start Deployment dialog box, set a time for uninstalling the Task.

   The uninstall script or command set up in the Task properties is run at the specified time.

Scripting Prism Tasks

Prism has its own scripting language for executing commands from the command line or through scripts. Commands let you automate tasks and uncover some of the hidden power contained within Prism. Commands can be Prism related, such as using the Client to take a "picture" of the user's important documents for backup purposes. Or, it can be unrelated to Prism, such as silently running a Windows XP Service Pack installation. Prism commands can also be grouped in sequence within a Prism script. (A Prism Script always has a .PTS extension). These scripts can be deployed as a script Task from the Console.

Example – Prism Command

Use the /TakePicture command as a command Task in the Console to backup your CEO's important documents each day. Tasks can be set to recur at intervals you define, for instance at 10:00 P.M. each day or every Monday morning. (See Scheduling a Repeating Task.)

Create a Picture definition file (.PWI) to capture specific items on the target computer—in this example set it up to capture specific file extensions. Then, specify this PWI file with the /TakePicture file command.

"C:\Program Files\Prism Suite\ClientInstallFiles\ptclient.exe" /TakePicture H:\backups\%PT_DAY%.PWF \\APPS1\prism\CEO.PWI

(See /TakePicture in this document and "Saving Picture Definitions" in the *Packaging Guide* or Editor online Help for more information.)

What Happens When You Use the Command Line?

When you use the command line to complete tasks, a log file is generated by default. This log file is called ClientLog.XML It records the date and time of all Prism commands and records any errors that occur. (See /DefineLogFile.)

Using the Command Line Options

The command line options can be launched:

- As a single command Task

  For example, run a single-line command to install the latest Microsoft patch. In this example, the patch is stored on your server APPS1.

  \\APPS1\patches\windowskb888763-x86-enu /quiet /passive /norestart

  Prism commands can be launched as any user you specify, so it is important to specify a user account with enough permissions to run the command. (For more information specifying the user account for a Task, see Task Properties | Execution tab.)

- By grouping multiple commands in a Prism script

  If you want to complete multiple commands within one Task, create a Prism script. Any text editor (like Notepad) can be used to author the script. Simply save the script with a .PTS extension. For example, save the following lines to a file named BACKUP.PTS. This script enables diagnostic logging, then performs a customized backup, and finally installs a Prism Package.

  /DefineLogFile /Diagnostic /Size -1

  /TakePicture H:\backups\%PT_DAY%.pwf \\APPS1\backups\registry.pwf

  /InstallPackage \\APPS1\packages\spreadsheet.pwc

Save the BACKUP.PTS file to a network location and assign it to the target computers via a Prism script Task within the Console.

Notes on Command Syntax

There are many ways to run commands. For example, you can run a command as Prism Task, launch it from a batch file, in a login script, by creating a shortcut, and so on. No matter how the command is launched, you must use the appropriate syntax:

- Each command must be preceded by Prism's executable file, either PTCLIENT.EXE or PICTAKER.EXE. If the folder where PTCLIENT.EXE or PICTAKER.EXE is located is not included in the path statement, include the full path name with the command such as C:\PRISM\PTCLIENT.EXE.

  There are two exceptions:

  — Commands within a script must not be preceded by the executable. (See the example above.)

  — Prism Self-Installing files are executables and do not require one of the Prism executables. (See Prism Self-Installing File Command Line Options.)

- When you specify a path, file name, or option that has an embedded space, you must put double quotes around it for Prism to recognize it. For example, a path may be "C:\PROGRAM FILES\PRISM\CLIENTINSTALLFILES\PTCLIENT.EXE", a file name "OFFICE UPGRADE.PWC", or an option "5:00 P.M."
- Commands are not case sensitive. They can by typed in all capital letters, lower case letters, or a combination of the two.
- Each command option must start with a forward slash, for example /TakePicture or /InstallPackage.
- The abbreviated form of the command can be used in place of the full command. For example, you can use /IP for /InstallPackage. The command /IP test.pwc is the same as /InstallPackage test.pwc.
- Commands are usually followed by additional information, such as where to store a Prism file or which file to install. In this document:
  - File names and values that are required are enclosed by < >.
    For example: <filename>.
  - A list of options that you can choose from are shown with a vertical bar **|** to separate them. For example: [/Priority Low | Normal | High | Preempt]
  - Optional information and switches are enclosed by [ ]. For example: [/All]
  - Do not type the bracket symbols (< > or [ ]) when using Prism commands. Each command option is described below the command in this document.
- Each command line task and all options for that task must exist on one line within a script file. This document may show commands that have wrapped to two lines because of formatting limitations or for readability. Examples follow each definition of the command line option.

Command Line Variables

Command lines and scripts can contain any of the following variables. The variables must be enclosed by the percent (%) sign. Prism interprets the text between the % symbols in the following order

WINSYSDIR	the SYSTEM folder
WINDIR	the WINDOWS folder

Application Data	user specific taskbar information
Desktop	the DESKTOP folder (the Windows background)
Favorites	the FAVORITES folder
Fonts	the FONTS folder
NetHood	the NETWORK NEIGHBORHOOD folder
Personal	the default folder for the own documents
Programs	the PROGRAMS folder in the **Start** menu
Recent	the DOCUMENTS folder in the **Start** menu
SendTo	the contents of **SendTo** drop down menus
Start Menu	the **Start** menu
Startup	the STARTUP folder in **Start | Programs**
Templates	the default folder for document templates
LASTERROR	the result of the previous command (See Command Line Errors.)
PT_YEAR	the current year in four characters (2004, 2005, …)
PT_YEAR4	the current year in four characters (2004, 2005, …)
PT_YEAR2	the current year using two characters (04, 05, …)
PT_MONTH	the number of the current month (01 12)
PT_DATE	the day of the current month (01 31)
PT_DAY	the number of the day of the week (Sunday = 01)
PT_HOUR	the current hour in military time (00 23)
PT_HOUR12	the current hour (01 12) followed by either AM or PM
PT_MINUTE	the current number of minutes past the hour (01 59)
PT_SECOND	the current seconds (01 59)
PT_CD	the first drive letter of your CD-ROM drive (D:, E:, and so on)
PT_OS	the operating system family of your computer (winNT)
PT_MAJOROS	the operating system family of your computer (winNT)
PT_MINOROS	the operating system of your computer (win2K, ...)
runerror	the exit value returned from a process initiated by the /Run command

If the text does not match any of the names listed above, environment variables are examined for possible matches. If no matching environment variables are found, the percent signs and text between them is simply removed.

