

Symantec Software Management 7.1 Best Practices

**The white paper provides valuable information for anyone leveraging Software Management in the Symantec Management Platform (Notification Server) environment. Learn the tips and tricks for using Software Management successfully. Warnings, recommendations, explanations, and best practices are covered.**

Contents

[Introduction 4](#_Toc301948033)

[General Configuration 4](#_Toc301948034)

[Software Library 4](#_Toc301948035)

[Software Management Plug-ins 6](#_Toc301948036)

[Licensing 7](#_Toc301948037)

[Software Catalog 9](#_Toc301948038)

[Process Flows 10](#_Toc301948039)

[Software Resource 10](#_Toc301948040)

[Import Method 12](#_Toc301948041)

[Manual Creation 13](#_Toc301948042)

[Software Component Details 14](#_Toc301948043)

[Package 16](#_Toc301948044)

[Command-line 18](#_Toc301948045)

[Rules 20](#_Toc301948046)

[Static File Expression 25](#_Toc301948047)

[Associations 26](#_Toc301948048)

[File Inventory 28](#_Toc301948049)

[Software Publishing 30](#_Toc301948050)

[Quick Delivery 30](#_Toc301948051)

[Advanced Configuration 32](#_Toc301948052)

[Targeting and Scheduling 36](#_Toc301948053)

[Task Server 38](#_Toc301948054)

[Managed Software Delivery 39](#_Toc301948055)

[General Configuration 39](#_Toc301948056)

[Software Resource Settings 42](#_Toc301948057)

[Advanced options 42](#_Toc301948058)

[Task Settings 47](#_Toc301948059)

[Sequencing 48](#_Toc301948060)

[Scheduling 55](#_Toc301948061)

[Rules 58](#_Toc301948062)

[Manual Execution 58](#_Toc301948063)

[Software Publishing 59](#_Toc301948064)

[The Activity Center 59](#_Toc301948065)

[Software Categories 60](#_Toc301948066)

[Where is my software? 61](#_Toc301948067)

[Drag and Drop 62](#_Toc301948068)

[Managing Software Products 64](#_Toc301948069)

[Reviewing known Products 64](#_Toc301948070)

[Creating Software Products 67](#_Toc301948071)

[The Software Portal 69](#_Toc301948072)

[Common Terms and Acronyms 69](#_Toc301948073)

[Software Publishing 70](#_Toc301948074)

[Software Resource Publishing 71](#_Toc301948075)

[Managed Software Delivery Policy (MSD) Publishing 72](#_Toc301948076)

[Software Request 74](#_Toc301948077)

[User Resources Intro 74](#_Toc301948078)

[User Interaction 76](#_Toc301948079)

[Requests and Updates 77](#_Toc301948080)

[Request Dialogs 78](#_Toc301948081)

[Software Approval 81](#_Toc301948082)

[Portals 81](#_Toc301948083)

[User Resources 84](#_Toc301948084)

[Filtering the Admin or Manager Portals 86](#_Toc301948085)

[Quick Delivery or MSD fulfilment 89](#_Toc301948086)

[Listed Software 89](#_Toc301948087)

[Managed Software Delivery Policy 90](#_Toc301948088)

[Populating the Portal 90](#_Toc301948089)

[Tables in Use 92](#_Toc301948090)

[Known Issues 93](#_Toc301948091)

[Tracking Deployments 93](#_Toc301948092)

[Task Server Statuses 94](#_Toc301948093)

[Provided Reports 95](#_Toc301948094)

[Conclusion 98](#_Toc301948095)

**Legend**



– Warning – These tips provide you warnings for known configuration issues and other items that may be important to know.



– Best Practices – From experience the Best Practices tips provides you methodology or considerations when using the applicable features.

# Introduction

Software Management in Notification Server 7.0 underwent the largest transformation of any of the products available from the 6.0 architecture. New functionality gave expanded abilities and provided some real ROI for the whole of Software Management. From new Detection capabilities to intelligent monitoring of Software, Software Management offers a large toolbox for not just deploying software, but for managing software.

In 7.1 the ease and ability of using Software Management was greatly enhanced by the introduction of the Silverlight interface (known as the Activity Center, a.k.a Enhanced Console Views). Feedback taken from users of 7.0 guided much of the changes found in version 7.1.

Back in 7.0, due to this influx of new functionality, the first release required a steep learning curve, and as users starting implementing Software Management the idiosyncrasies and problems began to surface. In 7.1 most of these early problems have been addressed, yet many these same configuration lessons learned in 7.0 apply to 7.1 as well. The potential still exists for problematic configurations. This document attempts to address these points, provide guidance for the new features and interfaces, and provides warnings to avoid the potential pitfalls, and provide configuration details to enable successful Software Management.

Best Practices are offered based off understanding of typical goals. In other words these are offered as they fit the majority of use cases. It is possible an environment will not cater to these Best Practices. As such the functionality behind the configuration options are explained, where possible, to help you make your own informed decision on how it should be configured for your environment. This document is provided “As Is” and implies no warranties, guarantees, or supportability.

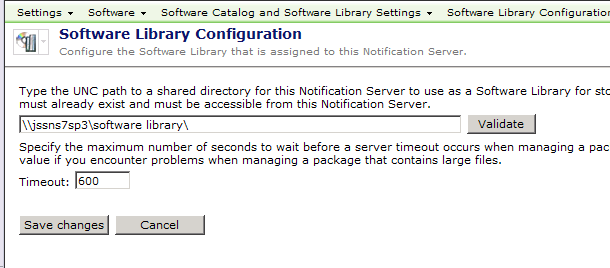
# General Configuration

For effective use of Software Management, it’s important to set all the general settings as needed for your environment. Though some of these settings won’t be known until later, I’ll cover all general settings for reference.

## Software Library

The Software Library is a location where all Packages are stored (when Software Library is selected as the Package Source). The Software Library has two requirements before it can be used successfully. One is a one-time only setting, and the other is required on any system that needs to run the Console in conjunction with the Software Library.

1. Software Library Location – This should be a location that has a lot of storage, or at least sufficient storage for all packages to be managed within the infrastructure. To set the location, follow these steps:
   1. At the desired location where you wish to store the packages within the Software Library create a base folder. When packages are added they will be added to this folder as subfolders.
   2. The folder requires a share that the Application Identity can access (preferably all administrators on the system should have access). It is recommended to keep the share name short for ease of configuration.
   3. In the Symantec Management Console, browse under Settings > All Settings > Software > Software Catalog and Software Library Settings > Software Library Configuration, as shown below:



* 1. Set the location as a UNC path to the desired location.
  2. Click Validate to ensure the path is set correctly. Reasons it may fail are:
     1. The Altiris Application ID, or the Account associated with NS, may not have rights to the UNC specified
     2. The UNC is not a valid share
     3. The UNC cannot be reached (DNS or other related network issues)
  3. Click Save changes to commit the settings once validated.

 Note the timeout value set here. For very large packages we have seen instances where this value should be increased beyond the default 600. Also note that very large files can also reach the timeout value when importing into the library.



If your Software Library resides on the same disk as a Package Source, the files will be duplicated into the Software Library share, taking twice the necessary disk space. If the package is local to the Software Library, you can use a UNC Package Source as the source instead of the Software Library, or mount the package in another location to use as a source. The benefit for having a separate source outside of the library is for package integrity.

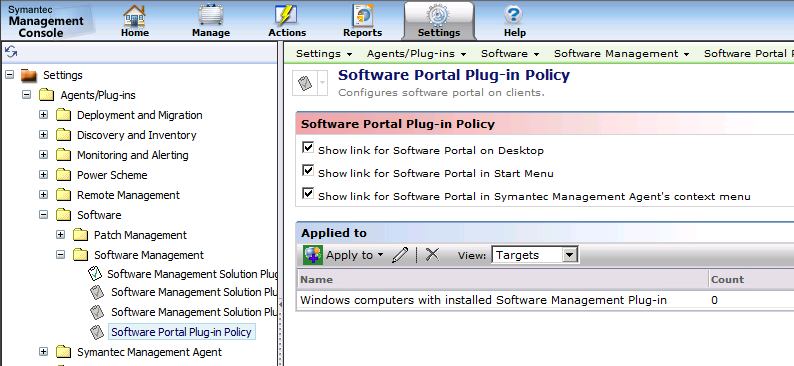
1. Java Runtime Environment – To use the Software Library option when configuring a Package, Java Runtime Environment (JRE) is required. You can go to [www.java.com](http://www.java.com) to obtain the free download of JRE.  
   *NOTE: the JRE needs to be installed and available on any system that runs the Symantec Management Console where Package creation is to be used.*

## Software Management Plug-ins

There are four Plug-ins for Software Management. The first and second ones are automatically installed as part of the Altiris Agent. The first is called the Software Management Framework Plug-in (SMFAgent.dll). This one is used closely with the second plug-in, which is the base Software Management Agent. This allows all Solutions and the NS itself to make use of the Software deployment and management capabilities for the distribution of packages, policies, and certain task types.

To extend the functionality for use within the Software Management Solution, the Software Management Solution Plug-in needs to be installed. The last “Plug-in” is the configuration of the Software Portal interface.

The following procedure walks through configuring and pushing out the Software Management Solution Plug-in and the Software Portal Plug-in Policy:

1. In the Symantec Management Console, go to Settings > All Settings > Software > Software Portal Settings > and select Software Portal Settings.
2. Set the maximum number of Open requests in the right-pane to a reasonable number of your environment.
3. If desired, change the Company Name and Logo to brand the Software Portal to your Company.
4. Click Save changes when set correctly.
5. In the Symantec Management Console, go to Settings > Agents and Plug-ins > All Agents and Plug-ins > Software > Software Management > Windows > and select Software Management Solution Agent for Windows – Install.
6. Normally the default filter suffices, but this can be changed at this time if required, as well as the schedule.
7. This Policy controls the rollout to ALL platforms, including Windows, Mac, Linux, and Unix. If you wish to control the Plug-in rollout separately, the Filter needs to be changed to reflect which platform to target. You can clone the Policy and configure for another platform, etc.
   1. NOTE: It is not recommended to try and edit the existing filter, since that filter is used in other places. If you wish something different, create your own filter and add that as the target.
8. Enable the Policy and Save changes.
9. In the same folder, now select the Software Portal Plug-in Policy. This screenshot shows the policy:  
   
10. Select the options for where you want the link for the Software Portal to appear. Options to consider are:
    1. If your Symantec Management Agent tray icon is hidden, uncheck the option for “Show link for Software Portal in Symantec Management Agent’s context menu”.
    2. The Start Menu is a popular choice as it’s out of the way but easily understood by typical end-users.
11. It is recommended to keep the current target as that enables the Software Portal on all Managed Systems.
12. Done! The Agent will now be deployed as the target computers check in for a new Configuration. As this is policy based it may take a number of hours for all machines to run the policy and install the Plug-in.

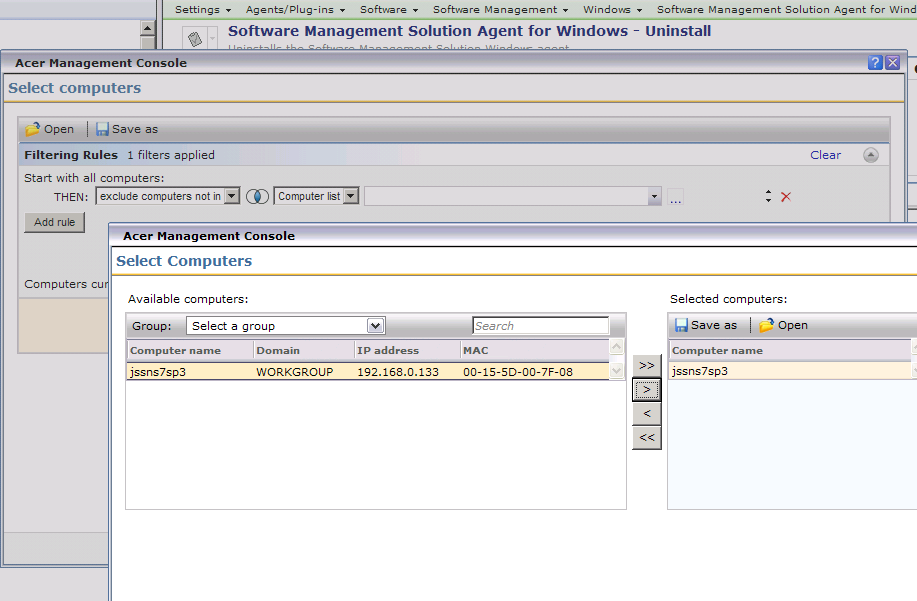
## Licensing

Protection against unauthorized use is required for any Software Company. Licensing ensures that only authorized organizations (AKA those who have purchased) can use the software, and that they only use it within the scope of their license. While required in some form or another, it is not desired that licensing get in the way of successfully using the software if properly licensed. For the Software Management Framework, no license is required. For the Software Management Solution, a valid license must be installed in order to use it.

Software Management Solution counts licenses by the number of computers that have the Software Management Solution Plug-in installed. The count is figured off of the table Inv\_AeX\_AC\_Client\_Agent which Basic Inventory populates. The count is based off of the Agent Name: Software Management Solution Agent and the corresponding Agent Count column.

Keep a tab on how many computers have the Software Management Solution Agent installed. Visit SIM in the Licensing section, and ensure you don’t roll the agent out to too many systems. If you need more licenses ensure you receive them and install them before you roll additional Plug-ins out.

If your Software Management Solution license has been exceeded, the following action can be taken to free up licenses.

1. In the Symantec Management Console, go to Manage > Filters > Software Filters > Agent and Plug-in Filters > and select All Windows Computers with Software Management Solution Agent Installed.
2. Click Update membership to ensure the Filter is up to date.
3. In the list of computers, find those computers you wish to remove a Software Management License from. Make a list of these computers  
   *NOTE: If you wish to make a dynamic filter, then it is unnecessary to create a list.*
4. In the Symantec Management Console, go to Settings > All Settings > Agents/Plug-ins > All Agents/Plug-ins > Software > Software Management > Windows > Software Management Solution Agent for Windows – Uninstall.
5. Under the Applied to section remove the default Filter by selecting it and clicking the red X delete button.
6. Now add the filter of the systems you wish to remove the Plug-in from. If it is a dynamic filter you’ve created, even better.
7. If you wish to use the list you acquired in step 3, follow these steps:
   1. Click Apply to > Computers.
   2. Click Add Rule.
   3. Change Then to Exclude computers not in, and the next field to Computer List.
   4. Click the ellipses <…> at the end of the row to open the computer selection dialog.
   5. From the left-hand pane select the computers to remove the Plug-in from. Note that you can use the filter to find systems if you have a large list.
   6. Add your selects by clicking the > button (or if you have filtered down to the list you wish, use the >> button). See this screenshot for a sample:  
      
   7. Click OK to apply the new filter.
8. Enable the policy, aka turn the policy to the ON status.
9. Click Save changes to save the application of the new filter.
10. Done!

As the targeted computers update their configuration, they will receive the new uninstall policy and remove the Plug-in. It then requires those targets to send an updated Basic Inventory. To finalize the process, the License Refresh must occur. The entire process can take time to propagate to all targeted systems.

## Software Catalog

There are a number of other settings/configurations that can be utilized by the Software Catalog. The following items are available and can be used as desired. For most general use they do not necessarily require configuration beyond the defaults. To locate these items, in the Symantec Management Console go to Settings > All Settings > Software > Software Catalog and Software Library Settings.

1. Clean up File Resources – This is an automatic process that reconciles data captured from multiple computers to synchronize file resources, if needed.

Best Practices!: The results of Software Discovery, called Add Remove Programs data or Installed Software, may show incomplete data after machines have reported Inventory. If this occurs, manually run the Clean File Resources task to correct the problem.

1. Installation Error Code Descriptions – There is a large default list (600+ entries). Only if you wish to clarify specific errors or add your own should you use this feature. It can be nice is large environments where the IT professionals working on issues may benefit from more verbose error messages.
2. Known As – This allows catalog information to be correlated from Company Name to a corresponding resource. This also allows multiple versions of a company’s name to map to the same resource. For example:
   1. Microsoft > Microsoft
   2. Microsoft Inc. > Microsoft
   3. Microsoft Corp > Microsoft
   4. Microsoft Corporation > Microsoft
   5. Also note that there is a Wildcard section to catch potential exceptions. Use “Microsoft” in the search field to see a list of those items already mapped.
3. Software Discovery – Software Discovery is executed by the Software Management Framework Agent (SMFAgent.dll), but is executed by Inventory Solution. Please see the Inventory Solution section of this document for more information. As a general rule, this built-in policy should be left disabled.

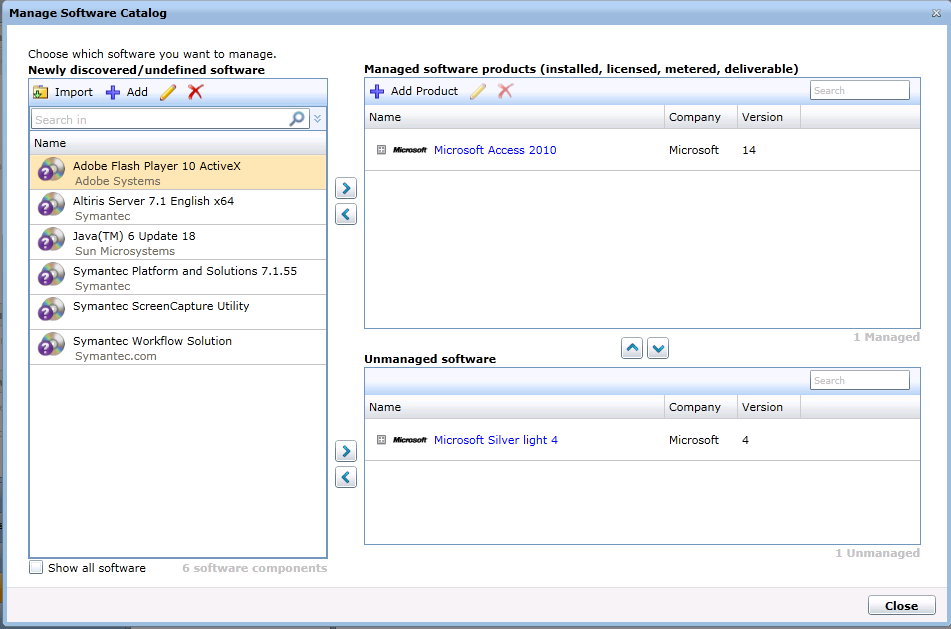
## Process Flows

The following Process Flows show how different mechanisms work with Software Management. These will be referenced in the subsequent sections as needed.

# Software Resource

A Software Resource houses Package and command-line information, and contains a myriad of metadata for a specific piece of Software. This includes Package information, or the physical files, the command-lines that can be executed against the package, detection and applicability rules, file information, associations with other Software Resources, and Software Publishing information. The culmination of this data makes up a single Software Resource, also known as a Software Component, in the Notification Server infrastructure.