Command Line Errors

Errors that can occur during command execution are given below.

0 No error occurred.

1 The file was already installed.

2 The user declined to have a file installed.

3 There was not enough disk space to install a file.

4 The computer did not have the correct operating system to install the file.

5 An attempt was made to overwrite a file that already exists.

6 There is no uninstall information for the file.

7 An error occurred attempting to install a Prism file.

8 Access to a file was denied.

9 A corrupted file was encountered.

10 A required file did not exist.

11 A single file was used twice in the same command.

12 An error occurred attempting to define an environment variable.

13 An attempt was made to play a script that was already playing.

14 An illegal command was encountered.

The "LASTERROR" variable is used to store the value of the result of the last operation.

Command Lines and Options

/DefineLogFile

/DefineLogFile

This command tells Prism how to store information on the status of the commands and scripts it executes.

There is one log file per computer session. The location of this log file is fixed, and depends on the operating system:

- Windows Vista:

  C:\ProgramData\Prism Deploy\PrismXL\ClientLog.XML

- All others:

  C:\Documents and Settings\All Users\Application Data\Prism Deploy\PrismXL\ClientLog.XML

The log file for previous computer sessions is retained in a separate file that whose name includes the original session's timestamp. For example:

ClientLog2007-05-29 09h05m13s.XML.

Alternate Form

/DLF

Syntax

/DefineLogFile [<filename>] [/Size <maximum size>] [/Diagnostic]

<filename>

This argument is deprecated, as the log file location is in a fixed location. If present, it is simply ignored. It is permitted for backwards compatibility, so that older scripts will remain functional.

<filename> is optional.

/Diagnostic

Includes more detail in the log file.

When /Diagnostic is used, each command or script is recorded in the log file. A diagnostic log file also provides greater detail about Prism files that have been installed with the /InstallPackage command.

Prism always writes information to the log file when an error occurs, even without /Diagnostic.

/Size <maximum size>

Defines the maximum size in kilobytes of the log files in the log file folder. The size specified can be up to 60,000,000 KB.

When the maximum is reached, log files are deleted starting with the oldest.

<maximum size> must not be zero. If you specify -1 for <maximum size>, the log file has no maximum size.

/DefineLogFile Example

This command tells Prism to create a diagnostic log file whose maximum size is 2 MB.

```
/DLF /size 2000 /diagnostic
```

/DefineRulesFile

/DefineRulesFile

This command tells Prism where to look for the text file that defines the rules for registry and INI file management.

Notes

- This command is only valid within a script.
- You can change the location of the rules file as many times as you want during execution by using this command repeatedly.
- If no rules file is specified and the rules file cannot be found, a default set of rules is used.
- Prism does not display an error message if the file you specified is not found or if Prism finds the file named but its format is incorrect.

Alternate Form

/DRF

Syntax

```
/DefineRulesFile [<filename>]
```

<filename>

The name of the file to use for the rules. The default file is called RULES.INI and is installed in the Prism folder. If you specify a file other than RULES.INI, the file must follow the same formatting as the default rules file. (See *Overview of the Rules.INI File* in the Editor online Help or *Packaging Guide*.)

If no path is provided, Prism looks for the file in the current directory. The file name must have the correct extension.

<filename> is optional. If you do not supply <filename>, Prism looks for the rules file created when Prism was installed (RULES.INI). It searches in this order:

1. in its own directory

2. the path

3. the current directory.

Caution: Prism does not display an error message if the file you specified is not found or Prism finds the file named but its format is incorrect.

/DefineRulesFile Example

The following command tells Prism to reference a RULES.INI file centrally located on a server, in the P:\PRISM directory.

```
/DefineRulesFile "P:\Prism\Rules.ini"
```

/EditFile

/EditFile

This command loads one or more Prism files into the Editor.

Notes

- If the editor is already open and is currently displaying a dialog box, the files are not loaded and the editor produces a message beep.
- This command is not available in the Client version of Prism. Only the Editor file, PICTAKER.EXE, supports this command line option.
- The editor never loads two copies of the same file. If a file is already being edited the appropriate Editor window is brought to the foreground.
- You must have write permissions to a file in order to edit it.

Alternate Form

/EF

Syntax

```
/EditFile <filename1> [<filename2> <filename3> ...]
```

<filename1>

A Prism file to edit.

If no path is provided, Prism looks for the file in the current directory. If no extension is given, Prism looks first for a Package that matches the given name. If a Package is not found, Prism then looks for a Picture file. If Prism finds neither a Package nor Picture, it looks for an executable Prism Self-Installing file.

<filename2> <filename3>

Optional additional Prism files to edit, separated by spaces.

These files follow the same rules as <filename1>. When a file in the list cannot be opened for editing, an error message is displayed and the rest of the files are not loaded.

/EditFile Example

The following command opens the baseline Picture C:\PRISM\BEFORE INSTALL.PWF and the Package C:\PRISM\APP.PWC in the Editor.

```
F:\Prism\Pictaker.exe /EditFile "C:\Prism\before install.pwf" C:\Prism\app.pwc
```

/ExitError

/ExitError

This command causes script execution to halt when an error number occurs that is equal to or greater than the number given by <error number>.

Notes

- This command is only valid within a script.
- If the error number is set to 14 or higher, Prism keeps trying to play a script even after it has encountered an error while trying to interpret the script.

Alternate Form

/EE

Syntax

```
/ExitError <error number> [/All]
```

<error number>

The smallest error number that causes the script to terminate.

The script errors that can occur are listed in Command Line Errors. You must supply a number greater than zero. The default value is 13.

/All

If /All is specified and the currently executing script was launched from within another script, execution of all scripts ceases. Otherwise, the current script stops.

/ExitError Example

The following command tells Prism to exit the script if an error occurs:

```
/ExitError 1
```

/ExitScript

/ExitScript

This command causes script execution to halt immediately.

Notes

- This command is only valid within a script.

Alternate Form

/ES

Syntax

```
/ExitScript [/All]
```

For more information on the arguments and examples, click one of the following:

/All

/All

If /All is specified and the currently executing script was launched from within another script, execution of all scripts ceases. Otherwise, the current script stops.

/ExitScript Example

You may want to use this command when using a script to determine if a condition has been met. (See Script Statements.)

```
if exists <filename>
/ExitScript /All
endif
```

/FindChanges

/FindChanges

This command finds the changes between a Prism file and the current state of the computer or between two Prism files. Changes that are found are placed in the file <Package name>.

Note: If the <before filename> file is a Package, the <after filename> file must be a Package as well.

Note: The default folder deletion behavior for the package is "Always delete". To over-ride this, use the /NoDeleteFolders switch, which sets the folder deletion behavior to "Never delete".

Alternate Form

/FC

Syntax

```
/FindChanges <before filename> <Package name> [<after filename>] [/Additions] [/Deletions]
[/Switch] [/NoDeleteFolders]
```

<after filename>

The Prism file representing the "after" in a before/after comparison.

If <after filename> is not provided, the "after" is the current state of the computer.

If no path is provided, Prism looks for the file in the current directory. If no extension is given, Prism looks first for a Picture file that matches the given name. If a Picture is not found, Prism looks for a Package. If Prism finds neither a Package nor Picture, it looks for a Prism Self-Installing file.

<before filename>

The Prism file representing the "before" in a before/after comparison. Settings and files that exist in the <after filename> file but not in the <before filename> file are stored in the <Package name> file. Files and settings that have changed since the <before filename> file was created are also stored in the <Package name> file.

If no path is provided, Prism looks for the file in the current directory. If no extension is given, Prism looks first for a Picture file that matches the given name. If none is found, Prism then looks for a Package. If Prism finds neither a Package nor Picture, it looks for an executable Prism Self-Installing file.

<Package name>

The name of the Prism file to be created to store the changes found during the comparison. This file must be a Package.

If no path is provided, Prism creates the file in the current directory. The extension of a Package is always .PWC.

/Additions

If /Additions is specified, the comparison between the Prism file and the current state of the computer or between two Prism files, finds any additions or modifications to files, folders, shortcuts or registry entries. /Additions is the default.

/Deletions

If /Deletions is specified, the comparison between the Prism file and the current state of the computer or between two Prism files, finds deletions that have occurred to files, folders, shortcuts or registry entries. /FindChanges does not find deletions by default. If you want to find both additions and deletions, you must include the /Additions switch as well as this switch.

/Switch

If /Switch is specified, you can switch the <before filename> and the <after filename>. This argument is useful if you want the current environment as the before picture.

/FindChanges Examples

The following command finds changes made since the Picture C:\PD\REGBACKUP.PWF was taken and store the Package as REG[current date].PWC (for example, REG0501.PWC). (See Command Line Variables.)

```
/FC "C:\PD\reg backup.pwf" reg%PT_MONTH%%PT_DATE %.pwc
```

The following command uses the current environment as the before state, uses C:\PD\REGBACKUP.PWF as the after state, and stores the changes in C:\PD\REGINFO.PWC.

```
/FC C: \PD\regbackup.pwf C:\PD\reginfo /switch
```

/InstallPackage

/InstallPackage

This command installs the Prism file identified by <filename> on a computer.

Alternate Form

/IP

Syntax

```
/InstallPackage <filename> [<uninstall filename>] [/Once] [/Update] [/Always] [/Identification
<name>] [/NoRollback] [/NoRollbackInfo] [/Schedule <date> <time>] [/CommonInfo] [/UserInfo]
[/CurrentUser] [/AllUsers] [/Prompt] [/BeforePrompt] [/NoBeforePrompt] [/DuringPrompt]
[/NoDuringPrompt] [/AfterPrompt] [/NoAfterPrompt] [/ShowErrors] [/NoShowErrors]
[/Priority Low | Normal | High | Preempt] [/NoCheckpointRestart]
```

<filename>

The Prism file to install.

If no path is provided, Prism looks for the file in the current directory. If no extension is given, Prism looks first for a Package file that matches the given name. If a Package is not found, Prism looks for a Picture file. If Prism finds neither a Package nor Picture, it looks for an executable Prism Self-Installing file.

The command /InstallPackage is optional. One or more file names without a preceding command implies the /InstallPackage command.

Note: he /InstallPackage command can be used with a Prism Self-Installing file, if the Client is invoked. In this case, the Prism Self-Installing file is installed as a Prism Package rather than being executed as a Prism Self-Installing file. (See /InstallPackage Examples for an example. See Prism Self-Installing File Command Line Options for options that can be used with a Prism Self-Installing file installed as an executable.)

<machine rollback filename> <user rollback filename>

This option gives Prism the names of the Uninstall files to create. The Uninstall files contain the information needed to undo the changes made when the Prism file was installed. The <machine rollback filename> holds all of the machine settings that need to be rolled back to restore the machine to its previous state. The <user rollback filename> holds all of the user settings that need to be rolled back to restore the machine to its previous state. (See "Understanding Uninstall Files" in the *Packaging Guide* or Editor online Help.)

If no file names are provided, Prism gives the Uninstall files the same name as the Prism file, with the extension .PWR. If Uninstall files for the Prism file already exist, Prism creates new .PWR files incremented by one for each time a new Uninstall file is created. For example, [<FILENAME 1.PWR> <FILENAME 2.PWR>], [<FILENAME 3.PWR> <FILENAME 4.PWR>], and so on.

By default, Prism creates the files in the %TEMP%\unapply folder.

Note: The folder in which Prism creates the Uninstall file by default can be modified.

- On the computer where the Editor is installed, you can specify a location for the Uninstall files in the Editor's **Options | Uninstall** tab.
- On computers where only the Client is installed and you are installing Prism files via the command line or executing a Prism Self-Installing file, you can modify the "Unapply Folder" registry value. This value is changed in the HKCU\Software\Lanovation\PictureTaker \Settings\Directories\Unapply Folder key. The easiest way to distribute this change to the target computers is with a Prism file.

/AfterPrompt

This option forces a prompt to be displayed after installation. If an "after prompt" has been set up through the Editor's **File Properties | Messages** tab for the Prism file, it is used as is. If an "after prompt" was not set up through the Editor, the default message is displayed. The default message tells the user that the file was successfully installed.

/AllUsers

This option overrides the selection under **Installation Method** in the Editor's **File Properties | Requirements** tab for the Prism file and distributes settings and files to all the computer's users. The files and settings installed include both those that are common to all of the users, as well as those targeted for a specific user in the original Prism file.

(For more information, see the Editor's **File Properties | Requirements** tab.)

/Always

If /Always is specified, the file is installed regardless of whether it has been successfully installed previously. /Always is the default.