Software Resources are organized in the Symantec Management Console under Manage > Software > Software Catalog. The Catalog view within the Enhanced Console View (ECV) contains a list of Newly discovered/undefined software, of Managed software products (installed, licensed, metered, deliverable), and Unmanaged software. See this screenshot for an example:



This is only a condensed view of the Catalog, typically software that is active in your environment. Actions you can do from this view are:

* Import – Import new Software via the installation files.
* Add – Manually create a Software Resource.
* Edit – Edit an existing Software Resource
* Add Product – Create a new Product   
  *NOTE: Products are how licensing and usage are tracked, so proper Product configuration is essential*

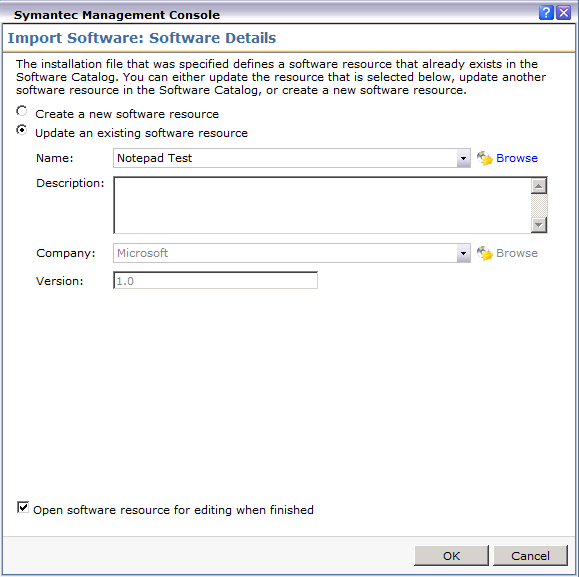
To view the entire Catalog in a single view, browse under Settings > Console > and select Views > browse under Views > Software > and select the Software Catalog. This gives you a full view of all Software Resources in your environment. To view software that can be delivered, browse in the Symantec Management Console under Manage > Software. This will load the Enhanced Console View (ECV). In the left-hand pane labeled “Deliverable Software”, choose either Software Releases or Software Updates. The items that populate the center pane have associated Packages and/or command-lines.

There are several ways Software Resources are created. They can be imported manually (this process is covered in the next segment), manually created, created by Inventory Solution running Add Remove Programs Inventory or alternatively via the Software Discovery Task, or imported through a data provider.

## Import Method

The best way to add deliverable software to your Software Catalog is to use the Import function. The import will auto-generate much of the fields needed for a Software Release, including a command-line (or multiple command-lines if using an MSI), version information, Package details, and if you are using an MSI a Detection Rule of MSI Product Code type is automatically generated.

Follow these basic steps to import Software into the Software Catalog. Please also review the manual steps as it will illustrate what is being done during the automatic import process.

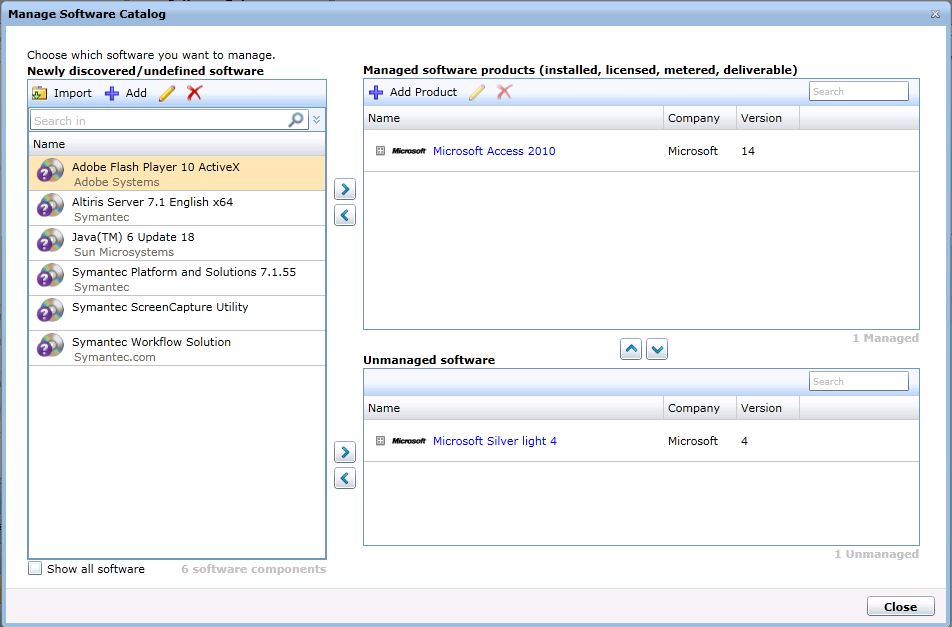
1. In the Symantec Management Console browse under Manage > Software. This will load the Enhanced Console View (ECV).
2. Within the Installed Software pane in the upper left, right-click in the white space and choose Import Software.
3. Select the appropriate Software type (most will be Software Release).
4. Choose an appropriate Source, such as the Software Library.
5. Click the Add button to browse to the source installation files of the software.
6. When you select a directory, this directory should only contain the files for this specific software. This is because the selected folder becomes the package root, and all files, folders, subfolders and files will be included in the Package.
7. The Package contents will be populated. Note that the Console will attempt to auto choose what file is to be the default Installation File. Use the “Set Installation File” button if you need to choose a file other than what was automatically chosen.
8. Click Next to create the Package.  
   *NOTE: this process may take some time if your files are overly large (GBs in size) or there are many files within the package. This is due to the hash created for each program file found within the package.*
9. The following screen allows you to either create a new Software Resource, or update an existing one, as shown in the following screenshot. If the import detects that an existing Software Resource appears to match what is being imported, that will be the default option.  
   
10. Check the option to Open software resource for editing when finished if you wish to review what was imported, make any changes, or add additional information.

From this point on any modifications will be covered in the manual process for created a Software Resource in the subsequent sections.

# Manual Creation

The following process runs through each configuration section of a Software Resource, and how it is configured. The process will help illustrate how each set of functions operate, and how they can be configured to meet your specific use-case.

## Software Component Details

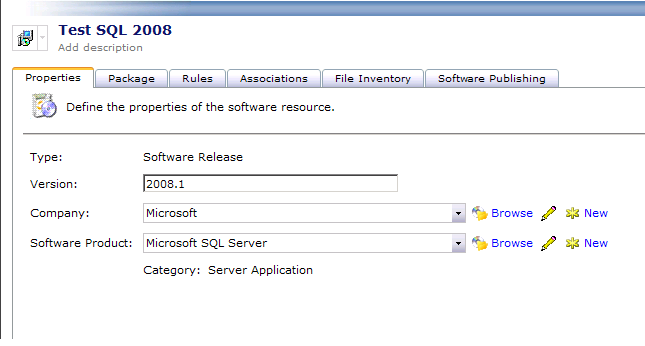
1. To create a Software Resource, log into the Symantec Management Console and browse under Manage > Software. If you are already in the Enhanced Console Views (Silverlight) you can click the Software option in the lower left-hand pane.
2. Right-click in the white space found under the Installed Software pane in the upper left and choose Managed Software Catalog. This will bring you to the Catalog management screen, as shown:  
   
3. Click the Add button on the icon bar for the left-side pane and choose the appropriate Software Resource type from the list:
   1. Software Release (most common)
   2. Service Pack
   3. Update
4. Fill in the resulting fields using the guidance below:
5. Name – This is the label or name that this Software Resource will show up as in lists or pickers.
6. Type: This will list what type of Software Resource it is from the above list.
7. Version: This field affects how the versions are calculated for a Product. It is recommend to use an approximation of the version of the application you are creating the Software Resource for.

One function of this version field is to check against changes made to the Resource. For example if the Version number changes, the components within the Resource are invalidated and must be refreshed. This means any changes you’ve made to the components of the Software Resource will go through the update process on the Notification Server and can expedite updates to the targeted clients.

1. Company: Used to put the manufacturer of the Software contained within the Resource. This affects reporting by Manufacturer so it is highly recommended to use a consistent naming scheme, and avoid differences such as:
   * Microsoft
   * Microsoft Inc.
   * MSoft
   * MS
2. Software Product: This ties the Software Component to a Product, as managed in the Symantec Management Platform.



In Software Management 7.1 most software actionable functionality is pointed to the Product level. It is highly recommended to use a consistent assignment to existing Products within the system. Application Metering, Asset Management, Licensing, and more are calculated against the Product level.

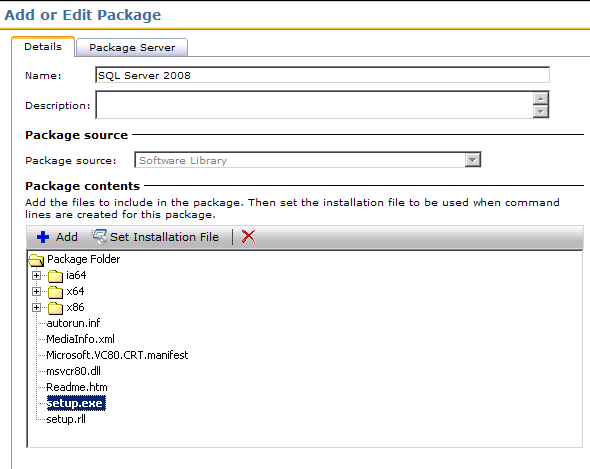
1. See the below screenshot for an example of the Properties.  
   

## Package

The Package within a Software Resource is a collection of physical files and folders (if you can really call a digital file “physical”). The most basic explanation is a Package is a folder and all files and subfolders/files therein. For all Package Source types except for the Software Library, this is all a Package is. The Software Library option adds another layer to include the default execution file no matter where it is contained within the folder structure of the package.

The Software Library is the default selection for a Package Source. A Package can be configured to use one of the other options, such as UNC, URL, or a local directory on the NS. See the previous section on Importing Software for details on how to do this.

To aid in understanding what source is best for you to use, the following details can help you make a decision. Please note that the Package Source does not necessarily indicate where a targeted client will obtain the Package from. It’s where the Notification Server will provide the Package from.

1. Access package from an existing UNC – The Notification Server will provide the UNC to any Package Server or Client that requests the Package. An HTTP link is also created off the UNC to be used to distribute the Package.
2. Access package from a URL – This reacts the same way as UNC only flip-flopped.
3. Access package from a directory on the Notification Server – The Notification Server will turn the designated folder into a UNC and URL share to provide to all Package Servers or Clients that request it.
4. Software Library – The Notification Server will import the package when configured. This process does the following:
   1. The contents of the folder (files and subfolders/files) to the location specified in the Configuration of the Software Library covered earlier in General Configuration section.
   2. Each file will be scanned. All EXE, DLL, or other programmatic files will be cataloged as part of the Software Resource.
   3. A hash will be generated for each of these files.
   4. A default Installation File will be set. The below screenshot shows a sample of this. Note that the Installation File will be bolded. By default the import process will try to analyze the most likely file and automatically set it. It is advised to check and ensure the right file is selected. To manually set it, select the file and click the Set Installation File button.  
      



The Package Import process can take a long time depending on the size of the package. While there are a lot of factors that contribute to this, including network speed between the NS and the Package Source and Resource availability on the Notification Server, I’ve seen large packages take up to 45 minutes to import. The specific IE session will be locked during this process, so open a new console if you need to continue working in the meantime. In 7.0 we saw issues where the Import would time out, but in later version and in 7.1 these have not been reported due to improvement and efficiencies introduced into the product.

Click on the Package Server tab. The default configuration is set to All Package Servers with Manual Prestaging. Despite the name, this feature basically means the package will not be delivered to Package Servers until a client has a policy that needs it.

This recommendation comes with a caveat. If you have a large distributed environment, you may not want all your Package Servers to host every package until they are needed. If this is the case the default should be OK. Otherwise, follow these steps:

1. Click on the Package Server tab after you have configured the Details tab of the Package.
2. Under the dropdown labeled Assign package to, change the option to All Package Servers.



If you leave the selection on Package Servers with Manual Prestaging, be aware of the implications of this. Package Servers will not download the package until another Package Server or Client is services requests the Package. This can cause a Package or Quick Delivery task to timeout due to the time it takes to stage the package and then download it. *Quick Delivery task timeout’s include the download time.*

Note you also have the option of specifying an alternate location for the Package Server to download the Package locally using the check box labeled: Package Destination on package servers (leave blank for default).

## Command-line

Consider the Command Line as the action against a package. After a Task or Policy has been delivered to a target, after applicability and detection rules have been executed, after the package has been downloaded, the execution occurs, as configured in the selected command-line. Obviously the command line is a string that executes against a particular file, but there are other aspects to the Command-line Object within Software Management to be mindful of.

The following walkthrough takes you through configuring the Command Line:

1. Under the Package tab of the Software Resource click the Add command button.
2. Provide a Name for the Command-line. If desired, provide a Description.

It starts to feel like there are a bazillion names within a Software Resource. This includes names for the Software Resource itself, Package(s), Command Line(s), Association Rule(s), Detection Rule(s), The Task(s) or Job(s) applied to it… after a while you may be tempted to put in generic names. This will make it difficult when managing the individual objects. Use a naming system you can easily use to recognize what an object is intended for.

1. The option Command-line requires a package can be unchecked if you wish to run a command-line against an installer that already exists on the target computers. For this example we are leaving it checked.
2. Set the Installation file type to the correct type. This will enable the Command-line builder for that type of Installer file.
3. Depending on what the command-line is for set the Command type appropriately. See below for an explanation of the settings:
   1. Custom – This can be used for anything as a catch all for any execution that doesn’t fit the other three options.



With a Custom type selected you cannot make it a default command line for the Software Resource. This also affects how a Detection Rule interacts with the command-line. For example the logic for detection of an install versus uninstall is not used with a custom command line. When an Install command line type is set, the detection status “Not Detected” initiates the remediation, whereas with an uninstall command type the detection status “Detected” will initiate the remediation.

* 1. Install – Typically used when the command-line will install an application. This will also be the default for a Software Portal request.
  2. Repair – If the command-line initiates a repair of an application, such as a corrupted MSI installation, use this option.
  3. Uninstall – Used for uninstalling the Software Resource.

1. Check the option ‘Set as the default for this command type’ if required. This will affect how “drag and drop” default settings are selected in the Silverlight interface.
2. Enter the command-line including all applicable switches.

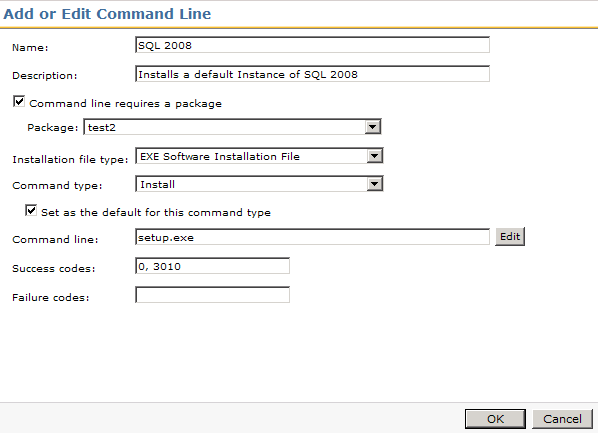
Often when time is pressing an administrator will not test the command-line before rolling out Software into Production. It is recommended to always test the command line before adding it to the Software Resource. The following items should be considered when generating a command line:

1. Can you successfully launch the installer using a command-prompt at a similar system to the ones being targeted?
2. When is executes, does it prompt the user for anything? This is important if the installation will be hidden from the user. If it is hidden, and there is a prompt, the prompt will sit in limbo until the Task or Policy is killed.
3. What rights are required to run this specific command-line? Make sure the job or task you use to roll this out is set with an account that can execute the command line successfully.

By testing, you can avoid rolling out badly configured Command-lines.

1. The success codes should be configured to contain those installer return codes that you wish to equal success. A common return code considered successful is 3010, which means the installation was successful but a reboot is required.

If the field is left blank, the default success code is 0 only. As soon as you enter a return code into the field, 0 is no longer considered a success unless it is explicitly added to the field, as shown in the following Screenshot.   
Please add 0 if you are adding other return codes to this field.



1. Inherently all codes that are not success are considered failures. However if you add specific failure codes here, they can be used in Task Server Jobs as part of the conditions. For example you may want one failure code to result in one action, and a different failure code in a different action.
2. Click OK to add the new command line.
3. Click Save changes to save the new command-line to the Software Resources.

Please note that I have asked the Resource to be saved after each major item is created. This will help save your progress so if a problem occurs in IE or on the server you will have much of your work saved.

You can add any number of command lines you need to the Software Resource. Typically you can create both an Install and Uninstall command line, and, if available, a command line that repairs the application.

## Rules

To add intelligence to the Software Management Process, a myriad of Rules have been provided. Rules are used for two primary purposes:

1. Detect whether or not a specific Software Resource is present on the target system.
2. Check to see if the specific Software Resource is applicable or not for the target system.

The two rules combined results in dynamic management of configured software applications. For example if I have an Application that has one for a specific version of windows and one for another, I can create one Managed Software Delivery Job that contains both the versions of the application. The detection rules would check to see if the Application is there, and the applicability rules would ensure the right operating system is in place. The result would be that the right version of the Application would be installed on the right version of Windows.

Both Detection and Applicability Rules use the same Rule types. The difference is that when a Detection Rule is marked True, the Software Resource is NOT applied unless it is an uninstall command line. When an Applicability Rule is marked as True, the Software Resource IS applied. If you think of it in the paradigm I provided above you can see how you would use the two rule types together. It is not necessary to use both or even one of these rules, but it does add intelligence to the rollout. While all Rules can be either type, some cater better for one type. See below for explanations.