/BeforePrompt

This option forces a prompt to be displayed before the file is installed. If a "before prompt" has been set up through the Editor's **File Properties | Messages** tab for the Prism file, it is used as is. If a "before prompt" was not set up through the Editor, the default message is displayed. The default message asks users to confirm that they want to install the file.

/CommonInfo

Install the files and settings that are common to all users of the same computer. Unless specific per user options (such as /UserInfo or /CurrentUser) are used in the command line, no individual user settings or files are installed.

/CurrentUser

This option overrides selection under **Installation Method** in the Editor's **File Properties | Requirements** tab for the Prism file and distributes the settings and files just to the current user.

If you later use the /UninstallPackage command to uninstall this Prism file, it will remove all users' changes (by default) on the target computer. Like the /InstallPackage command, the /UninstallPackage command does allow user specific switches.

/DuringPrompt

This option forces a prompt to be displayed during installation. If a "during prompt" has been set up through the Editor's **File Properties | Messages** tab for the Prism file, it is used as is. If a "during prompt" was not set up through the Editor, the default message is displayed. The default message tells the user that the file is being installed.

/Update

If /Update is specified, the Prism file is only to be installed if it has never been previously installed or if it has been modified since the last time it was installed for the currently logged in user.

This option cannot be used with Prism Self-Installing files.

/Identification <name>

This option lets you define the name that Prism records in the registry when installing a file. When using the /Once, /Update, or /Schedule option, Prism looks under this name for the last time the Prism file was installed. The default value for this name is the current Windows user. By explicitly supplying one identifier for a computer, you can prevent the same Prism file from being installed multiple times, for each user of that computer.

Note: If the Prism file being installed contains user specific information, such as Registry (Current User) or desktop shortcuts, the user who is logged in at the time of distribution is the only user who will get these changes. This applies unless the Editor's **Properties | Requirements** tab for the Prism file is set to be installed for "Unattended/All Users."

/NoAfterPrompt

This option ensures that no prompt is displayed after installation. When this command is used, the settings for the "after prompt" set up through the Editor's **File Properties | Messages** tab for the Prism file are ignored.

/NoBeforePrompt

This option ensures that no prompt is displayed before the file is installed. When this command is used, the settings for the "before prompt" set up through the Editor's **File Properties | Messages** tab for the Prism file are ignored.

/NoCheckpointRestart

If the installation of the Prism file fails before it is complete, this option prevents the creation of a checkpoint file (.PKGCHKPT) on the target computer. It also prevents use of an existing checkpoint file during reinstallation of the Prism file.

Use of this option overrides settings on the Editor's **Options | Checkpoint/Restart** tab in the Prism file.

/NoDuringPrompt

This option ensures that no prompt is displayed during installation. When this command is used, the settings for the "during prompt" set up through the Editor's **File Properties | Messages** tab for the Prism file are ignored.

/NoRollback

This option tells Prism not to create an Uninstall file when the Prism file is installed. If something goes wrong during a distribution, the changes that were made up to the point of failure are rolled back, but a PWR file is not created. Once the operation has been completed successfully, the rollback information is discarded.

/NoRollbackInfo

No uninstall information is maintained, not even during the Prism file installation. So, if the operation is interrupted, the computer is left in a state of mid install.

/NoRollbackInfo is designed for situations where performance is paramount. In some cases, it can increase the speed of the operation by as much as 10–15 percent. However, it should not be used except in special situations.

This option automatically includes the /NoCheckpointRestart option.

/Once

If /Once is specified, the Prism file is only to be installed if it has never been previously installed to the currently logged in user of the computer. If you want to install the Prism file only once for each computer, use the /Identification switch along with the /Once switch.

This option cannot be used with Prism Self-Installing files.

/Priority Low | Normal | High | Preempt

The /Priority switch controls how many time slices Prism gets in relationship to other processes running. It must be followed by one of the keywords Low, Normal, High, or Preempt. When no other applications are running, /Priority Preempt only boosts performance by about 4 percent. When a lot of other applications are running, the gain can be as high as 30 40 percent.

/Prompt

Use this option to Prism display a prompt when it installs the Prism file. Through the Editor's **File Properties | Messages** tab for the Prism file, you can specify the message you want displayed. When /Prompt is used in the command line, the option selected for Display message before installing file is ignored. You are always asked first.

If no message is specified in the file's properties, the default message "Are you sure you want to install the file [filename] on your computer?" is displayed.

When /Prompt is used along with /Once, /Update, /Always, or /Schedule, Prism checks to see if one of these switches would cause the file to be installed. If so, Prism prompts the user.

/Prompt offers the same functionality as using the three options /ShowErrors /BeforePrompt /AfterPrompt.

/Schedule <date> <time>

When this option is specified, Prism checks the current date and time. If it is later than the date and time specified by /Schedule, the Prism file is installed if it has never been successfully installed before or if the last time it was installed was before the specified date and time.

The date must be in the format MM/DD/YY. The time must be in the format HH:MM. It can either be in military time or followed by AM or PM. If either the date or time contain spaces, they must be enclosed by quotes.

<time> is optional. If it is not specified, midnight is assumed.

/ShowErrors

If an error occurs, the user should be notified using a message box.

The commands /Prompt, /BeforePrompt, and /AfterPrompt imply /ShowErrors. Using /ShowErrors with these commands is permitted but not necessary.

/Update

If /Update is specified, the Prism file is only to be installed if it has never been previously installed or if it has been modified since the last time it was installed for the currently logged in user.

This option cannot be used with Prism Self-Installing files.

/UserInfo

Install files and settings that need to be repeated for each user of the computer. These items include the Start menu, desktop, application data folders, and the HKEY_CURRENT_USER area of the registry.

Unless the /CommonInfo option is also used in the command line, no settings or files are be installed that are common to all of the users on the target computer.

The selection under **Installation Method** in the Editor's **File Properties | Requirements** tab for the Prism file is used to determine if these settings should be installed for just the current user or for all users. An exception would be if this setting in the file's properties is overridden by either the /CurrentUser or /AllUsers option.