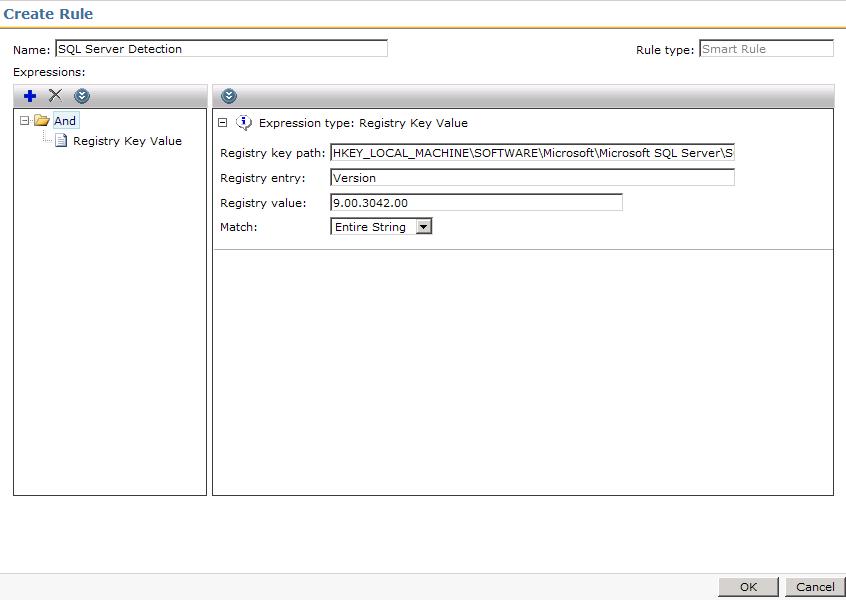
The following table helps illustrate what occurs when a Rule is evaluated in different configurations:

|  |  |  |
| --- | --- | --- |
| **Command line Type (for Detection)** | **Rule True** | **Rule False** |
| Detection True - Install |  | Remediation |
| Detection True - Custom or Repair |  | Remediation |
| Detection True - Uninstall | Remediation |  |
| Applicability Rule | Remediation |  |

Here is a list of Rules and their known practical use:

1. 64-bit Windows Installed – This Rule tells the Inventory Rule Agent to check the 64-bit sections of the Registry and File System, respectively, included in the base rule.
2. File Version – This can ensure that an update only applies to systems with the right version of an application installed. Conversely it can be used to see if the specific Software is already installed or not.
3. MSI Product Code – For MSIs this option can be used to detect the presence of the application associated with the MSI. Since the MSI Product Code will be different for different releases, you could use this to detect and apply upgrades. Note that when you Import an MSI via the Import method to create a Software Resource, the MSI Product Code Detection Rule is automatically created for you.
4. Multilinqual User Interface Installed – I have not used this one, but it can assist in targeting those systems that have language packs installed that may require different versions of a patch or software.
5. Processor Type – There are three options available. These point to actual hardware concerning the platform type. The three options are:
   1. x86
   2. Intel Itanium Processor Family (IPF)
   3. x64 (AMD and Intel)
6. Registry Key Exists – I consider this a more general Rule to the Registry Key Value. This simply looks for the existence of a key, but not the value of that key. For example if you have the Office 2007 Compatibility Pack to deploy you may simply want to check for the existence of a major version of office and not be concerned on the minor updates or service packs.
7. Registry Key to File Version – As the name implies this allows a check against a version of a file if it is stored in the registry. The operators allow you to specify how the rule works with found values.
8. Registry Key Value – This is a common rule that is generally used. In the below example of creating and configuring a Rule, this Rule will be implemented.
9. Registry Key Version – I recommend using the Registry Key Value as it is more straight-forward. If the version is not a String, however, this Rule should be used.
10. Registry Key/File Path to File Version – This is a more complex rule that allows you to compare a registry entry with a file version.
11. Registry Key/File Path to Product Version – This rule allows you to check a Product Version as found in the Registry.
12. Windows Language – This rule allows you to target only the language you need to, whether for an application installation or an update that is language specific.
13. Windows Version – This allows you to ensure that either you are running against the right version of windows, or that you are not running against the wrong version of windows. This is especially useful for Windows Updates.
14. Software File Expression – This relies on the Installed Software inventory data stored in an XML cache locally on the client. If the specified entry is found in the XML, the rule applies.
15. Static File Expression – This allows the check for a physical file on the file system. The path should include the filename. Also take special note of the Version field. Leave blank if the version is not applicable.
16. Static Shortcut Target – I have not used this rule.

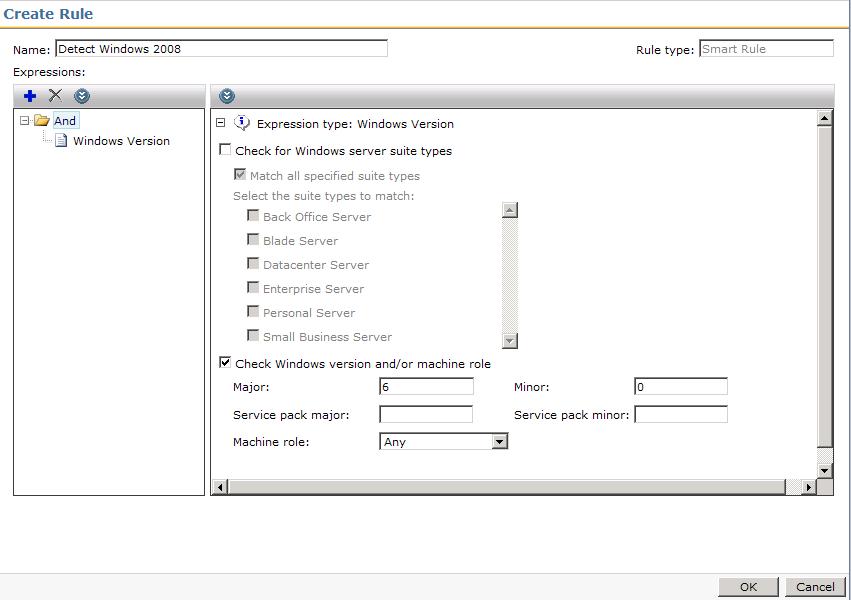
For the Detection Rule example, we’ll use the Registry Key Value option since it is very dynamic in its uses.

1. Launch the Editor for the Software Resource you wish to add the Rule to.
2. Under the Rules tab, click the New button (yellow asterisk) next to the Detection Rule dropdown.
3. Name the rule so it is easily recognizable. For example SQL Server 2008 Detection.
4. Click the Add a new rule expression button (the blue plus **+**), move under the Standard Rule section, and click on Registry Key Value.
5. Provide the Registry path. One way to ensure path reliability is to browse in the registry on a system that has the software installed, using the following steps:
   1. Go to Start > Run > Type regedit > Click OK.
   2. Browse in the Registry hives in the left pane to the Key you will be evaluating for the rule.
   3. Right-click on the key and choose Copy Key Name.
   4. Go back to the Rule setup and Paste in the copied key name. This will give you the proper path.
6. Provide the Registry entry. This is the name of the value within the Key specified in the previous step.
7. Provide the Value of the key.
8. The Match criterion is defaulted to Entire String. To make this rule more dynamic, you can choose Substring to allow partial matches. For example if you want all versions of SQL matching 9.00, you can put 9.00 as the value and any version, such as 9.00.3042.00, will be considered a match.
9. Click OK to add the criterion. This screenshot shows an example:  
   
10. Click OK to save the Rule. Note that you can have multiple detection points for a rule. This allows more granularity to ensure the target system correctly evaluates if the Software in question is installed or not.



The Registry rules are provided only for String values. DWORD or other value types are not supported and will not correctly fire.

For the Applicability Rule example, we’ll use the Windows Version option as this is a common Applicability Rule I’ve seen used.

1. Launch the Editor for the Software Resource you wish to add the Rule to.
2. Under the Rules tab, click the New button (yellow asterisk) next to the Detection Rule dropdown.
3. Name the rule so it is easily recognizable. For example SQL Server 2008 Detection.
4. Click the Add a new rule expression button (the blue plus **+**), move under the Standard Rule section, and click on Windows Version.
5. You can select a specific Windows server suite type using the checkbox and options under Check for Windows server suite types.
6. Check the box Check Windows version and/or machine role.
7. In the Major field place the major version number for the version of Windows you are targeting. In this example we’re using Windows 2008, the value of which is 6.
8. In the Minor field put the minor version number. In this example 2008’s value is 0.
9. Click OK to save the criteria. The following screenshot shows the configured Rule:  
   
10. Click OK to save the Rule. As with a Detection Rule you can specify multiple criteria.

Rules add great intelligence to Software Management. It is highly advised to test a Rule before rolling it out to ensure the rule accurately detects, or NOT detects, the given Software it is associated to. Testing both True and False for the rule will ensure it doesn’t detect software as being there when it isn’t, or detects it as not being there when it is.

Another note concerning Rules, how you use the Operators (And, Or) is important. When using the tree, you must put rules underneath the operator you wish to use. For example, this setup will not work:

AND (Folder)

* Detect Key
* Or
* Detect Key

All 3 options are directly under the AND operator so the system believes that all detections must be detected to be compliant. That is the reason for the ‘Not Detected’.

In order to build the detection rule correctly the detection rule must be define as:

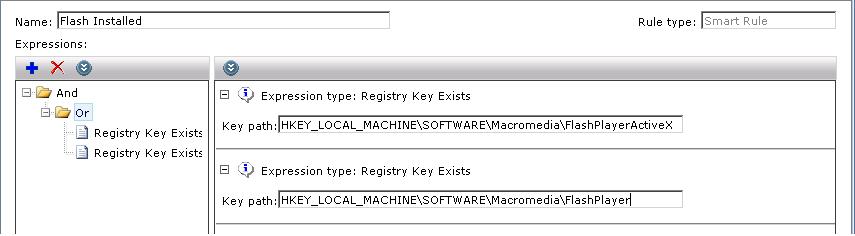
AND (Folder)

  -OR (Folder)

    -Detect Key

    -Detect Key

Now both Key detections are part of the OR statement and the detection works successfully. Here is a screen shot of my successful detection rule. In order to add the Detection Keys under the OR statement you must high light the OR statement before clicking the + button. Or you can right click the OR statement and choose ‘Add’.



This detection rule worked successfully when either of the keys existed.

### Static File Expression

This Rule has a few tricks to it in order to get it to work properly. Please note the following when using this rule type:

1. The Base folder does need to be selected, and the File path will need to originate in the Base folder, meaning it is a relative path to where the base folder is located. For example:
   * Base folder: Program Files, path: Altiris\Altiris Agent\AeXNSAgent.exe
2. The File Path should include the name of the File to be used in the Rule, as also shown above.
3. The Version can remain blank. This has the most success rate as the qualifiers can easily be misused. A Blank version will look for any version of the indicated file.
4. It is recommended to stick to = if possible if version is needed. This takes out the guess work on how to configure the version number (ie 7.0 versus 7.0.1154.5).
5. For the Software Resource this detection rule is a part of, you need to add the file in question under the File Inventory tab. This was required until 7.1, where we have had success without having to add the file to the File Inventory tab, however if you run into problems it is a good troubleshooting step.

## Associations

Associations allow the linking of different Software Resources. For example if you have a Software Resource for SQL 2008, you can create a second Software Resource for SQL 2008 SP1 and select it as an Update to the SQL 2008 Resource. The following Association Types are available:

* Conflicts with – Using this Rule you can avoid installing Software that has known compatibility issues with existing Software. This will help Administrators know what Software should not be installed together on a target computer.

Conflicts with is a messaging feature only, and will not stop an execution if the resource referenced in the Conflicts with is installed on the target systems.

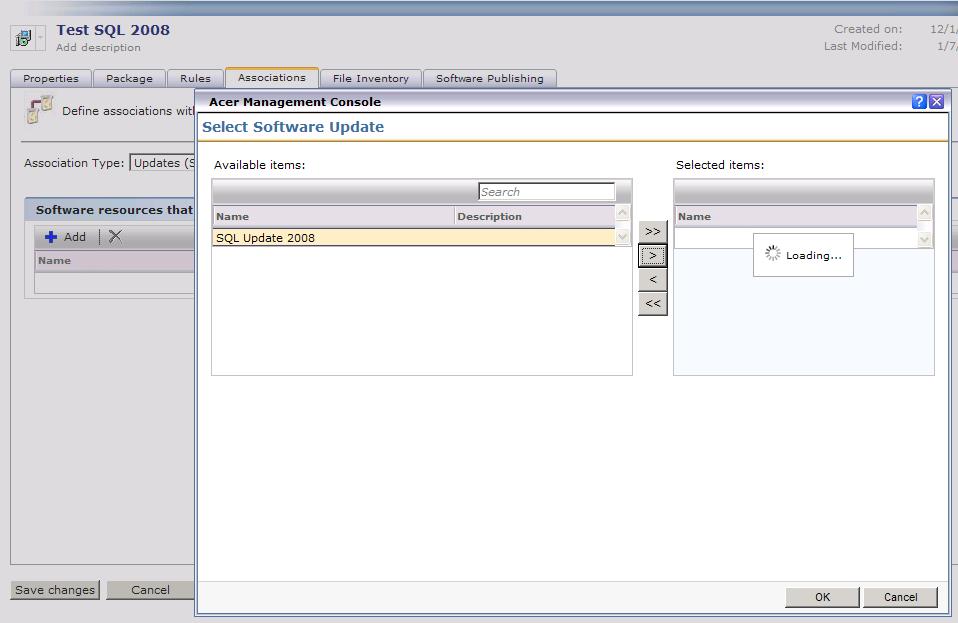
* Contains – To help manage what updates or resources contain multiple items that may already exist as stand-alone, use the Contains Association to track these.
* Depends on – This Rule can automate the running of Prerequisites for the Software Resource. For example if a particular application requires the JRE (Java Runtime Environment) you can create a Software Resource that deploys JRE. That JRE resource will be referenced as a Depends on association to the primary Software Resource.



Depends on will install the dependent software if this option is selected within the associated Managed Software Delivery Policy when pushing out this software. This can remove any guesswork for ensuring systems have prerequisites.

* Software Channel Targets Software Release – I could not find a use-case or documentation on this Association.
* Supersedes – This is particularly applicable for roll-up hotfixes or updates. You can select all updates that are contained in the rollup by using this Association Type. This, like Conflicts with, is informational only and does not result in different results when rolling the software out.
* Updates (Service Packs) – You can add Resources that add a Service Pack to the editing Software Resource. With these associations you can automatically update your install base. The Managed Software Delivery Policies referencing the Resource can have the Service Packs added. Both Updates and Service packs can be selected when scheduling the associated Software Resource for execution, ensuring that with each install the needed updates are applied automatically.
* Updates (Software Updates) – This is the same type of Rule as Service Packs for individual Updates to the Software Resource.

For this article I’ll detail adding an Update to a Software Resource. The following example adds an update to Microsoft SQL Server 2008.

1. Launch the Editor for the Software Resource you wish to add the Association to.
2. Click on the Associations tab.
3. From the Associations Type dropdown select Updates (Software Updates).
4. Click the Add button under the heading Software resources that update this software resource.
5. In the resulting window use the search function to find the Update to apply.
6. Once found, highlight the update in the left-hand pane and click the greater-than symbol > to add that update. See the following screenshot for an example. You can use multi-select with Ctrl to add more than one update at a time.  
   
7. Click OK to make the Association.
8. Done!

## File Inventory

File Inventory has two primary purposes:

* Provide data to be used by Detection or Applicability Rules
* Provide executable information for Application Metering for both monitoring and blacklisting (denying execution) applications.

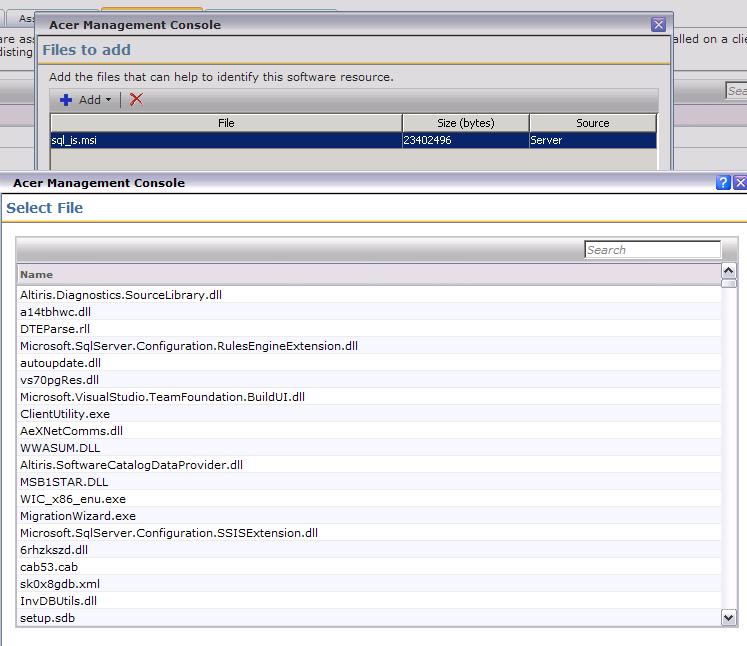
How files are added depends on which processes are in use in the environment. The following methods populate this information:

1. You can add files manually through a local browse on the server (with the ability to network browse).
2. Use data captured through Inventory Solution’s Audit Scan. This must be added manually in the same manner as the Browse functionality works.
3. Use Software captured by the Software Discovery process Inventory Solution executes. The option is found on any main Inventory Task or Policy page, labeled: Software – Windows Add/Remove Programs…
4. For MSIs the data is automatically populated when the Software Resource is deployed through a Managed Software Delivery Policy

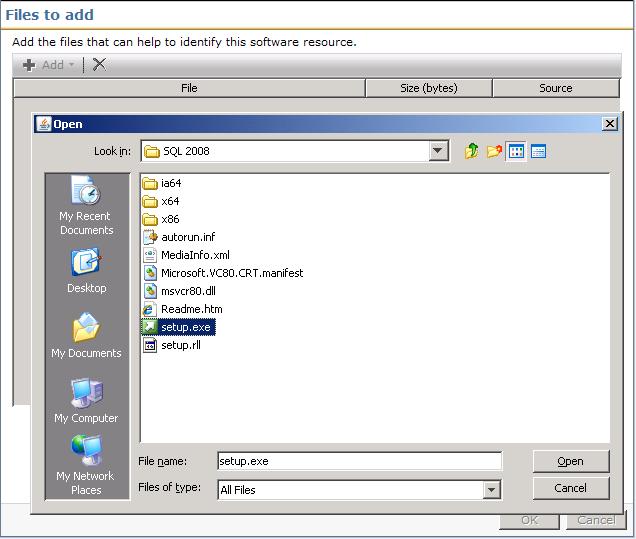


The Browse function requires the JRE (Java Runtime Environment) as discussed concerning using the Software Library as the Package Source. As a reminder the JRE needs to be available on the system running the console, whether remote or local.

To add a Database File, follow this process:

1. Launch the Editor for the Software Resource you wish to add the File Inventory item to.
2. Select the File Inventory tab.
3. Click the Add button.
4. In the resulting Window click Add > Server file.
5. A new window will pop up providing a search/selection window for what Application files are known by the Notification Server. See this screenshot for an example:  
   
6. Use the Search field to narrow your search to find the file needed. If you cannot find the file you need, use the Local browse method to add the file details to the Software Resource.
7. Click OK to add the File. Repeat the steps as needed, and/or if you also want to include Local Files see the next section detailing the process.
8. Done!

To add a File you’ve browsed to, follow this process:

1. Launch the Editor for the Software Resource you wish to add the File Inventory item to.
2. Select the File Inventory tab.
3. Click the Add button.
4. In the resulting Window click Add > Local file.
5. The resulting browse window uses the JRE. You can browse the local Server, or put in another path (UNC or through the Network object). See this screenshot for an example:  
   
6. Once you’ve located the applicable file, select it and click Open.
7. The file will be added to the list. Click OK to add the selected files to the Software Resource.
8. Done! Don’t forget to click Save changes on the Software Resource.

## Software Publishing

This tab controls Software Portal access for this Software Resource. Please see the section on the Software Portal for information on how to use this feature.

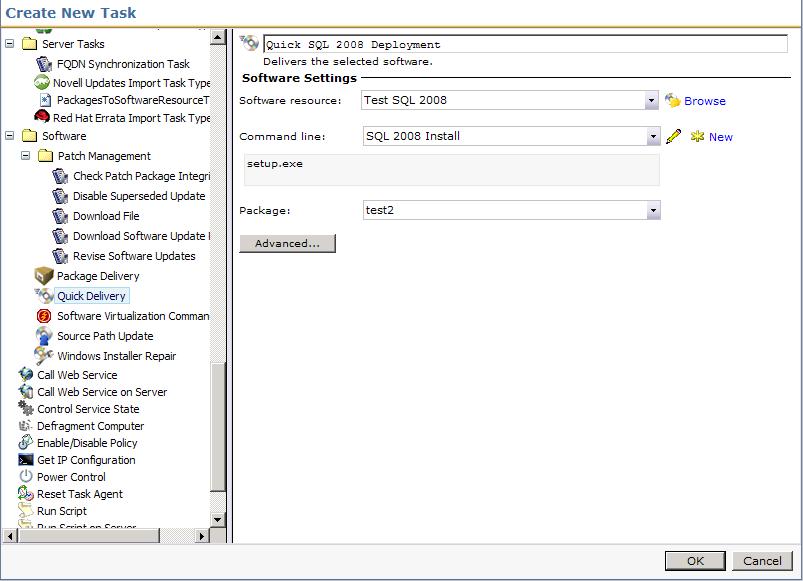
# Quick Delivery

As the name implies, Quick Delivery is primarily used for single Software Deployment events. Other uses, such as running commands or other scripts, can also be used. This is not considered a “Managed” method of deploying Software. Running primarily on Task Server, Quick Delivery Tasks use the Site Server method of quickly deploying a package and command to target clients. The Configuration options are limited on a Quick Delivery, noted in the processes below.

The basic creation of a Quick Delivery Task is covered below. While there are Wizards and contextual links that provide other methods, the base creation is covered here:

1. In the Symantec Management Console, browse under Manage and select Jobs and Tasks, or if you are already in the Activity Center (ECV) click on the Jobs / Tasks section found in the left-hand column.
2. Browse under Jobs and Tasks > System Jobs and Tasks > Software > and select Quick Delivery.
3. Right-click on Quick Delivery and choose New > Task.
4. From the left-hand pane in the resulting window, browse down to the Software section (folder) and select Quick Delivery.
5. In the right-hand pane Name the Quick Delivery Task so it is recognizable when you need to go back to it.
6. In the Software Resource multi-function field type in the name or use the drop down to select the Resource to use.
7. Use the field adjacent to Command line to select the appropriate command line for the Task to run. Be sure to pick the right one if you have different types created in the Resource.

It is advised to review the command-line under the field. It will show you what command will execute, and can help avoid picking the wrong command-line. Running an uninstall when you’re trying to install is quite ineffective. ☺

1. Use the Package field to select the Package to use. Since a Software Resource can have more than one package this step is required before saving a newly created Quick Delivery Task. See this screenshot for an example:  
   
2. Click Advanced. Use the next section to configure the Advanced section.
3. When completed, click OK to save the Task.

## Advanced Configuration

After the Task is set, the Advanced options provide fine tuning of the Task, and can change drastically how the Task runs. This section covers the different options and how they affect the Task. First I’ll details the options, and then provide a walk-through for some of the more popular options.

**Download Options**

By default the Altiris Agent will download files to the following location: *install\_path*\Program Files\Altiris\Altiris Agent\Agents\SoftwareManagement\Software Delivery\. Each package is placed inside a folder matching the GUID of the Package and a subsequent Cache folder. At times an administrator may want to put the package in a non-default location. Change the Destination download location radial to Location on destination computer. The path will be local relative to the target systems.



NOTE: The download will occur directly to the path given. It does not use the GUID folder method used by a Managed Software Delivery Global Option for the download path. Each Quick Delivery needs a unique path or else all files will end up in the same folder, causing overwrite, validation problems.

The Download using: options present a unique situation. In versions 6.x the defaults found here were also found under the Altiris Agent Configuration Policies. This is no longer the case. The settings for Downloading in the Altiris Agent Configurations and the options here are exclusive. While it does add to confusion, it allows you to configure more options than previously. There are two primary use cases for manipulating the options here:

1. Control limited bandwidth between the Package source and target clients, particularly in WAN environments or poor VPN connections.
2. Have the execution occur locally directly from the Package Server instead of downloading it first.



The two checkboxes can be checked, giving the appearance that the two settings are mutually exclusive. Since they are talking about the opposite end of the download or run remote equation, you should only use one at a time.

The option: Delete package from client computer can be used to control how long the source files are available. For most Quick Delivery Tasks an immediate delete fits the use-case model. This may not be the case, so consider how long the source files should stay for the circumstance and set it appropriately.

The nature of Quick Delivery Tasks suggests you’ll only need the files for the Installation. Check the option to delete the package as it is not checked by default. One type of exception is for those applications that install via an MSI and where the Source files may be needed in the future.

**Run Options**

The Run As options provides functions for what user context will be used for the installation.



The default option on a Quick Delivery is logged on user. This default will not work for users who do not have install rights on their local systems. It is advised to change this when configuring a Quick Delivery Task for the first time.

Generally the Altiris Agent credentials are sufficient for an install. Since the package is downloaded, all necessary rights exist on the local system. Some exceptions to this are details here:

* The install accesses network resources or location during the install. This does not include the ability of the application to contact network resources after the installation.
* The source files for the install will be executed from a remote location, such as a UNC or the option to execute from the Package Server is selected.
* The application must be run under the User’s context that will launch the application for use.

For the first two use-cases, Specific user should be used, one that has rights to the resources or network locations to be used during the execution. For the last one the execution must be set to Current logged-on user.



If the users do not have installation rights to their systems, and you have an application that only works when installed under the user’s own context, use a Managed Software Delivery Policy (MSD). The MSD can be configured with current user credentials and it will elevate the install session for that user to allow the install to proceed.

**Task Options**

The two options under this Task Options tab include the ability to have Task Server execute other Tasks while this one is executing, and to define the length of the execution.

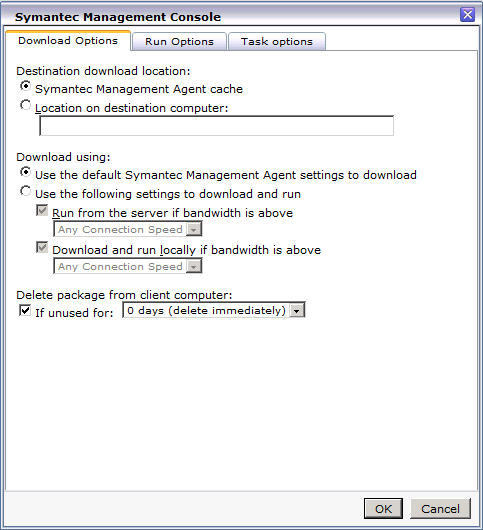
The option, End task after \_\_ minutes, is important to note. The default may not work for a larger install. In 7.1 GA 30 minutes was the default. In 7.1 SP1 5 hours is the default. While these defaults work for short installs, larger packages with longer installs may exceed this amount. This is also compounded in hierarchy environments where the package needs to trickle down to the lower level packages servers. This entire process is included in the timeout, so plan accordingly.



When the Quick Delivery Task executes, it first has to download the Package. The download time is calculated into the Task kill setting. Set the length for enough time for the targets to download the package and execute the command-line successfully.

**Walkthrough**

The following process uses common settings to configure the Advanced options.

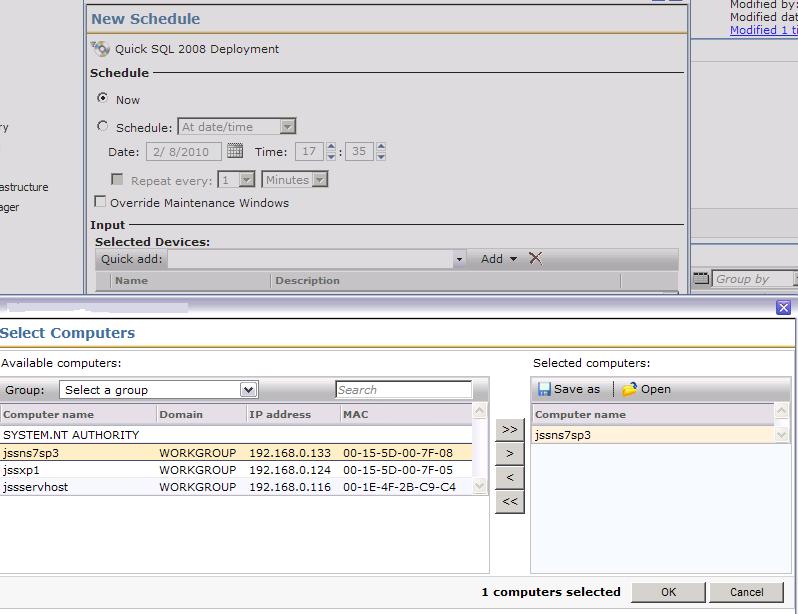
1. Select the Quick Delivery Task to configure, or if you are still in the creation process, click the Advanced button.
2. Under the Download Options tab, under the Delete package from client computer settings, check the box labeled: If unused for:.
3. From the drop down select an appropriate length of time. For this example I left 0 days selected. NOTE: This option doesn’t immediately delete the package, but a maintenance process will remove them soon after the execution. See this screenshot for an example:  
   
4. Click the Run Options tab.
5. Change the option to Altiris Agent credentials.
6. Uncheck the option Allow user interaction. In most cases I’ve worked with administrators have not wanted users to see an install as this often generates a Helpdesk call.
7. Click the Run Options tab.
8. Change the designated minutes to 300 if it is set to 30. This gives clients more time to download and execute the task. This setting will solely depend on environmental and situational elements, including:
   * How soon Package Servers will have this package available
   * The size of the Package and how many files are contained within
   * The length of the Execution on target systems
9. Click OK to save the changes.
10. If newly creating a Task, Click OK to create it. If editing an existing Task, click Save changes to apply the new settings.

## Targeting and Scheduling

The targeting for a Quick Delivery uses the common Task Server scheduling controls. If you are familiar with this, the following walk-through should be familiar. The scheduling also uses the standard Task Server controls.

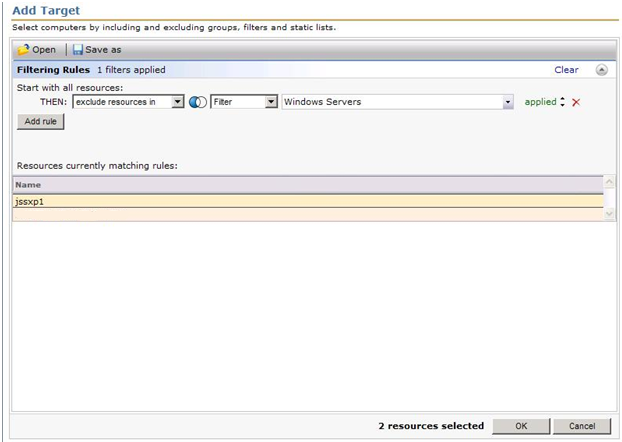
**Manual Selection**

To add targets to the Task and schedule the execution, follow these steps:

1. Browse to the Quick Delivery Task to schedule/execute.
2. Under the setting page, under the Task Status section, click on New Schedule.
3. Choose Now or a specific schedule using the radial button and applicable controls. Due to the nature of a Quick Delivery, for this example I’ll use Now.
4. Under the Input section, click Add v > Computers or Devices. If you have a Target saved of systems to use, you can alternately use Target from the dropdown and link to it. Use this screenshot for reference:  
   
5. In the resulting window select the systems you wish to run the job on. The following methods can assist in finding designated computers:
   1. Use the Group: dropdown to filter results by Organizational objects/views.
   2. Use the Search field to filter the results based on computer name.
   3. Multi-select computers using the Ctrl and Shift keys and use the >> button to transfer them over.
6. Click OK once the computers you need are included.
7. Click Schedule to commit the execution/schedule.
8. The status can be reviewed in the Input section as each execution will be listed. Double-click on the entry to track the status of the Quick Delivery task.

In a large environment, obviously using the above method would be cumbersome and take significant time. Targets are the answer for this. Targets are dynamic collections of computers. The following example shows a basic Target being created.

**Targets**

1. Browse to the Quick Delivery Task to schedule/execute.
2. Under the setting page, under the Task Status section, click on New Schedule.
3. Choose Now or a specific schedule using the radial button and applicable controls. Due to the nature of a Quick Delivery, for this example I’ll use Now.
4. Under the Input section, click Add v > Target.
5. Click the Add Rule button.
6. Leave the selection exclude resources in, and the type to Filter.
7. Type Windows Servers and click the dropdown arrow in the third field.
8. Select Windows Servers from the dropdown (we will be excluding all Windows servers based on our criteria).
9. Click Update results to see what is returned. See this screenshot for an example:  
   
10. Click OK to apply the Target.
11. Done! You can now execute the task or set the schedule for execution.

Alternatively you can create your own Filter, to exclude or not exclude, a list of computers. The Filters can be static or dynamic. In this way it gives you full flexibility when setting up your targets for a Quick Delivery Task.

## Task Server

Task Server is the engine behind Quick Delivery Tasks. These Tasks are no initiated on the Altiris Agent side, but are pushed down from the Target system’s Task Server at the scheduled time. The process is as follows, in high level view. Use the previously covered items to fill in the specifics on this process:

1. An administrator configures a Quick Delivery Task by choosing a Software Resource and a Command Line.
2. A Filter, or basically a collection of computers, is selected to receive the Task.
3. A schedule is set. A popular choice is the “Now” option.
4. The Notification Server passes the Task to the applicable Task Servers.
5. The Task Servers take the Task and pushes it down to the Client Task Agent running within the Altiris Agent at the Target Systems.
6. The Client Task Agent accepts the Task and executes the Task.
7. This Agent first employs the Package download mechanisms within the Altiris Agent to download the package. Normal statuses for the download are passed up as with any internal NS policy or Managed Software Delivery policy.  
   *NOTE: I’ve stated it before but it’s worth mentioning again. The download is considered part of the Task’s execution time.*
8. Once the package is downloaded, the Command-line is executed per the configuration within the Software Resource.
9. The execution completes and provides the applicable exit code back to the Client Task Agent.
10. The Client Task Agent passes the status up to the Task Server.

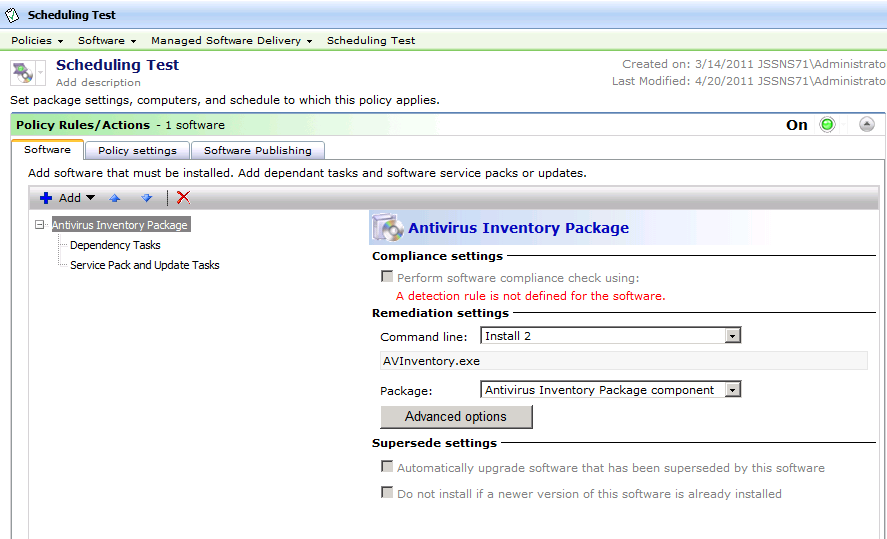
If your Task Server environment is healthy and properly distributed (Site Servers configured to spread the load out) the Executions should begin in real time. Latency is minimized and the completion of the Task should occur quickly when you allow for the length of the execution. If there are few task servers for a large target, the tasks will not all execute at once as the Task Server moves through the list.

# Managed Software Delivery

Managed Software Delivery Policies are the intelligent way to deploy and manage software. These policies are versatile, and enable you to make use of the functionality discussed in configuration terms about the Software Components. Unlike Quick Delivery Tasks, Managed Software Delivery policies are Agent-based. The policies and executions are tracked on the Agent side.

## General Configuration

Setting up a basic Managed Delivery policy is easy. Most of the configuration is done on the front end when you create the Software Resource. The following walkthrough takes you through the basic setup of a Managed Software Delivery (MSD) Policy.

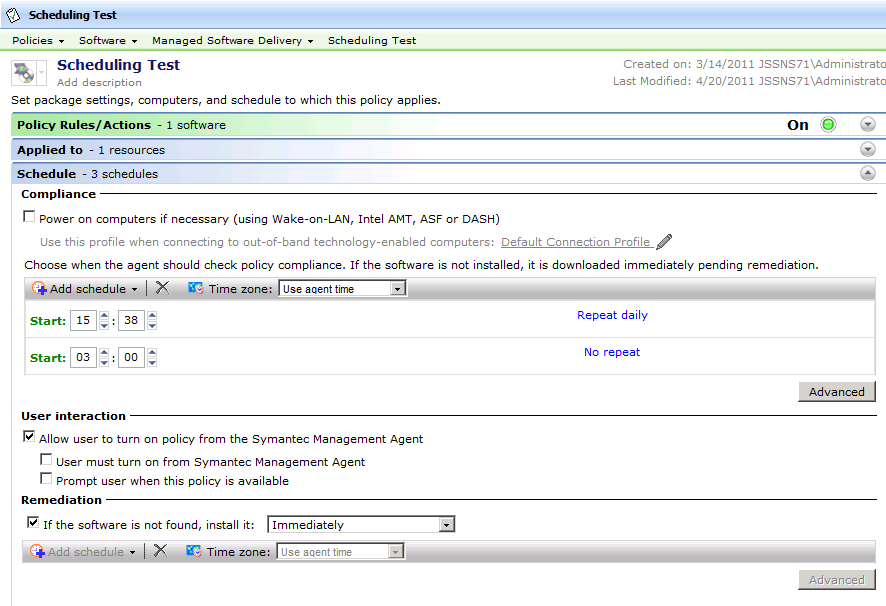
1. In the Symantec Management Console browse under the Manage menu and click Policies.
2. In the upper left-hand pane browse down through Software > and select Managed Software Delivery.
3. Right-click on the Managed Software Delivery folder and choose New > Managed Software Delivery.
4. Name the Policy. As the name is not initially selected it is easy to fail to name it, which may result in confusion as the default name New Managed Software Delivery. Click on the name field to activate it.
5. If desired, add a description by clicking on Add description found below the Name field.
6. Under the section Policy Rules/Actions, under the Software tab, click the + Add button and select a Software resource.
7. In the picker use the Search field to find the applicable or desired Software Resource and click OK. See this screenshot for an example of an added Software Resource:  
   
8. In the right-pane you can select what Package and Command line the Policy will use. Be sure to select the correct command line. If an uninstall command-line is selected inadvertently, this policy will not behave as expected.

Note that the above example shows “A detection rule is not defined for the software.” This does not mean the Policy will not run any Detection Rule, in fact it will run one. The default detection rule uses the Softwarecache.xml where the Software Discovery data is held. When software is installed, an entry is put into this cache to be reported the next time Inventory runs. If no detection rule is defined it will look for the software using this information in the cache file

1. Scroll down and open the Applied to section by clicking on the expand/shrink arrow button.
2. Click Apply to v dropdown and choose Computers.
3. The Filter interface should be familiar if you walked through the Quick Delivery section concerning creating a Target. Add machines using the same method described on page 32.
4. You should now see a row for the Computer List you just created, including a Count column that shows you how many machines are targeted for the Policy.
5. Scroll down and open the Schedule section by clicking the arrow button.
6. Starting with 7.0 MR4 and 7.1, there are two schedules that can be applied to a Managed Software Delivery Policy. The first schedule is the Compliance check, the second the remediation.