/InstallPackage Examples

The following command installs the Package P:\PRISM\APP.PWC if the date is 5/1/04 12:00 A.M. or later and the file has not been installed since then. It also creates an uninstall file in the %TEMP%\Uninstall folder. In the registry, Prism records the time and date when the Package was installed, using the environment variable %computername% for the current computer's name. This Package is installed once on each computer.

```
/IP "P:\Prism\app.pwc" /schedule 05/01/04 /Identification "%computername%"
```

The following command updates the computer's "common" settings if the Package has been edited since the last time they were installed. It does not change the individual user settings. If the common settings are to be updated, a window is displayed during the installation. The **/Update** option prohibits using this command for a Prism Self-Installing file.

```
/InstallPackage Package.pwc /CommonInfo /Update /DuringPrompt
```

The following command updates the settings for the current user only. It leaves everything else on the computer alone. With this command, a prompt is displayed before the installation.

```
/InstallPackage Package.pwc /UserInfo /CurrentUser /BeforePrompt
```

The following command installs a Prism Self-Installing file as a Package and updates the settings for all of the users on the target computer. It adds switches to make sure that no prompts are

displayed during the operation. Since the Prism Self-Installing file is being installed as a Package, an entry is not added to the Windows Control Panel's *Add/Remove Programs*.

```
/InstallPackage SIF.exe /UserInfo /AllUsers /NoBeforePrompt /NoDuringPrompt /NoAfterPrompt
```

The following command makes sure that all the computer's users have the settings they need from this Package. It also updates the common settings.

```
PTClient /InstallPackage Package.pwc /AllUsers
```

The following command installs the Package for all of the computer's users who have not previously gotten it. It tells the current user when the operation is finished. Obviously, nothing can be said to the users who are not logged in yet. Because of the **/Once** switch this command cannot be used with a Prism Self-Installing file.

```
/InstallPackage Package.pwc /UserInfo /AllUsers /Once /AfterPrompt
```

The following command reinstalls all the "common" information from the Package. It also makes sure that no messages are displayed during the operation.

```
/InstallPackage Package.pwc /CommonInfo /NoBeforePrompt /NoDuringPrompt /NoAfterPrompt
```

/MessageError

/MessageError

This command defines which errors cause a message to be displayed to the user. A message is displayed when an error number occurs that is greater than or equal to the number given by <error number>.

Notes

- This command is only valid within a script.
- Set this number to a small value to help diagnose script problems.
- Set this number to zero to generate an error message after every command.

Alternate Form

/ME

Syntax

```
/MessageError <error number>
```

<error number>

The smallest error number that causes an error message to be displayed. (See Command Line Errors for a list of error messages.) You must supply a number greater than zero. The default value is 13.

/MessageError Example

The following command may be added to a script to make sure the script is working properly. You are notified if Prism encounters any errors.

```
/MessageError 1
```

/OverwriteFiles

/OverwriteFiles

This command defines whether existing Prism files should be overwritten. For instance, if the command /TakePicture PICTURE1.PWF is executed and the PICTURE.PWF file already exists, it is replaced with a newly created Picture.

Alternate Form

/OF

Syntax

/OverwriteFiles True | False

Notes

- Alternative forms for True are 1 and Yes.
- Alternative forms for False are 0 and No.
- This command is only valid within a script.

True | False

True indicates that existing files should be overwritten by the /TakePicture or /FindChanges commands.

False indicates that existing files should not be overwritten. When a command would replace a file that already exists, the command is not executed and error number 5 is generated. (See Command Line Errors for a list of error messages.)

The default value is **True**.

/OverwriteFiles Example

The following command can be added to a script to make sure that the backup Picture REGBACKUP.PWF is never overwritten. Once the file is created, Prism will not overwrite REGBACKUP.PWF the next time the script runs.

```
/OverwriteFiles False
/TP H:\backups\regbackup.pwf "P:\Prism\regbackup.pwi"
```

/PlayScript

/PlayScript

This command causes the text file identified by **<filename>** to be opened. Each line of the script is executed in turn. (See Script Statements for information on controlling the execution of scripts.)

Notes:

- Prism commands within the script are not preceded by the Client executable (PTCLIENT.EXE), since the Client is already running when it plays the script.
- You can play scripts from within other scripts.

Alternate Form

/PS

Syntax

/PlayScript <filename>

<filename>

A text file to execute.

If no path is provided, Prism looks for the file in the current directory. The extension of the file must be specified if .PTS, the default extension, is not used.

PlayScript Example

The following command launches the script P:\PRISM\SCRIPTS\ROLLOUT.PTS which installs several Prism files for various applications.

```
"P:\Prism\PTClient.exe" /PlayScript

"P:\Prism\scripts\rollout.pts"
```

/PrintFile

/PrintFile

This command prints one or more Prism files.

Notes

- When neither /Files nor /Registry is specified, both types of information are printed.
- The computer must have a default printer defined.

Alternate Form

/PF

Syntax

/PrintFile <filename1> [<filename2> <filename3> ...] [/All] [/Files] [/Registry] [/Prompt]

<filename1>

The name of a Prism file to print.

If no path is provided, Prism looks for the file in the current directory. If no extension is given, Prism looks first for a Package that matches the given name. If a Package is not found, Prism then looks for a Picture file. If Prism finds neither a Package nor Picture, it looks for an executable Prism Self-Installing file.

<filename2> <filename3>

The names of additional Prism files to print, separated by spaces. These additional names are optional.

These files follow the same rules as <filename1>. When a file in the list cannot be opened, an error message is displayed but the rest of the files are printed.

/All

When this option is used, both file and registry information is printed. This option is the default.

/Files

When this option is used, only the file information (name, date, time, and size) stored in the Prism file is printed.

/Prompt

This option displays the **Print** dialog box before the file is printed.

/Registry

When this option is used, only the registry information stored in the Prism file is printed.

/PrintFile Example

The following command prints all the registry information stored in the Picture H:\BACKUP\BACKUP.PWF.

```
/PrintFile H:\backup\backup.pwf /registry
```

/Run

/Run

Prism supports a /Run command for use within scripts. The /Run command lets you launch external DOS or Windows programs from a Prism script.

The /Wait switch (default) causes script execution to wait for completion of the launched program. The /Continue switch causes script execution to proceed immediately after the program is launched.

The optional /Minimized | /Maximized | /Hidden switches (default is none, meaning Normal) control the appearance state of the launched window.

Syntax

```
/Run [/Wait | /Continue] [/Minimized | /Maximized | /Hidden] <external command>
```

/Continue

Script execution should resume immediately after the specified program is launched.

/Wait

Causes script execution to pause until the external program is completed.
/Wait is the default when neither of the options is provided.

<external command>

The DOS or Windows program to be launched. You do not need to place quote marks (") around the program and its arguments. But, you must place quote marks around single arguments or file names that contain spaces.

The entire command-line can contain environment variables or Prism variables.