A common request involves having target clients download the package in advance, or even well in advance, of the execution. The download is tied directly to the compliancy check (running of the detection rule). Set the Compliance check well in advance to allow systems time to download the package. This is useful for large packages or servicing WAN sites.

1. Under the Compliance section click the Add schedule v Dropdown and select an appropriate Schedule option. The different settings here will be covered later.
2. Under the Remediation section, either leave the default set to immediately or set a schedule later than the compliance schedule. Here is a screenshot showing the two schedule options:



1. After adding the schedules, browse back to the top and be sure to turn the Policy on by clicking on the On/Off toggle dropdown and selecting On.
2. Click Save changes to save and commit the policy.

How do I get the ASAP option to run a Compliance check immediately? This is a common question, especially from those customers who were familiar with the ASAP option found in version 6.x of Software Delivery. In the above screenshot, you’ll notice there is a reoccurring schedule and a run once schedule with no repeat. ASAP is achieved by having a Run Once > No repeat schedule where the Start date is in the past. If you want ASAP to work right out of the gate, click Advanced and change the start date to the day prior.

## Software Resource Settings

After you’ve added a Software Resource to a Managed Software Delivery Policy, you have a myriad of options in how the policy interacts with the Software Resource during detection and deployment. This section covers all the settings involved. When selecting a Software Resource in the Rules/Actions list, the right-hand pane presents the settings surrounding the Resource.

**Compliance settings**

By default the Compliance Settings will automatically choose what Detection Rule was selected during the creation of the Software Resource. You can toggle the detection on or off by checking or removing the check to the box labeled: Perform software compliance check using:. You can also manipulate the detection rule by clicking on the hyper link named after the Detection Rule name. You will see the View Rule dialog, which allows you to edit the Expressions and rules defined.

**WARNING:** Changing a Detection Rule in this window will result in the Software Resource being updated with the changes. Only make changes that you wish to make to the Software Resource directly. Also note the default detection rule explained under the Managed Delivery Policy walkthrough preceding this section.

**Remediation settings**

By default the Install Command-line is selected in the Command-line dropdown. If the Resource has more than one command-line, use the selection dropdown to choose the appropriate one. Below the dropdown the Command-line is listed for reference so you can ensure you’ve selected the right one. The Package dropdown allows you to select a different Package if the Software Resource has more than one Package defined. For most use cases typically only one package is required or specified.

**Supersede settings**

The two checkboxes listed here allow greater control of how updates and newer versions are handled in regards to the Policy. The logic to allow automatic updates or to avoid applying a lower version than what’s already installed avoids any issues stemming from the wrong versions being applied.

* Automatically upgrade software that has been superseded by this software – Any software detected that this software supersedes will initiate the update.
* Do not install if a newer version of this software is already installed – This uses any Software Resources marked as superseding the selected Resource.

Note that the options will be grayed out if no Supersede associations are found for the selected Software Resource.

## Advanced options

At the bottom of the Remediation settings section Advanced options are offered. While these are similar to those Advanced Options covered earlier for Quick Delivery Tasks, there are key differences. This section covers the different options and how they affect the Policy. First I’ll details the options, and then provide a walk-through for some of the more popular options.

**Download Options**

By default the Altiris Agent will download files to the following location: *install\_path*\Program Files\Altiris\Altiris Agent\Agents\SoftwareManagement\Software Delivery\. Each package is placed inside a folder matching the GUID of the Package and a subsequent Cache folder. At times an administrator may want to put the package in a non-default location. Change the Destination download location radial to Location on destination computer. The path will be local relative to the target systems.

The Download using: options present a unique situation. In versions 6.x the defaults found here were also found under the Altiris Agent Configuration Policies. This is no longer the case. The settings for Downloading in the Altiris Agent Configurations and the options here are exclusive. While it does add to confusion, it allows you to configure more options than previously. There are two primary use cases for manipulating the options here:

1. Control limited bandwidth between the Package source and target clients, particularly in WAN environments or poor VPN connections.
2. Have the execution occur locally directly from the Package Server instead of downloading it first.



The two checkboxes can be checked, giving the appearance that the two settings can be used together. Since they are mutually exclusive concerning downloading or running remotely, you should only use one at a time.

The option: Delete package from client computer can be used to control how long the source files are available. For most Managed Software Delivery Policies the default is 7 days. You may need to increase this, or deselect the option altogether when considering how long the source files should stay for the circumstance.

The nature of installations may suggest you’ll only need the files for the Installation. Check the option to delete the package to clean these up, especially if hard drive space is at a premium. One type of exception is for those applications that install via an MSI and where the Source files may be needed in the future for a Repair, Component addition, etc.

**Run**

The Run As options provides options for what user context will be used for the installation.



The default option on a Managed Software Delivery Policy was changed to be the Symantec Management Agent credentials. The policy will use the Agent credentials to elevate the execution from a logged on user that will allow this configuration to work even for users that do not have install rights. Note that we execute in a separate session so any rights elevation will not be usable by the logged on user. We have seen occasional rights issue despite the elevation, so unless there is a reason to target the logged on user, it is advised to only use this when the installation requires the user’s own session.

Generally the Symantec Management Agent credentials are sufficient for an install. Since the package is downloaded, all necessary rights exist on the local system. Some exceptions to this are detailed here:

* The install accesses network resources or location during the install. This does not include the ability of the application to contact network resources after the installation.
* The source files for the install will be executed from a remote location, such as a UNC or the option to execute from the Package Server is selected.
* The application must be run under the User’s context that will launch the application for use.

For the first two use-cases, Specific user should be used, one that has rights to the resources or network locations to be used during the execution. For the last one the execution must be set to Current logged-on user.



For applications that only work when installed under the user’s own context, use the configuration Current user credentials and it will elevate the install session for that user to allow the install to proceed. Occasionally we’ve seen issues with this, so test before deployment.

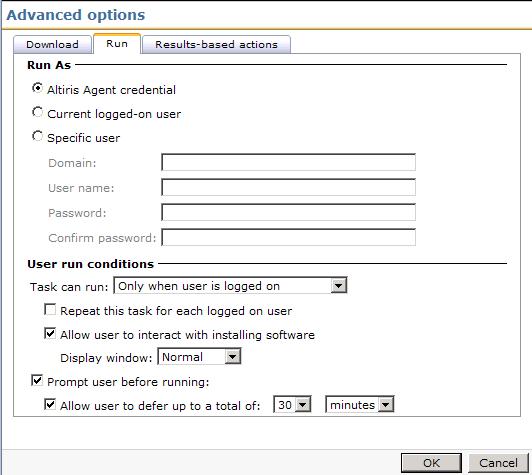
User run conditions allow greater control over how and when the policy executes. The following options are available:

1. Task can run: - This option allows an administrator to control in what state the system is in for execution regarding user log on. Scenarios include:
   1. Only when user is logged on – This is the only option if you selected Current logged-on user under the Run As section, though if you selected another Run As you can still specify this to ensure someone is logged on when the execution occurs. This is important if your installation requires some sort of user interaction when it executes.
   2. Whether or not the user is logged on – This option is used to allow the install no matter what user state the system is running under. This one will allow the policy to execute the soonest.
   3. Only when no user is logged on – This allows an administrator to restrict resource intensive installs to run when no user is actively using the system.



Option C can cause difficulties depending on how users manage the power state of their computers. For example if they log on quick enough after booting up the computer the execution may miss the small window of opportunity. Also if users don’t typically log out, but shut down their systems at the end of the day, this can also delay the remediation.

1. Repeat this task for each logged on user – if the installation you are running requires to run under the user’s profile to apply correctly, this option is very useful to ensure it runs for every user who uses the system.
2. Allow user to interact with installation software – To put it simply, this pipes the execution to the User’s desktop, running in interactive mode. You can also select how the execution shows, from the following list:
   * Normal
   * Hidden
   * Maximized
   * Minimized
3. Prompt user before running – This setting allows the user to defer the execution. This helps mitigate any work the user is conducting when the installation begins to happen. Users can save their work before the installation to avoid any potential data loss.

This screenshot shows a typical configuration of this tab:  


**Results-based actions**

Outside of the actual execution of the Software Resource, these settings allow how the Symantec Management Agent and Plug-ins interact with the task.

* Upon success: This allows the Task to execute a Log off or reboot of the system after a successful completion.

 It is highly recommended to allow the Symantec Management Agent to initiate a reboot. This allows the Agent to properly handle all Agent processes so nothing is lost on the reboot. For example the progress of the Managed Software Delivery may be lost if the installer executes the reboot (stored in the file AeXSWDPolicy.xml). In the command-line suppress the reboot and select the Upon success option to initiated the reboot.

* + Allow user to defer action up to: Use this option to allow the user to save and make any other preparations for the reboot. This is highly recommended when configuring a reboot to minimize user disruption.
  + Force running applications to close – If an open application prevents the shutdown, this option will forcibly end the application to conduct the reboot. Use with caution.
* Terminate after: This option is a fail-safe to prevent hung or sluggish installs from locking up the Altiris Agent Policy queue. Generally putting in a value that is safely beyond the expected execution time, but not too long to paralyze the Agent, should be used.
* Upon failure: The following selections are available:
  + Abort – This will stop the MSD from continuing. If you’ve configured more than one resource or task in the MSD, any subsequent items will not be run.
  + Continue – this will instruct the MSD to continue on to subsequent items in the MSD.
  + Restart – This will retry the remediation execution. Max retries should be set to an appropriate number.

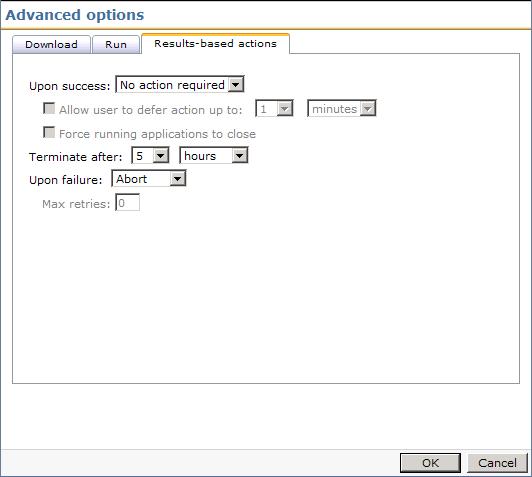
**Walkthrough**

The following process uses common settings to configure the Advanced options of a Managed Software Delivery Policy.

1. Select the Managed Software Delivery Policy to configure, or if you are still in the creation process, click the Advanced options button under the selected Software Resource.
2. Under the Download Options tab, under the Delete package from client computer settings, check the box labeled: If unused for:.
3. From the drop down select an appropriate length of time. For this example I set 7 days.



This setting only comes into effect when the Managed Software Delivery Policy no longer applies to the targeted system. This means under normal circumstances the package will not be deleted. If you use a dynamic filter, or you disable the policy, then the setting will apply to the Package.

1. Click the Run tab.
2. Change the option to Symantec Management Agent credentials.
3. Leave the option checked: Allow user interaction. Since you can suppress visually the execution via the command line or by the setting of how it starts (choosing hidden), this option can remain checked. In few occasions we’ve found user-based installs to work using this option checked.
4. Check the option Prompt user before running.
5. Check the option Allow user to defer up to a total of.
6. Input an appropriate length of time, such as 30 minutes or 1 hour.
7. Click the Results-based actions tab.
8. Change the Terminate after: setting to 5 hours (in 7.1 this is the default).
9. Change the option Upon Failure to Abort. See this screenshot for an example:  
   
10. Click OK to save the changes.
11. If newly creating a Managed Policy, Click OK to create it. If editing an existing Managed Policy, click Save changes to apply the new settings.

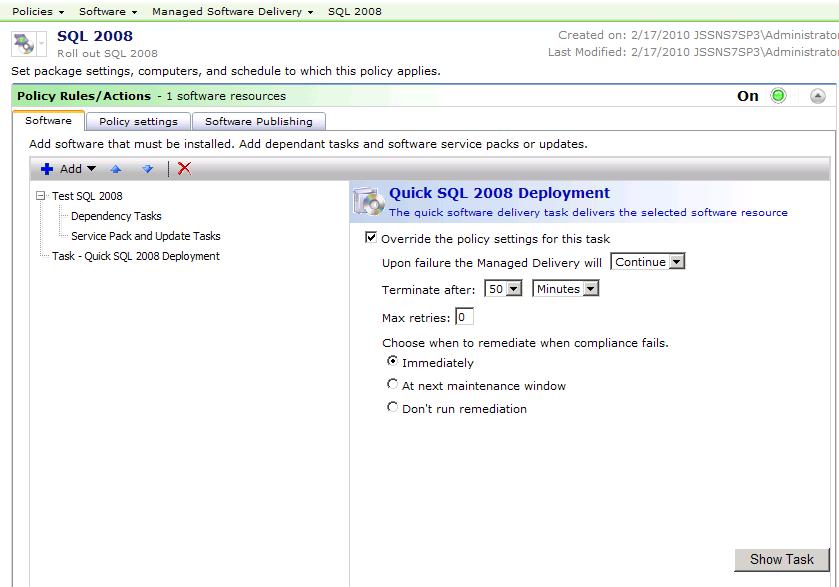
## Task Settings

You can also add any Task Server Task to a Managed Delivery. The following example illustrates how to accomplish this:

1. In the Managed Software Delivery Policy, under the Policy Rules/Actions, Software tab, Click the + ADD button and select Task



When adding a Task, do not have a Software Resource selected or the Task will replace it. If the Resource is selected you can deselect it by moving to another section such as Applied to.

1. Select the desired task from the list. Use the Search field to narrow the results if necessary.
2. The Task will be added to the list. For sequencing information, see the subsequent section. See this screenshot for an example:  
   
3. You have the ability to override the policy settings for the task. The following walkthrough provides an example with use-case information explained.
   1. Check the box Override the policy settings for this task.
   2. Choose Continue from the dropdown next to the label Upon failure the Managed Delivery will.
   3. Change the Terminate after setting to 50 Minutes.
   4. Leave the other settings as is.
4. If you need to adjust the Task itself, you can click the Show Task button in the lower right.
5. Be sure to Save changes when you’ve completed the Task configuration.

## Sequencing

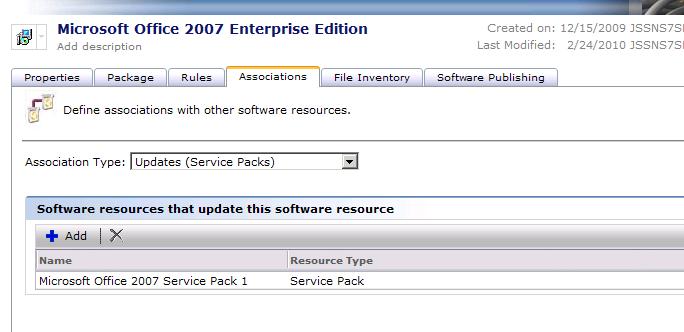
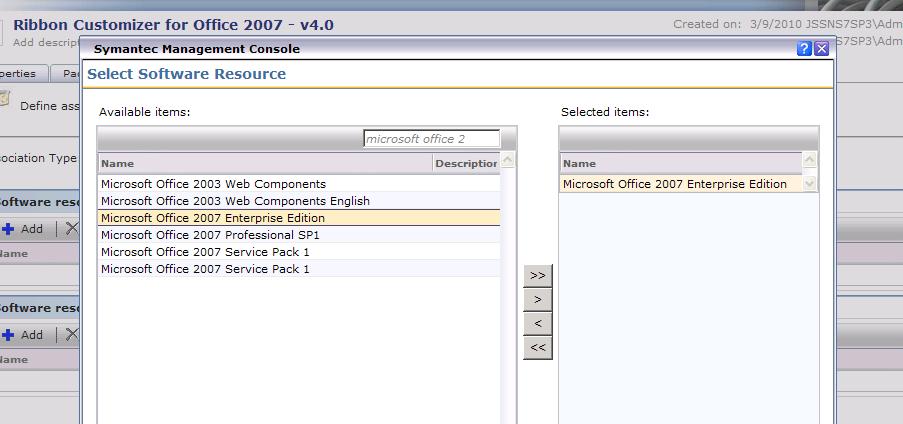
Managed Software Delivery Policies allow a sequence of Software Resources and/or Tasks. This allows grouping of like executions or a string of executions to create a more comprehensive job. The following 2 primary use cases explain benefits of using a Managed Software Delivery in this manner:

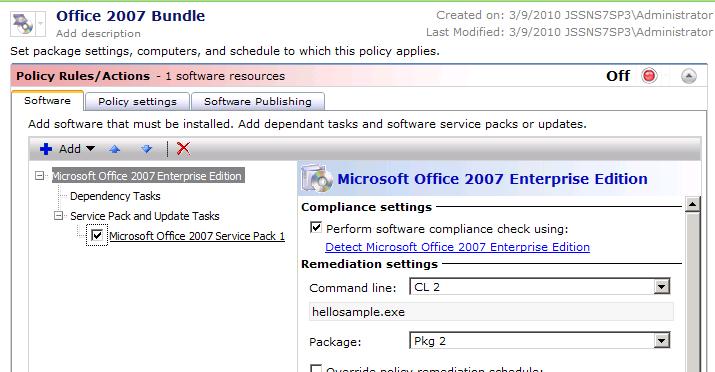
1. Complex Software Management – For software that contain a base install and multiple updates, service packs, plug-ins, and/or add-ons creating one MSD to manage the rollout of all the components will simplify the rollout.
2. Core Software Management – Most companies have a core set of software that all Users should have available. This usually includes Microsoft Office, Cisco or other VPN client, Anti-virus Software (Symantec Endpoint Protection being one good example), branded version of Internet Explorer, etc.

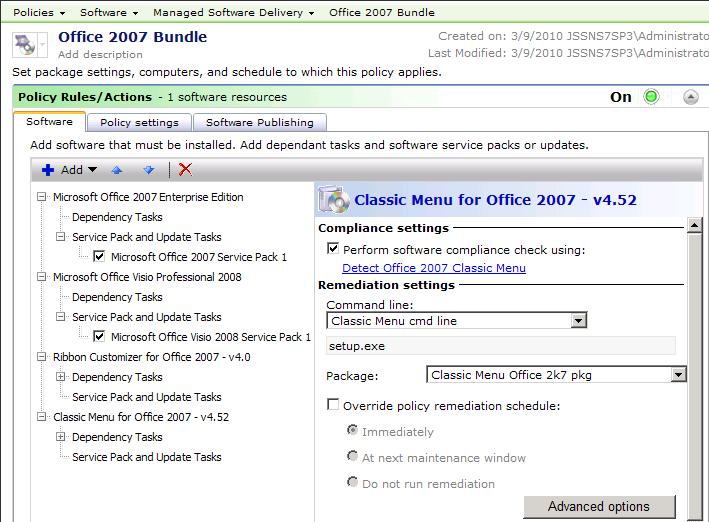
 Be selective when you add multiple Resources and Task to a Policy. The more action points in a job, the greater chance to run into issues. The greater the complexity will also factor in, creating more potential points of failure.