/ScriptVer

/ScriptVer

Use this command to add installation requirement rules to a Prism Script. While you are learning the rule syntax, create rules through the **File Properties | Requirements** tab for the Prism file or the Console user-defined groups, then copy and paste the rule into a script.

The ScriptVer command determines how the lines following it are evaluated. Any logic statements (using IF and ELSEIF) preceding a /ScriptVer command are evaluated according to the rules followed by the other commands. Any logic statements following a /ScriptVer command are evaluated using the rules syntax described in Rule Variables and Operators for User-Defined

Configuration Groups or "Rule Variables and Operators for Installation Rules" in the *Packaging Guide* or Editor online Help.

Syntax

/ScriptVer <version number>

<version number>

The version of Prism used to evaluate the script. Use the major revision number of Prism, for example 5 for version 5.x.

Note: This command is compatible only with Prism version 5 or later.

/ScriptVer Example

The following script contrasts the use of a Package to check the computer before installing a Prism file with use of the /ScriptVer command and an installation rule.

```
rem
rem Sample Script using /ScriptVer command
rem
rem This part of the script uses a Package to check the target computer before installing Testing.pwc

/InstallPackage %pddir%\checksomething.pwc
If %lasterror% >= "43"
 /InstallPackage %pddir%\subdir\testing.pwc
Endif

rem This part of the script uses the /ScriptVer command and installation rule before installing
Testing.pwc

/ScriptVer 5
If ("%userdomain%" = "yourcompanydomain") AND (EXISTS <Registry Key>
"HKLM\Software\FooCo\Installed")
 /InstallPackage %pddir%\subdir\testing.pwc
Endif
```

/Set <variable>

/Set <variable>

Sets an environment variable for Prism to use. The variable can be used anywhere in the current script or within a Prism file that is being created or installed.

Note

- This command is only valid within a script.
- The variables you define using **/Set** are only valid with the currently running script environment. Other programs are not able to use these variables.

Syntax

/Set <variable> [<value>]

<variable>

The environment variable to change or add.

<value>

The new value for the variable. If this value is missing or blank, the variable is removed.

/Set <variable> Example

The following example sets the variable PDDIR for the location of Prism directory where all the Prism files are located. The variable is used in place of the Prism path throughout the script.

```
/Set PDDIR "P:\apps\Prism\changefiles"
/IP %PDDIR%\app1.pwc
/IP %PDDIR%\app2.pwc
```

/TakePicture

/TakePicture

This command takes a Picture of the computer and places it in <filename>.

Alternate Form

/TP

Syntax

```
/TakePicture <filename> [/Before | /Hardware | /NoHardware | <Picture Definition file>]
```

<filename>

The file for the new Picture.

If no path is provided, Prism creates the file in the current directory. The extension of the file must be .PWF. If no extension is given, .PWF is appended to the file name.

/Before | /Hardware | /NoHardware | <Picture definition file>

The following options tell Prism what information to place in the Picture. You may only provide one of these options.

- **/Before** stores the standard information needed later to find newly installed software or to diagnose problems. The Picture option **Baseline Picture** is used. /Before is the default.
- **/Hardware** stores a complete Picture of the computer's current Windows settings. The Picture option **Windows Settings** is used.
- **/NoHardware** stores a complete Picture of the computer's Windows settings, but leaves out information that might depend on the exact adapters, display, keyboard, and so on in use by the computer. That is, the Picture option **Windows Settings Without Hardware** is used.
- **<Picture definition file>** is the definition file you created using the **Save** option in the **Custom Picture Options** dialog box. Only the options you specifically included in the definition file are used when taking the Picture. If no path is provided, Prism looks for the file in the current directory. The extension of the file must be .PWI.

/TakePicture Example

The following example takes a Picture of the user's registry, as defined by the PWI file on the server. The Picture file REG[currentdate].PWF is stored in the user's home directory on the server.

```
/TP H:\backups\reg%PT_MONTH%%PT_DATE%.pwf "P:\Prism\reg.pwi"
```

/UninstallPackage

/UninstallPackage

This command reverses the changes made to a computer when <filename> was installed.

Alternate Form

/UP

Syntax

/UninstallPackage <unique identifier> | <filename> [/All]

[/Identification <name>] [/CommonInfo] [/UserInfo] [/CurrentUser]

[/AllUsers] [/Prompt] [/ShowErrors]

<filename>

The Prism file to uninstall. If no path is provided, the current directory is searched for the file. If no extension is given, .PWC is assumed.

Note: The /UninstallPackage command can be used to roll back a Prism Self-Installing file if the Client was invoked when the file was installed and it was installed as a Prism Package, rather than being executed as an .EXE. (See "Understanding Uninstall Files" in the Packaging Guide or Editor online Help for information on uninstalling or rolling back Prism Self-Installing files.)

<unique identifier>

This option specifies the unique identifier that Prism assigns to each Prism file when it is created (for example, C3A5D830 190D 11D3 A76C 00105A179930). The Prism file associated with this identifier is uninstalled. The identifier can be found by viewing the **File Properties | Information** tab for the Prism file.

(For more information, see the Editor's **File Properties | Information** tab.)

/All

If the Prism file was installed more than once, use this option to uninstall all changes made with this Prism file. Without the /All switch, only the changes made with the most recent installation of the Prism file are rolled back.

/AllUsers

By default the /UninstallPackage command removes all users' changes from the computer, as well as all settings and files that are common to all of the users. So, this option is unnecessary, but it may be useful as a reminder of the default behavior of the command.

/CommonInfo

Uninstall the files and settings that are common to all users of the same computer. Unless specific per user options (such as /UserInfo or /CurrentUser) are used in the command line, no individual user settings or files are uninstalled.

/CurrentUser

This option overrides the default behavior of removing all users' changes on the target computer. It also overrides any user specific settings used when the Prism file was installed originally. With this option added to the /UninstallPackage command, only the changes for the current user are rolled back.

/Identification <name>

If the /Identification option was used when the Prism file was installed, it must be used to uninstall the file as well.

/Prompt

Use this option Prism to display a prompt before uninstalling the Prism file so the operation can be canceled. If the file has been installed multiple times, using the /Prompt option gives the user the choice of uninstalling all instances of the file or just the last instance. If the file has been installed multiple times but the /Prompt option is not used, only the changes made by the last installation of the file are removed.

Both /Prompt and /ShowErrors cause the user to be asked first. If the file has been installed more than once, these prompts also cause a dialog box to be displayed, asking the user whether all installations or only the latest installations are to be removed.

/ShowErrors

If an error occurs, the user should be notified using a message box.

The /Prompt command implies /ShowErrors. Using /ShowErrors with this command is permitted but not necessary.

/UserInfo

Uninstall files and settings that need to be repeated for each user of the computer. These items include the Start menu, desktop, application data folders, and the HKEY_CURRENT_USER area of the registry.

Unless the /CommonInfo option is also used in the command line, no settings or files are uninstalled that are common to all of the users on the target computer.

/UninstallPackage Examples

The following command uninstalls the Package, APP.PWC, which is no longer needed. The user will be prompted before Prism uninstalls the file.

```
/UP P:\Prism\app.pwc /prompt
```

The following command removes the settings and files in the Prism file from the current user. It leaves everything else alone. It asks the user before doing this. If the settings have been installed more than once, it also asks whether to uninstall all changes or the most recent changes.

```
/UninstallPackage Package.pwc /UserInfo /CurrentUser /Prompt
```

Self-Installing File Command Line Options

Prism Self-Installing File Command Line Options

A Prism Self-Installing file can be installed from the command line or by placing the file name in a script. To install the Prism Self-Installing file, typing the name alone is sufficient. You can also add the following Prism options, just as you would with a command.

Note: A Prism Self-Installing file can be installed as a Prism Package by invoking the Client and using the /InstallPackage command. (See /InstallPackage for an example.)

Syntax

<filename> [/NoRollback] [/NoRollbackInfo] [/CommonInfo] [/UserInfo] [/CurrentUser] [/AllUsers] [/Prompt] [/BeforePrompt] [/NoBeforePrompt] [/DuringPrompt] [/NoDuringPrompt] [/AfterPrompt] [/NoAfterPrompt] [/NoShowErrors] [/Priority Low | Normal | High | Preempt]

<filename>:

The name of the Prism Self-Installing file to install. The file name without a preceding command executes the file.

/AfterPrompt

Force a prompt to be displayed after installation. If an "after prompt" has been set up through the **File Properties | Messages** tab, it is used as is. If an "after prompt" was not set up through the Editor, the default message is displayed. The default message tells the user that the file was successfully installed.

(For more information, see the Editor's **File Properties | Messages** tab.)

/AllUsers

This option overrides the selection for **Installation Method** in the **File Properties | Requirements** tab for the Prism file and distributes settings and files to all the computer's users. The files and settings installed include both those that are common to all of the users, as well as those targeted for a specific user in the original Prism Self-Installing file.

(For more information, see the Editor's **File Properties | Requirements** tab.)

/BeforePrompt

This option forces a prompt to be displayed before the file is installed. If a "before prompt" has been set up through the **File Properties | Messages** tab, it is used as is. If a "before prompt" was not set up through the Editor, the default message is displayed. The default message asks users to confirm that they want to install the file.

(For more information, see the Editor's **File Properties | Messages** tab.)

/CommonInfo

Install the files and settings that are common to all users of the same computer. Unless specific per-user options (such as /UserInfo or /CurrentUser) are used in the command line, no individual user settings or files are installed.

/CurrentUser

This option overrides the selection for **Installation Method** in the **File Properties | Requirements** tab for the Prism file and distributes the settings and files to just the current user.

(For more information, see the Editor's **File Properties | Requirements** tab.)

/DuringPrompt

This option forces a prompt to be displayed during installation. If a "during prompt" has been set up through the **File Properties | Messages** tab, it is used as is. If a "during prompt" was not set up through the Editor, the default message is displayed. The default message tells the user that the file is being installed.

(For more information, see the Editor's **File Properties | Messages** tab.)

/NoAfterPrompt

This option ensures that no prompt is displayed after installation. When this command is used the settings for the "after prompt" set up through the **File Properties | Messages** tab are ignored.

(For more information, see the Editor's **File Properties | Messages** tab.)

/NoBeforePrompt

This option ensures that no prompt is displayed before the file is installed. When this command is used, the settings for the "before prompt" set up through the **File Properties | Messages** tab are ignored.

(For more information, see the Editor's **File Properties | Messages** tab.)

/NoDuringPrompt

This option ensures that no prompt is displayed during installation. When this command is used, the settings for the "during prompt" set up through the **File Properties | Messages** tab are ignored.

(For more information, see the Editor's **File Properties | Messages** tab.)

/NoShowErrors

No error messages are displayed to the user during the Prism Self-Installing file installation. Use this option in situations where a user is not available to respond to the error message or an error may interfere with the file installation. By default, errors are displayed without this option.

/NoRollback

This option tells Prism not to create a Uninstall file when the Prism Self-Installing file is installed. If something goes wrong during a distribution, the changes that were made up to the point of failure are rolled back. Once the operation has been completed successfully, the rollback information is discarded.

Note: See "Understanding Uninstall Files" in the *Packaging Guide* or Editor online Help for information on Prism Self-Installing files and Uninstall files.

/NoRollbackInfo

With this option, uninstall information is not maintained, not even during the Prism file installation. With this option, if the operation is interrupted, the target computer is left in a state of mid-install.

/NoRollbackInfo is designed for situations where performance is paramount. In some cases, it can increase the speed of the operation by as much as 10-15 percent. However, it should not be used except in special situations.

/NoShowErrors

No error messages are displayed to the user during the Prism Self-Installing file installation. Use this option in situations where a user is not available to respond to the error message or an error may interfere with the file installation. By default, errors are displayed without this option.

/Priority Low | Normal | High | Preempt

The new /Priority switch controls how many time-slices Prism gets in relationship to other processes running. It must be followed by one of the keywords Low, Normal, High, or Preempt. When no other applications are running, /Priority Preempt only boosts performance by about 4 percent. When many other applications are running, however, the gain can be as high as 30-40 percent.

/Prompt

Use this option to display a prompt when the Prism Self-Installing file is installed. You can specify the message you want displayed in the File Properties **Before Message** dialog box. When **/Prompt** is used in the command line, the option selected for **Display message before installing file** is ignored. You are always asked first.

If no message is specified in the file's properties, the default message "Are you sure you want to install the file [filename] on your computer?" is displayed.

(For more information, see the Editor's **File Properties | Messages** tab.)

/UserInfo

Install files and settings that need to be repeated for each user of the computer. These items include the Start menu, desktop, application data folders, and the HKEY_CURRENT_USER area of the registry.