To illustrate how this can be used, I’ll provide a walk through for each of the use cases listed above. This first example uses Microsoft Office for an example. Note that the steps are in the order the options appear (to avoid excessive scrolling back and forth).

**Complex Software Management**

1. Create Software Resources for the following components:
   1. Microsoft Office 2007 Enterprise Edition
   2. Microsoft Office Visio Professional 2008
   3. Microsoft Office 2007 service pack 1
   4. Microsoft Visual Studio 2008 Service Pack 1
   5. Ribbon Customizer for Office 2007 - v4.0
   6. Classic Menu for Office 2007 - v4.52
2. Create detection rules for each Resource. If the install has a MSI Product Code association, use this for the software detection. Other examples are registry keys. For example, for Office 2007 you can use the key: HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Office\12.0\Common\ProductVersion, with a value name of Lastproduct. You can specify a specific value, or simply use the existence of that key.
3. Create the following Service Pack Associations within the associated Software Resources:
   1. Browse to the Software Resource for Microsoft Office 2007 Enterprise Edition.
   2. Click on the Associations tab.
   3. From the Association Type dropdown, select Updates (Service Packs).
   4. Under the caption, Software resources that update this software resource, click the + Add button.
   5. From the list, locate the Software Resource for Microsoft Office 2007 service pack 1.
   6. Select and click the > button to move the Resource over to the right-hand pane.
   7. Click OK to save the association.
   8. You should see the association now represented in the list, as shown in this screenshot:  
      
   9. Click Save changes on the main Software Resource page to complete the process.
   10. Now go back through steps a through I except use Visio as the Parent Resource, and Visio SP1 in the association of the Visio Resource.
4. Create the following dependent associations:
   1. Browse to the Software Resource for Ribbon Customizer for Office 2007 4.0.
   2. Click on the Associations tab.
   3. From the Association Type dropdown, select Depends On.
   4. Under the field labeled Software Resources that this Software Resource depends on, click the Add + button.
   5. All Software Resources are shown on the resulting page, so use the search filter to find Microsoft Office 2007 Enterprise Edition.
   6. Select Office and use the > button to move it over to the selected right pane, as shown in this screenshot:  
      
   7. Click OK to apply the new association.
   8. Click Save changes on the main Software Resource page to complete the process.
   9. Now go back through the steps, but choose the Software Resource Classic Menu for Office 2007 - v4.52 to add the ‘depends on’ association to.
5. Create a new Managed Software Delivery Policy (MSD).
6. In the Symantec Management Console, browse under Manage > Policies.
7. In the left-hand tree browse under Policies > Software > and select Managed Software Delivery.
8. In the resulting right-hand pane click the Add + button.
9. In the Name field, provide a name, such as: Office 2007 Bundle.
10. Turn the Policy On by clicking the red/green switch and clicking On.
11. Click the Add + button and choose Software resource.
12. Use the Search filter, if necessary, to locate and select Microsoft Office 2007 Enterprise Edition.
13. Click OK to add the resource to the MSD.
    1. Note that by default a command line and package will be selected. A Detection rule will also be selected, so ensure all settings are as you intend them. Normally this is not an issue unless you have multiples of any item.
    2. Check to ensure the Service Pack shows up under the Software Resource, as shown in this screenshot:

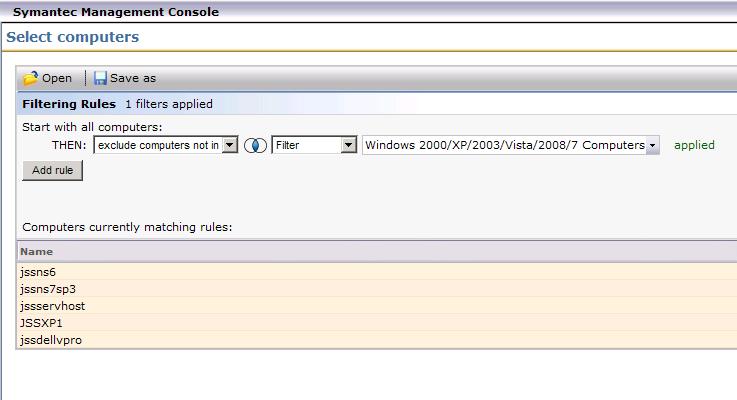


1. Click the Add + button and choose Software resource.
2. Use the Search filter, if necessary, to locate and select Microsoft Office Visio Professional 2008.
3. Click OK to add the resource to the MSD.
4. Click the Add + button and choose Software resource.
5. Use the Search filter, if necessary, to locate and select Ribbon Customizer for Office 2007 - v4.0.
6. Click OK to add the resource to the MSD.
7. Click the Add + button and choose Software resource.
8. Use the Search filter, if necessary, to locate and select Classic Menu for Office 2007 - v4.52.
9. Click OK to add the resource to the MSD.
10. Ensure the Software Resources are in the right order. Select and move entries using the up and down arrows. This screenshot shows the finished list:  
    
11. For the Advanced options for each Software Resource, see the previous section labeled Advanced options.



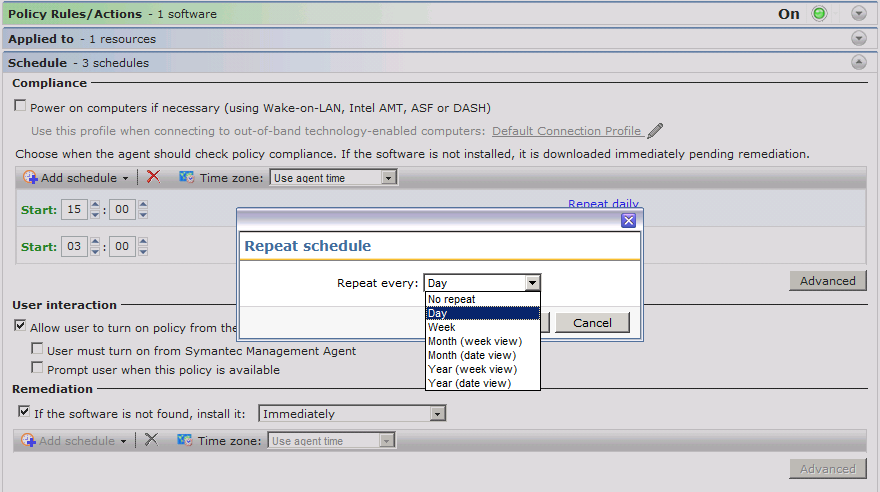
Ensure you go through the Advanced options for all resources added to a MSD. For example if an Office Add/on is only applicable per user, you may need to change the Execution environment to run as the Logged-on User in order to have a successful deployment.

1. Expand the Applied to section.
2. Add a filter to the Policy. For this example we’ll apply it to all Windows Computers.
   1. Click the Applied to button, and choose Computers.
   2. Click the Add Rule button.
   3. Change the 1st option to Exclude all computers not in.  
      *NOTE: The double-negative usage may be confusing. To put it into perspective, note that the Filter starts including all computers, thus the wording, while not intuitive, is accurately labeled in the context.*
   4. Leave the 2nd option on Filter.
   5. In the right field type Windows 2000, and click the dropdown arrow.
   6. Choose the filter Windows 2000/XP/2003/Vista/2008/7 Computers.
   7. Click the Update results button. You should see a list of applicable computers, as shown in example on this screenshot:



* 1. Click OK to apply the filter.

1. Expand the Schedule section.
2. Add a schedule to the Compliance check for the MSD. The schedule in this example will be a reoccurring one that runs daily. This allows computer to check for compliance once a day.
   1. Click Add Schedule and select Scheduled Time.
   2. Input a time of day best suited to run the install. I this example I chose 18:00 (6:00pm).
   3. Click the hyperlink default label: No repeat.
   4. From the dropdown in the resulting prompt, choose daily, as shown in this screenshot:



* 1. Click OK to add the repeat setting.

1. Check the Remediation schedule to ensure it is set as desired. For example immediately is default, but if you wish to wait until off production hours you can set a time after day hours.
2. Click Save changes at the bottom to complete the process!

### Scheduling

There is a known situation that has been seen on occasion. Unfortunately we have not been able to reproduce the issue so that development can diagnose the issue, and it is rare enough that at this point it is only being documented as a possible issue.

* Set the Start window to something else besides 00:00, and 24:00. These two numbers are essentially the same number, and under rare circumstances the scheduled at the Agent can error when trying to evaluate the schedule, causing CPU issues on the target PC.

**Core Software Management**

This series will be a smaller subset of what was introduced under the Complex Software Management section previously covered. In this example I’m introducing a list of common software to deploy.

1. Create Software Resources for the following components:
   1. Symantec Endpoint Protection
   2. Microsoft Office 2007 Professional
   3. Company branded Internet Explorer
   4. Cisco VPN Client
   5. Adobe Acrobat Reader
   6. Adobe Flash Player
2. Create detection rules for each Resource.
3. Create a new Managed Software Delivery Policy (MSD).
4. In the Symantec Management Console, browse under Manage > Policies.
5. In the left-hand tree browse under Policies > Software > and select Managed Software Delivery.
6. In the resulting right-hand pane click the Add + button.
7. In the Name field, provide a name, such as: Common Software Suite.
8. Turn the Policy On by clicking the red/green switch and clicking On.
9. Go through the following process for each of the Software Resources to add to the Policy.
   1. Click the Add + button and choose Software resource.
   2. Use the Search filter, if necessary, to locate and select Microsoft Office 2007 Enterprise Edition.
   3. Click OK to add the resource to the MSD.
   4. Note that by default a command line and package will be selected. A Detection rule will also be selected, so ensure all settings are as you intend them. Normally this is not an issue unless you have multiples of any item.
   5. Click the Add + button and choose Software resource.
10. Ensure the Software Resources are in the right order. Select and move entries using the up and down arrows. In this example it is imperative to put Antivirus first!
11. For the Advanced options for each Software Resource, see the previous section labeled Advanced options.

If one of the applications, such as Symantec Endpoint Protection, requires a reboot, suppress the reboot in the command-line and set a reboot as an after successful completion action within the Advanced options.

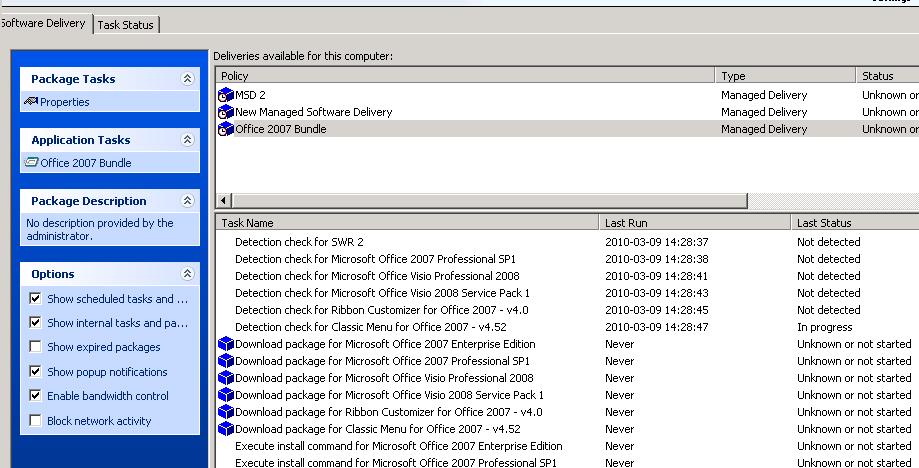
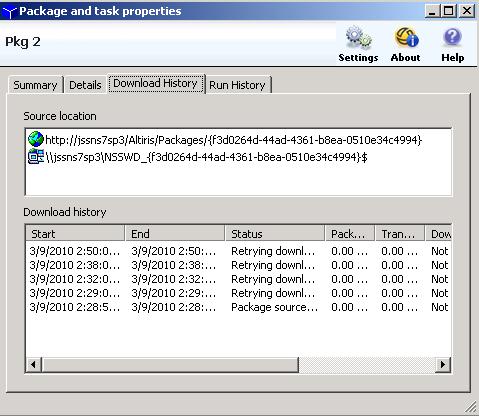
.

1. Add a filter to the Policy
2. Expand the Schedule section.
3. Add a schedule to the MSD.
4. Click Save changes at the bottom to complete the process!

As always, fully test your Managed Software Delivery Policy before attempting to roll it out on any medium or large scale. Tests will reveal issues and allow you to adjust the configuration to address them before full rollout.

**Testing**

Tracking the execution of an MSD on the client helps the testing and troubleshooting process. To access the UI and view the execution, follow these steps:

1. On the client computer double-click on the Symantec Management Agent icon. If the icon is not visible (hidden per policy), browse under C:\Program Files\Altiris\Altiris Agent\ and execute AeXAgentActivate.exe.
2. Select the Software Delivery tab.
3. From the list of available software, highlight the MSD to view (Office 2007 Bundle in this example).
4. The lower pane will show the progress of the MSD execution, as shown in this screenshot:  
   
5. To see the details of a particular download or execution, double-click on a line in the bottom pane that has a blue package icon next to it. You can review the details of the execution, including the download and run histories.
6. The following screenshot shows an issue with a package download:  
   

## Rules

This section covers how Rules are executed within a Managed Software Delivery Policy.

* Detection Rules – Before anything else runs, the detection rules for an MSD runs. If an application is detected, the associated Software Resource will NOT be executed. The status will show as Detected.
  + Consider this rule one that ensures you aren’t rolling the same software to a system that already has it.
* Applicability Rules – For those resources that were not detected, if an Applicability Rule exists it will be executed against the system. Unlike Detection Rules, only if the Applicability Rule is detected will the MSD execute!
  + A common Applicability Rule is Windows Version. The update or Software may not support Windows 7, for example, so the rule will only succeed on Windows systems that are not 7.
  + Another type of Applicability Rule is an update to software that has to exist on the target system. This helps stop those executions that will fail the prerequisite check from throwing an error. Better if it doesn’t run when it doesn’t need to.

Rules add intelligence to the Software Management Process. In version 6.1 of Software Delivery, all intelligence had to be handled at the server end. Inventory provided what we knew was already on the system, and the NS provided the SQL logic in a Collection to automatically roll out Software to those systems that needed it. This was far from real-time and sometimes resulted in machines executing Software it didn’t need or already had. It didn’t take much to be out of sync between the Server and the client as Inventory had to update so the NS would know what had changed.

Rules execute real time, so they will know the moment they run if a system is compliant or not. If not compliant, it runs remediation to resolve the compliance issue.

## Manual Execution

By default Managed Software Delivery Policies can be executed locally at the Agent User interface. This helps when troubleshooting an MSD execution or if you simply need to run a compliance check ASAP. The following process details how to execute the MSD manually:

1. On the client computer double-click on the Symantec Management Agent icon. If the icon is not visible (hidden per policy), browse under C:\Program Files\Altiris\Altiris Agent\ and execute AeXAgentActivate.exe.
2. Select the Software Delivery tab.
3. From the list of available software, highlight the MSD to view (Office 2007 Bundle in this example).
4. Look under the Application Tasks heading. The link here will execute the MSD immediately.

When executing an MSD in this manner, it does not mean the applicable Software Resources will automatically install. The same Rules are applied as in a normally scheduled delivery. Detection and Applicability checks will occur and influence what items run or are skipped.

## Software Publishing

Managed Software Delivery policies can be requested from the Software Portal. This allows users to add an application or suite of applications for automatic deployment and management. This will be covered in detail under the Software Portal section of this document.

# The Activity Center

The name has changed, possibly to “Enhanced Console Views”, but due to the ease of use I will call the new Silverlight interface available in 7.1, the Activity Center. This interface provides a fast, easy to use console that is reminiscent of Outlook®. It allows a user to quickly jump between common areas in the console, such as Policies, Computer lists, Software lists, Tasks and Jobs, etc. Due to the Silverlight technology, these sections are cached locally and take virtually no time to switch between.

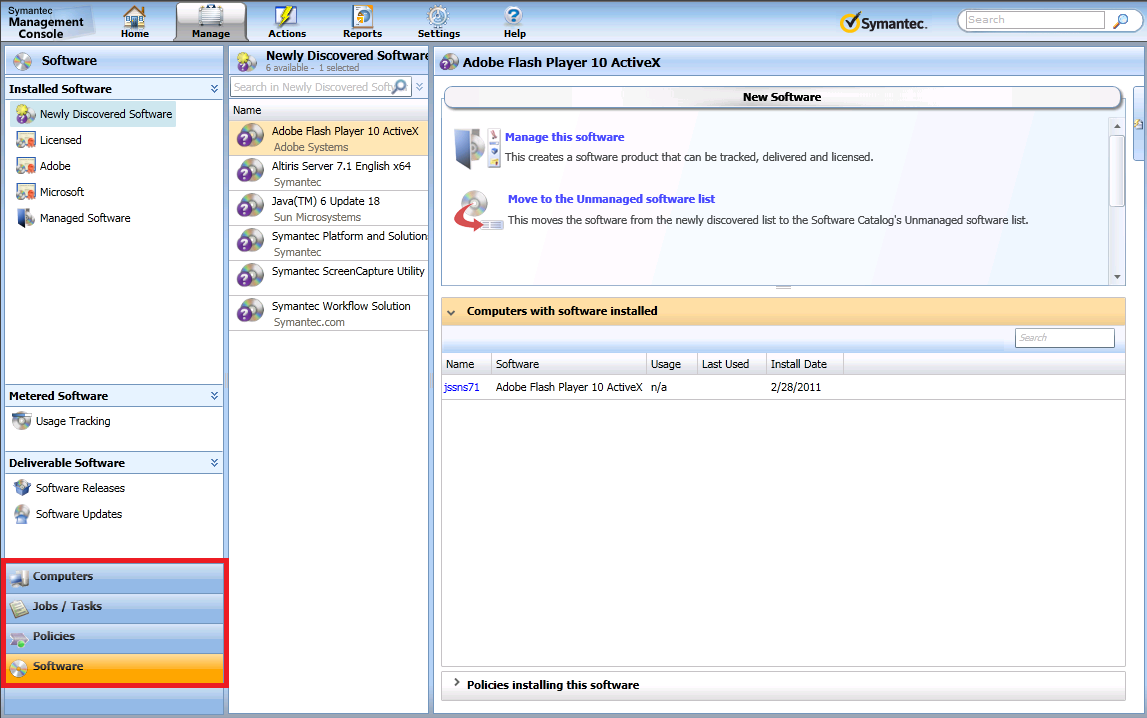
I used to keep my most used tabs in the old 7.0 console open and running in separate pages in order to speed my management of NS. Save for Settings, this is no longer needed as the most common functions are all encapsulated within the Activity Center interface. This console can be accessed via the following methods:

* Manage > Computers
* Manage > Software
* Manage > Jobs and Tasks
* Manage > Policies

There is one other method, but it will also open the Software Catalog interface within the Activity Center:

* Manage > Software Catalog

The following screenshot shows the layout, with the section Software selected:



Each of the sections highlighted above are also targeted using the Manage tab, but once you have the Activity Center loaded it is quicker to switch using the large buttons, and virtually instantaneous.

Each of the sections has its own features and options in the 3 main panes within the activity center. As this guide covers Software Management, I will focus on the Software section as shown in the above screenshot.

## Software Categories

In the left-most pane software is split into different categories. This allows ease of management and quick filtering. When a category is chosen on the left, the middle pane will list the applicable software. You can further filter the results using the search field at the top. Beyond that you see additional information or actions in the right pane, depending on what category on the left is chosen.

The following example shows a flow using this interface to interact with software that is configured to be rolled out via Software Management.

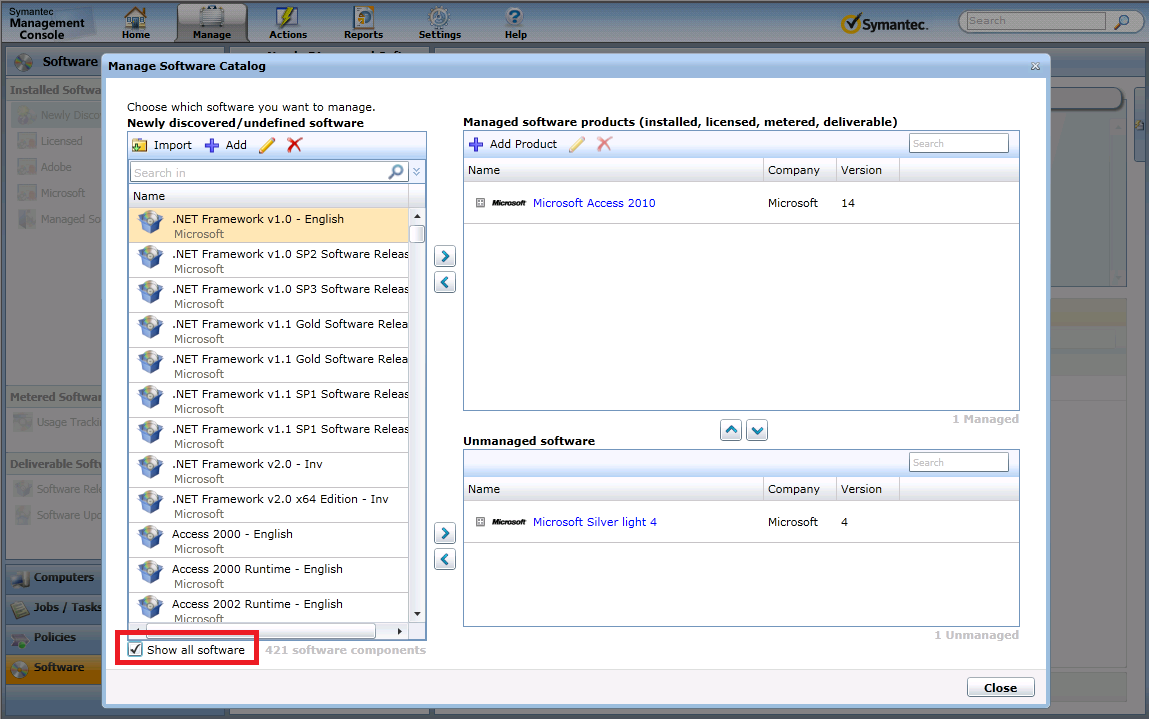
1. In the Symantec Management Console go to Manage > Software, or, if already in the Activity Center, click on the Software link in the lower left.
2. In the left pane select Software Releases. This contains all deliverable software resources that are not of the types Update or Service Pack.
3. In the middle pane type “antivirus”. This should bring up the Antivirus Inventory Package.
4. There are several actions or details you can execute or obtain from the right pane.
   1. Manage this software – This allows you to setup a Product or assign the software to an existing Product. Please see the subsequent section Managing Software Products for more details.
   2. Computers with software installed – This lists all computers that have reported this software either through the Software Discovery process (Windows Add Remove Programs options shown in any Inventory Policy), or a Targeted Software Inventory Policy.
   3. Policies installing this software – This shows all Policies that have this software assigned by name and by target’s name.

## Where is my software?

Using “Newly Discovered Software”, “Managed Software”, and the options under Deliverable Software, you will see a subset of all Software Resources in the environment. So how do you see all Software Resource, regardless of state or type?

**Software Catalog**

One area to view many of your Software Components is the Software Catalog. This can be accessed by going to Manage > Software Catalog. This opens an inset pane over the main Activity Center interface, as shown in this screenshot:



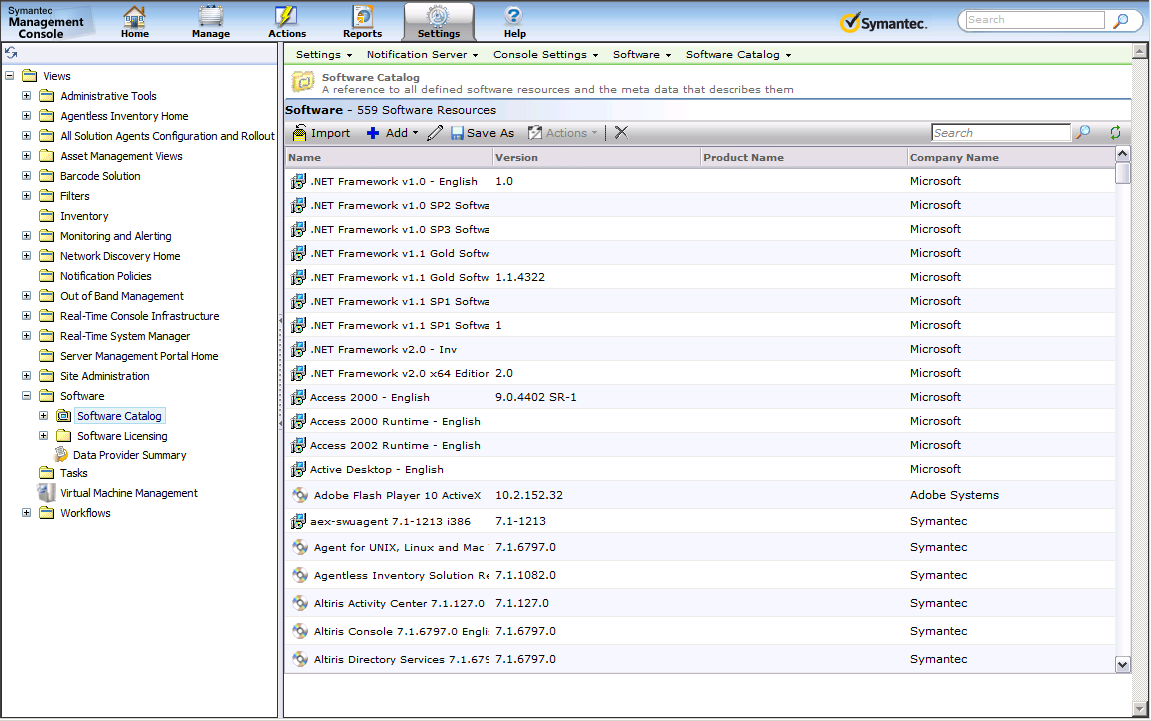
Note that the option “Show all software” is checked. Normally this option is unchecked and you will only get recently discovered resources in this list. Also note that the upper right pane is labeled “Managed software products”. This changed in the SP1 release of 7.1 as this used to show a list of Managed Software, not Products. But note that not all resources are still shown. The Activity Center’s Software Catalog has criteria for showing software. That criteria is important!



Only Software that has been discovered in the environment via Inventory will show in the undefined software list! Even so I have seen even discovered software not show up in this list.

So what is to be done? Where can I see a comprehensive list of everything in the database, whether it be managed, deliverable, discovered, or imported via a data provider? Fortunately the old views are available in this scenario if software you are looking for does not appear in any of the Activity Center’s enhanced views.

1. In the Symantec Management Console go to Settings > Console > and select Views.
2. In the left-hand pane expand Software and select Software Catalog.
3. In the interface use the search field to find the resource you are interested in, as shown in this screenshot:

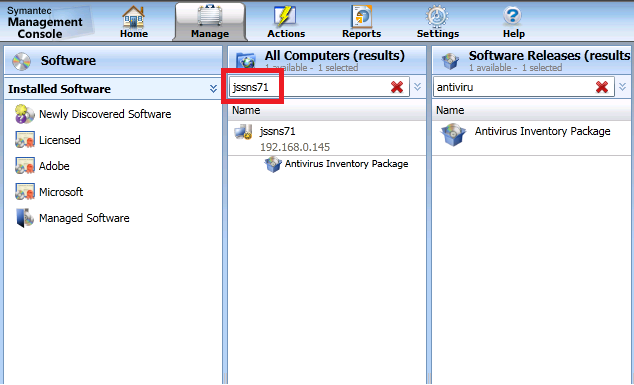


## Drag and Drop

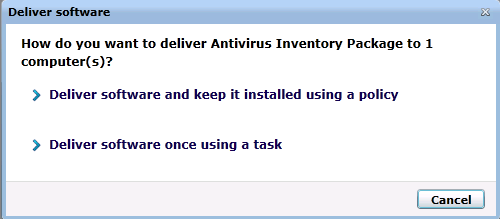
The Activity Center gives the option to use drag and drop. This works well for quick one-off deployments, enabling an administrator to drag a particular Software Component to a single computer. This also works dragging software to a predefined filter. The filters must show under User Created Resources within the All Computer Views section under Computers. There are some caveats to be aware of when using drag and drop with Software Management. Also there are some methods suggested when rolling out software in this manner.

The following walkthrough takes you through the process:

1. In the Symantec Management Console browse under Manage > Computers.
2. In the All Computers view, use the Search field to filter out until you have the computer desired.
3. Now click on the Software section to switch over to the Software section.
4. Highlight the Software Releases section in the left pane so all software that is deliverable is shown.
5. Use the search field to filter down until you select the software desired.
6. Once the software is highlighted, left-click and hold down on the icon to the left of the name.
7. The last computer view you had selected under the Computers section will fly out, as shown in this screenshot:



1. When you let go of the selected Software on top of the desired computer, you will get prompted for how you want to deliver the software, as shown:



1. The first option will deliver the software via a Managed Software Delivery Policy, while the second option will use a Quick Delivery Task.
2. As an alternative, if you wish to target a filter, instead of dragging the option to the computer list, drag the icon down to the section header for Software. Holding the pointer over this section will expand it, then allowing you to navigate through the User Created Resources. Hovering allows you to open subsections, etc.
3. Drop the software on the desired filter to complete the process.



NOTE! When you use this method a single Managed Software Delivery Policy is generated, along with a target for that one system. If this method is used a lot, the system will begin to be flooded with these one-off policy requests. This issue will be addressed in 7.1 SP2 where it will look for an already existing Managed Delivery Policy to service the request to avoid excessive duplicates.

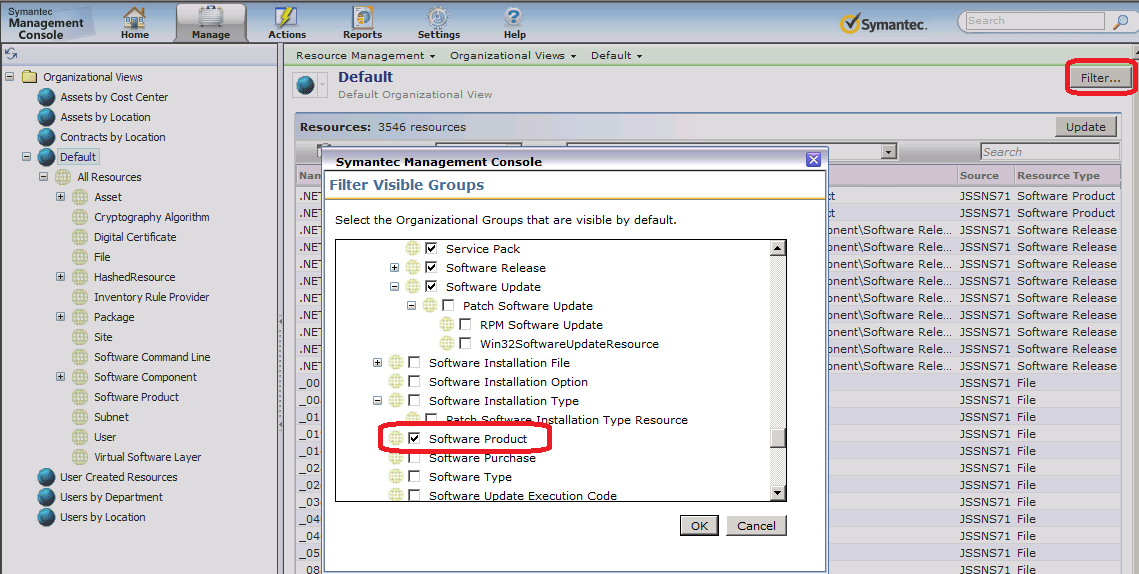
# Managing Software Products

The emphasis on Software Components in 7.0 created some logistical or management nightmares when it came to tracking what software was out in the environment. By taking a step back to the Product level, most of these issues have been made moot. Software Components, whether captured during Inventory or created manually by an administrator, will now by manually or dynamically assigned to a Product. This removes any need to reconcile duplicate resources as the Product rules will automatically assign them to the correct Product.

## Reviewing known Products

You can view Products under the Resources screens. The following provides the full steps of how to review Software Products:

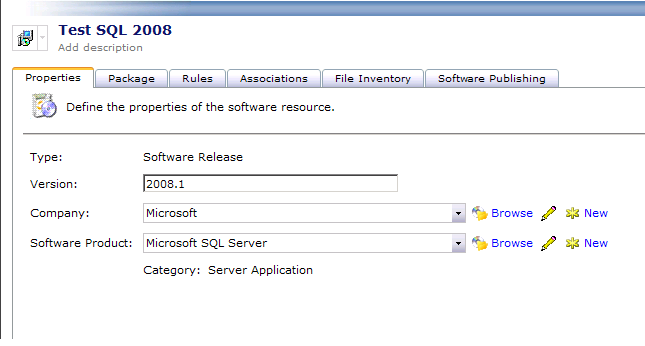
* 1. To view what Products are available, in the Symantec Management Console browse under Manage > All Resources > and select the “Default” node. The following steps will add the category if it is not already available in the tree list.
  2. In the right pane click the Filter… button.
  3. In the list, find the entry for Software Product and ensure it is checked, as shown in this screenshot:



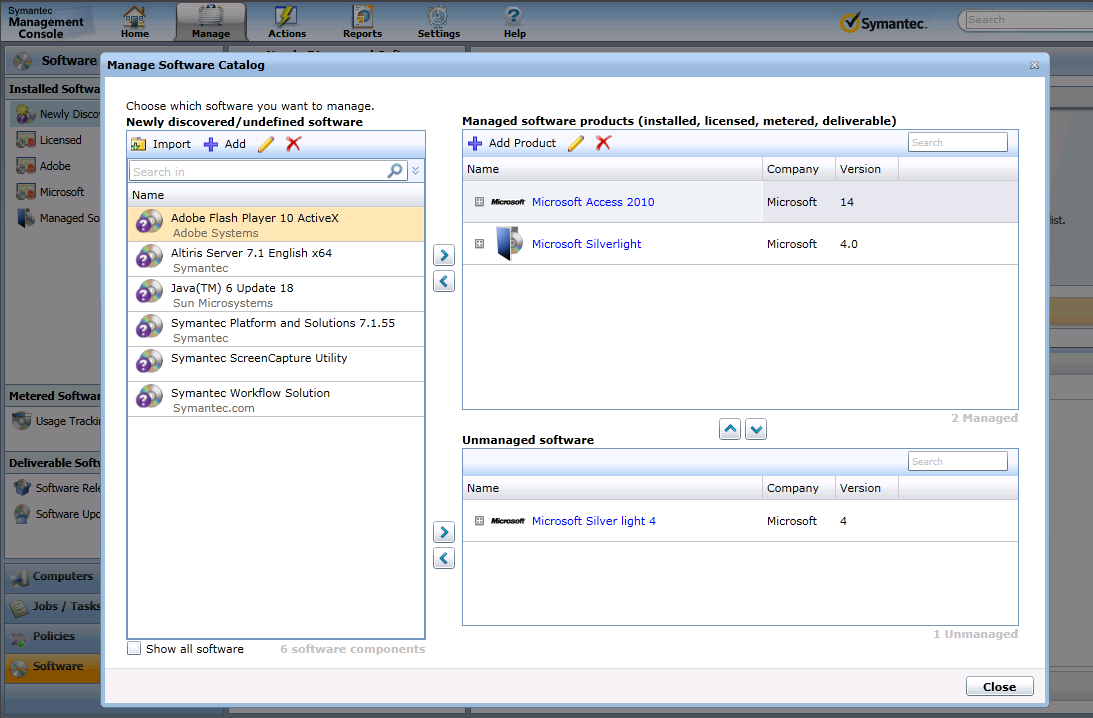
* 1. Click OK to add the category.
  2. In the resulting left-hand tree, browse under Default > All Resources > and select Software Product.
  3. You can now search through what Software Products are available.

Note that you cannot edit Products from this location. You can open Resource Manager, which will give you additional information than what is displayed in the grid. Products can be created and managed through Asset Management, or through the Software Catalog. Since not everyone will necessarily have Asset, I will focus on the Software Catalog.

To access the Software Catalog, in the Symantec Management Console browse under Manage > and select the Software Catalog. This will load the Catalog interface as a pop up window. Note that all active Products will be shown in the upper right pane, as shown in this screenshot:



From here you can assign, create, or edit Software Products using a simple UI provided by the Software Management Framework. However in 7.1 SP1 this field has been removed, which pushes you to use the Silverlight Catalog area for assigning, editing, and creating Software Products. This ensures the correct method and details are used/provided.



First, you can search for existing Software Products using the search field.



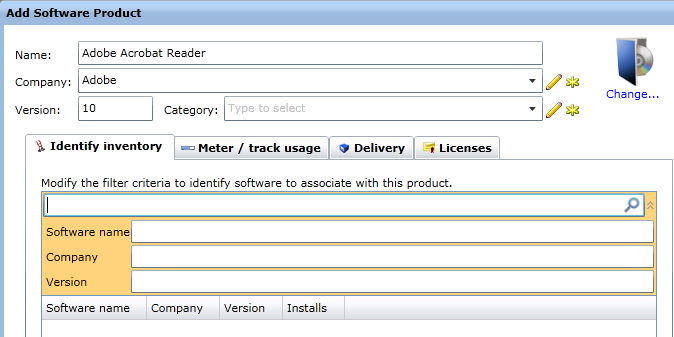
You will notice that not all Software Products are shown in this list. Only manageable Software Products will show. For a Product to be managed it needs to be associated with a Software Component.

I wanted to point out that there is a significant change in the edit mode for a Software Resource between 7.1 and 7.1 SP1. This was in an attempt to let the NS automatically manage discovered Software Products. This ensures that the assignment is done correctly. As previously shown, in 7.1 you had the ability to manipulate the Product directly from the Software Resource Edit/Creation screen. Now you must use the new UI.