Unless the /CommonInfo option is also used in the command line, no settings or files that are common to all of the users on the target computer are installed.

The option selected under **Installation Method** in the **File Properties | Requirements** tab for the Prism file is used to determine if these settings should be installed for just the current user or for all users. An exception would be if this setting in the file's properties is overridden by either the /CurrentUser or /AllUsers option with this command.

(For more information, see the Editor's **File Properties | Requirements** tab.)

Prism Self-Installing File Example

The following command makes sure that all the computer's users have the settings they need from this Prism Self-Installing file. It also updates the common settings.

```
MY_SIF.exe /AllUsers
```

Script Statements

In addition to the commands covered in Using the Command Line Options, scripts can contain blank lines and any of the statements described in this section. These statements help you control how a script is executed by running commands only when certain conditions are met.

elseif [not] [exists] <name1> [<operator> <name2>]

This statement has the same syntax as an if statement. The elseif statement is evaluated as true when the following conditions are met

- The if statement to which it corresponds had a false evaluation.
- None of the elseif statements, following the if statement and up to this point, had a true evaluation.
- The if statement to which it corresponds has not been followed by an else statement.
- The statement itself has a true evaluation.

When these conditions are met, any commands following this statement up to the next else, elseif, or endif statement are executed.

Notes

- Every if command can optionally be followed by as many elseif statements as desired.
- An elseif statement cannot follow an else statement.

Alternate Form

else if

Syntax

```
elseif [not] [exists] <name1> [<operator> <name2>]
```

not

You can reverse the logic in the statement following the **if** command by following it with **not**. For example if not filename = filename2. You can also use the symbol **!** for this option.

<name1>

The first item to be evaluated. If there is nothing following <name1>, the command is evaluated immediately.

- If <name1> is preceded by exists, <name1> is considered a file name. The statement is true if the file exists. If no path is provided, the current directory is searched for the file. The file name must have the correct extension.
- If <name1> is a number, the statement is true if the number is not zero. If <name1> is not a number, the statement is true if <name1> is not blank.

<operator>

One or two characters which define the way <name1> and <name2> are compared.

If <operator> is supplied, <name2> must be supplied as well. If <name1> and <name2> are both numbers, they are compared numerically. If either one is not a number, they are compared alphabetically, with the one that is first alphabetically considered less than the one that is last alphabetically.

The following operators are available

For the operator	The statement is true if
== (or =)	<name1> is exactly the same as <name2>
!= (or <>)	<name1> is not exactly the same as <name2>
>=	<name1> is greater than or equal to <name2>
>	<name1> is greater than <name2>
<=	<name1> is less than or equal to <name2>
<	<name1> is less than <name2>

<name2>

The item to be compared with <name1> to evaluate the statement.

else

This statement follows an if statement and one or more optional elseif statements. If an evaluation of true has not resulted from the if statement itself or any of the elseif statements, all the statements following this one are executed, up to the endif statement.

Notes

- Once an else statement has followed an if statement, no more elseif statements may follow.
- An if statement can only be followed by a single else statement.

endif

This statement follows an **if** statement. In between the **if** and **endif** statement there can be **elseif** statements, an **else** statement, and other commands.

Note: Every if statement in a script must be followed eventually by a corresponding endif statement.

Alternate Form

end if

if [not] [exists] <name1> [<operator> <name2>]

This command evaluates the entire statement to see if the commands following this one should be processed. An evaluation of true causes the statements between this command and the next else, elseif, or endif command to be executed.

Notes

- Every if command can be followed by one or more elseif commands and/or a single else statement.
- Every if command must eventually be followed by a matching endif command.
- If statements can be nested as deeply as desired.

Syntax

if [not] [exists] <name1> [<operator> <name2>]

exists

If exists is specified, <name1> is treated as a file name and <operator> and <name2> must not be present in the script statement.

rem

This statement indicates a comment. It is always ignored.

Sample Script

The following shows a sample script, using Prism commands. The script file extension is always .PTS.

```
rem ************************************************************************
rem Prism Script for Company XYZ
rem Created by: Administrator
rem Creation Date: January 5, 2004
rem Latest Revision: February 14,2005
rem
rem Description: This script checks the OS of the system and applies
rem Packages based on that OS (98 or W2K). The script also takes a
rem backup picture of the local "My Documents" folder.
rem
rem ************************************************************************

rem Set variable for Prism directory, define log file, define rules file

/Set PDDIR \\server1\PD\Packages
/DefineLogFile H:\logs\PD.log /size 2000
/DefineRulesFile %pddir%\rules.ini

rem Check the OS of the current system and apply OS-specific Packages
```

```
rem CheckOS.PWC will apply to 2000 systems only. If it errors out (lasterror = 4)
rem then the script will apply 98-specific Packages.
rem Else it applies 2000-specific Packages.
rem
/InstallPackage %pddir%\checkos.pwc
If %lasterror% == "4"
/InstallPackage %pddir%\98\set_computername_var.pwc /always
/InstallPackage %pddir%\98\virus_software.pwc /once
/InstallPackage %pddir%\98\suite.pwc /once
/InstallPackage %pddir%\98\faxclient.pwc /once
else
/InstallPackage %pddir%\w2k\virus_software.pwc /once
/InstallPackage %pddir%\w2k\suite.pwc /once
/InstallPackage %pddir%\w2k\faxclient.pwc /once
endif

rem Apply Packages that are the same for 98 and W2K.

/InstallPackage %pddir%\prephomedir.pwc /once
/InstallPackage %pddir%\createlocaldirs.pwc /once /identification %computername%
/InstallPackage %pddir%\mandatoryXYZshortcuts.pwc /always
/InstallPackage %pddir%\timecardapp.pwc /once
/InstallPackage %pddir%\latest_DATS.pwc /update

rem Take a Picture of the "My Documents" folder to use as backup.
rem The Picture occurs on Mondays and is named the current month and date.
rem If the file already exists for that day, the script exits.

If %pt_day% == "02"
If not exists H:\backups\%pt_month%%pt_date%.pwf
/TakePicture H:\backups\%pt_month%%pt_date%.pwf %pddir%\mydocs.pwi
endif
endif
rem ***************************************************************************
rem install a Windows installer file and exit the script if the %runerror% is not zero.
/Run msiexec /i \\server1\PD\Application.msi /qn /l*vx c:\MSIlog\Application.log
IF %runerror%  != 0
        /Exitscript %runerror%
endif
```

Index

T

U

V

W