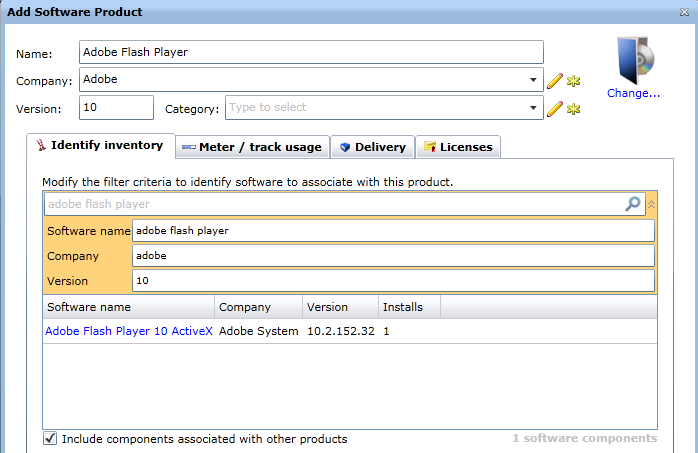
## Creating Software Products

To create a Software Product, follow these steps. These steps can also be observed when editing a Software Product.

1. In the Symantec Management Console browse under Manage > and select Software Catalog.
2. When the Manage Software Catalog window appears, click on the Add Product button.
3. The top section of the dialog is for labeling and identification purposes. It will not be used when calculating or auto-assigning Software Components to the Product. Provide your Product’s details, as shown for an example in this screenshot:



1. Provide one or more values in the provided three fields. This will automatically search for Software Components that match the criteria, as shown in this example:

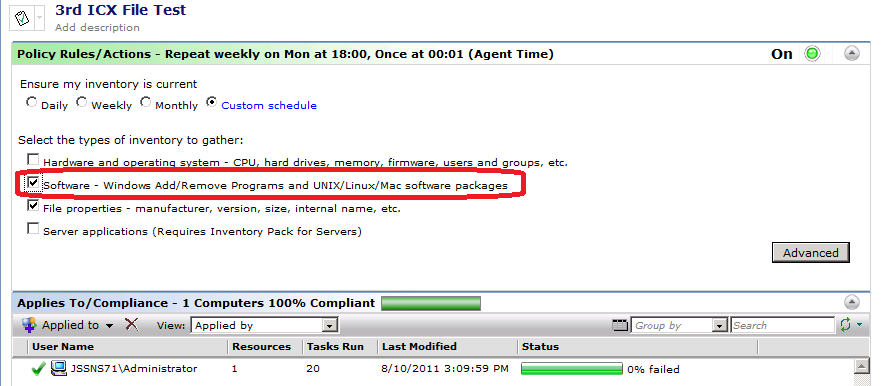


1. Note that it found one match based off the criteria I provided.
2. Check the box “Include components associated with other products” to ensure you are not missing components based on previous associations (whether made manually or automatically).
3. You should fine-tune the values so it includes, and excludes, the software resources you want. Review these examples:
   1. In the above example, if I only wanted the ActiveX components and not the actual flash player, I could change the Software name to: Adobe Flash Player 10 ActiveX.
   2. If I only wanted the 10.2 versions of the ActiveX to be associated here, I would change the Version value to 10.2, thus excluding any other 10.x versions.
   3. If Adobe had a release of this software that had a misspelling in the Company name (i.e. Addobe) I could remove the Company designation altogether, if I trust the other two criteria/values.
4. Click OK to make the change.
5. Note that the associations will be made immediately for the software that shows up in the lower list. When new software comes in that matches the criteria, the association is made during the following Scheduled Task:
   1. NS.Nightly schedule to associate Software components to software product…
6. Done!

Once you have made the associations you are done as far as assigning Products go.

NOTE: It has been noted that only certain Products are displayed in this user interface. If a Product does not show up, it can only be modified if Asset is installed. If it doesn’t show in the list, it does not have enough criteria and a new one should be generated.

**Software Inventory**

You may have noticed that in the Software Catalog Add Product interface the default tab is Identify inventory. The Inventory that this ties into is the Software Discovery module. This is captured during the Software Inventory Policies. The specific option within the Inventory Policy interface is shown in the screenshot below, namely Software – Windows Add/Remove Programs and UNIX/Linux/Mac software packages:  


This is the process that provides the Inventory needed to make the associations between Software Components and Products.

# The Software Portal

Troubleshooting issues where the Software Portal is not working correctly can be challenging. Much of what happens is not visible in either the Software Portal itself, or through the Symantec Management Console. This white paper provides greater details in how the Software Portal works, which should provide methods to assist in resolving issues. These details can be used to assist in troubleshooting as the process can be followed from start to finish, from when the user loads the Software Portal to when the Software is delivered to the target machines.

## Common Terms and Acronyms

The following acronyms will be used commonly throughout this section:

* SMP – Symantec Management Platform
* NS – Notification Server
* Child – A child Notification Server within a hierarchy
* Parent – The parent or top level Notification Server within a hierarchy
* SWP – Software Portal –   
  *NOTE: Often the code base that runs the Software Portal, including the Assemblies and code called through the Altiris Service, will be referred too as the “Software Portal”, or “SWP”*
* SWM – Software Management Solution
* SMF – Software Management Framework
* MSD – Managed Software Delivery Policy
* AD – Active Directory
* SID – Security Identifier
* SLA – Service Level Agreement

## Software Publishing

There are two basic objects that can be published to the Software Portal. A Software Resource can be published directly to the Portal, and in MR4 each Command Line available in the Resource can be individually published. A Managed Software Delivery can also be published directly to the Software Portal.

While this article will not cover the basic usage in any details, know that you can publish to single users or groups, and there are two settings associated with each user or group:

1. Approved or Requires Approval radial – This allows administrators to control when software is pushed to a user who requests it, or allow free approval for common applications.
2. Recommended – This checkbox should be checked for any Software you want to appear in the Portal by default when a user launches the SWP.

When a Resource or Policy is published to the portal, a record is logged into the Item table, with a set of referenced records in the table **SWP\_PublishingItemSetting** (See **Figure 3 – Software Portal tables**). These references allow the security settings tie into Users who load the Software Portal. The relationship between the Publishing Item (in the Item table) and the Publishing item settings in **SWP\_PublishingItemSetting** are a one-to-many relationship.

When you save the settings and users or groups assigned under a Software Publishing tab (whether for a MSD Policy or a Software Resource) the stored procedure **spSWP\_UpdatePublishingSettings** is called to add the appropriate entries to the **SWP\_PublishingItemSetting** table.

These references are made using the **ItemReference** table. This table contains the following columns:

* ParentItemGuid – The parent of the reference, by Item or Resource Guid.
* ChildItemGuid – The child of the reference, by Item or Resource Guid.
* Hint – This is the type of reference that the row represents.

The Publishing Item, located in the Item table, has only one type of reference, as illustrated by this table:

|  |  |  |  |
| --- | --- | --- | --- |
| Reference hint | Reference type | Parent Item | Child Item |
| PublishingItem to Software | Depended child | Software Resource or MDP | Publishing Item |

Note that the Hint is “PublishingItem to Software”, which is the exclusive type for the Software Portal’s reference from a Publishingitem to its PublishingItemSettings.

Please see **Figure 1 – PublishingItem Diagram** for more information.

### Software Resource Publishing

Previous to SWM 7.0 SP2 MR4, the Software Portal choose the default Command-line for the type “Install” automatically when you published it to the Software Portal. While this works for basic end-user publication, it did not fit those who gave greater access to command-types to their Helpdesk or IT personnel. In light of this now the Command-lines can be associated directly, giving the flexibility that was found in the 6.x Portal, while still providing the more user-friendly 7.0 Software Portal.

When a specific Command Line is published to the Portal, a PublishingItem is created for that publication. This means if multiple command lines within the Software Resource are published, each command-line will have its own PublishingItem, with corresponding PublishingItemSettings.

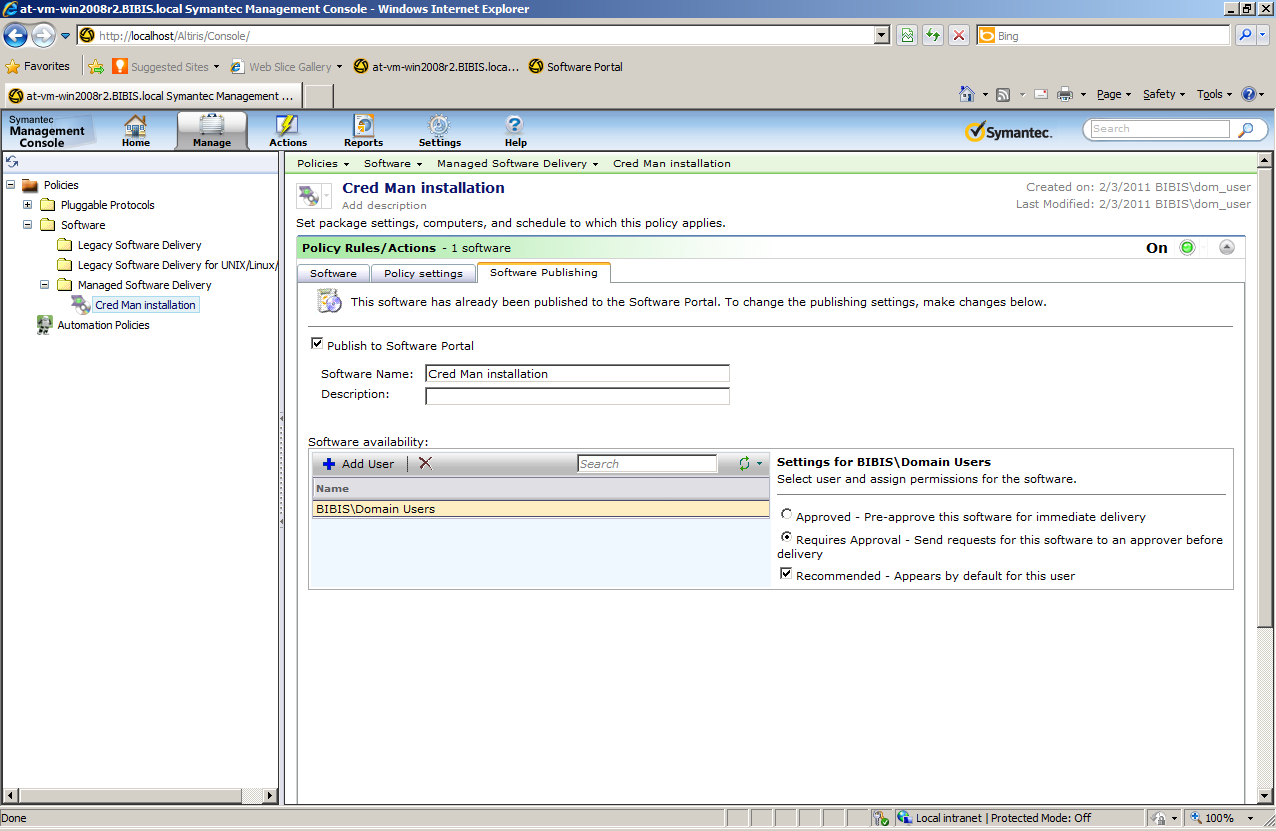
For the radial setting “approved” or “requires approval”, two PublishingItemSettings are created, for each user and group in the list, one for approved and one for requires approval.



**Figure 1 - Publishing Software**

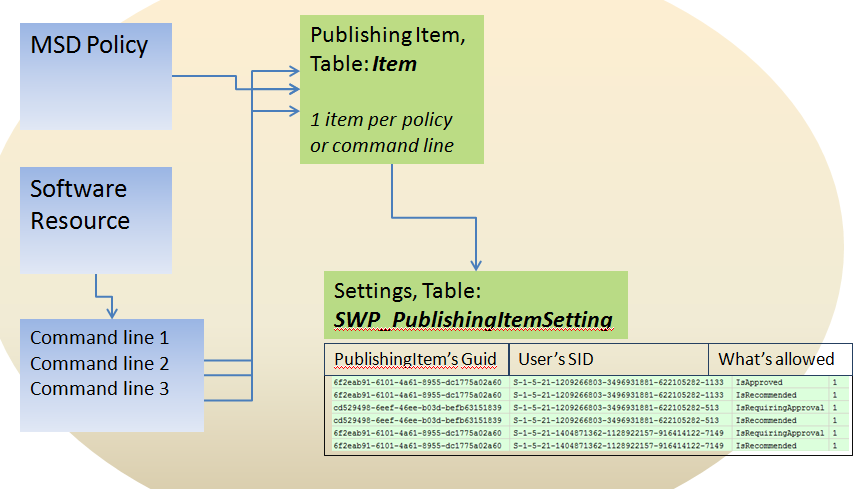
### Managed Software Delivery Policy (MSD) Publishing

The basic associations and references created for a Managed Software Delivery Policy are the same as for a Software Resource and its command lines. A MSD Policy is ever only a single PublishingItem, though the same variety of PublishingItemSettings are available depending how many groups and users and what settings are selected, as shown in **Figure 2**.



**Figure 2 - Publishing MSD**

This section may have additional items added as additional research is conducted on the Software Portal and how it works in conjunction with a MSD Policy.



**Figure 3 – PublishingItem Diagram**

## Software Request

A Software Request is a key feature of the Software Portal. It’s what makes the Portal work when an end-user or an IT administrator or Helpdesk user requests to install Software on a system they are logged onto. Whether a user has any Software selected for them or not, directly from their user account or through a domain group, they will be able to open the Portal.

To put it more succinctly, as long as the User can authenticate on the Domain, they will be able to load the Software Portal by putting in their credentials when prompted to load the SWP.

### User Resources Intro

When a user first accesses the Software Portal, if the user already exists within the SMP (whether through a Connector Solution or an Active Directory import) the Portal will tie into that User Resource object. If no user exists within the NS, a User Resource with a naming scheme of name.domain is created. The user will be prompted to fill out information concerning their profile, including their email addresses for notification purposes.

More technical information is not available on the initial user setup or User Profile at this time. Please see User Resources section under the **Software Approvals** for more information on User Resources.

There are ways to manipulate how this Profile page appears when users first access the SWP. These are unsupported. If a problem occurs while trying any of these customizations, Symantec is not obligated or required to assist fixing the issue.

While speaking with a user he indicated there was nothing to stop an end user from making any random request for software using the “Request Unlisted Software” option, a request that may fall into a black hole if the Portal is not setup to require approval. If you decide to try some of these customizations, please backup the files you edit before making changes so you will have the ability to revert your changes should you have unexpected results.

*NOTE: These changes are not supported and are provided "as is". Use at your own risk as there are no guarantees and no support if you implement one or more of these methods. See the instructions below to ensure you backup the original configuration before applying changes in case you need to revert.*

One button that an administrator might not want his Symantec Administrators to use in the Software Portal is the “Manage” button.

https://imconsole-kb.ges.symantec.com/library/BUSINESS/Joel_Smith/custswp1.jpg

Change the style=”display:none” for the Manage button in the Home.aspx file found under the Software Portal installation directory <ADDIT>

                                <td rowspan="2" class="toolbarButton" id="btnManage" runat="server" onmouseover="ToolbarButton\_OnMouseOver(this,'background-image: url(../images/headerToolbarBackgroundHighlighted.png)');"

                                    onmouseout="ToolbarButton\_OnMouseOut(this);" onclick="ToolbarButton\_OnMouseClick('/Altiris/SoftwarePortal/ManagerPortal/ViewSoftwareRequest.aspx');"

                                    style="background-image: url(../images/headerToolbarBackground.png**); display: none**">

                                    <div style="height: 41px; width:64px;">

                                        <img alt="" src="../images/btnManage.gif" class="toolbarImage" id="btnManageImg" runat="server"/>

                                        <div class="toolbarText" style="color: #000000;">

                                            <%= GetLocalisedString("Manage") %></div>

                                    </div>

                                </td>

To hide the “Request Unlisted Software” button, there are several methods, including rewriting the ManualRequestConfirmation.aspx to state that this button shouldn’t be used for now, but for most of us making a simple edit is the preferred method.

https://imconsole-kb.ges.symantec.com/library/BUSINESS/Joel_Smith/custswp2.jpg

To remove this button, add Javascript code into the AddRemoveProgram.aspx page that would hide the Request Unlisted Software button.

Into the existing <script> add:

                function DisableRequestUnlisted()

                {

                                var toolbar = ExpSecCtrlAvailableSoftware\_gcSoftwareToolbar;

                                if (toolbar == null)

                                {

                                                setTimeout('DisableRequestUnlisted', 200);

                                }

                                else

                                {

                                                toolbar.toolbarButtonControls[2].setEnabled(false);

                                }

                }

And to the body tag add attribute onload="DisableRequestUnlisted()"

You must verify how does gcSoftwareToolbar gets resolved. This control can get a different name depending on the version of ASP/IIS.

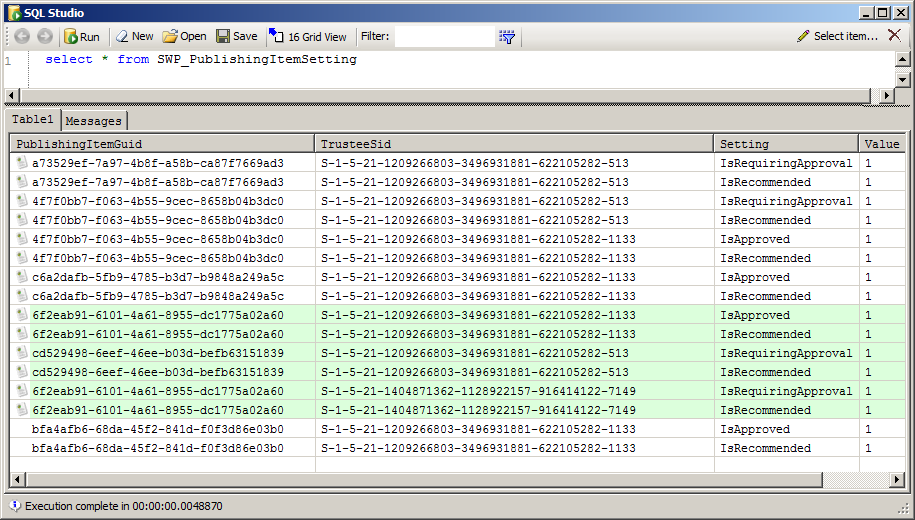
Other modifications can customize the initial User Profile prompt and the Software Portal itself. If you do this extensively, please make two copies of all files, one for the custom portal and one for the default. This will enable Symantec Support to troubleshoot issues that do not stem from customizations by accessing the default Portal.

### User Interaction

Users who access the Software Portal will only view and request published Software Resources and MSD items (with a publisheditem) that are available for him or her. The Software Portal is designed to work with NT users and groups (local), or Microsoft Active Directory. For any software to show up on the portal, the accessing user must have assignments of a PublishingItem to his or her User account, whether directly or through Group assignment.

Every local or AD User has a SID (Security Identifier), which is a unique identifier in Active Directory, or within the local NT security for the NS. Every NT group also is assigned such a SID.

The **SWP\_PublishingItemSetting** table contains relations between PublishingItems and SIDs of users and groups for which these publishing items are available. Figure 4 shows how this association is represented in the table:



**Figure 4 – SID Associations**

### Requests and Updates

Any PublishingItem can have none or many associated Request Items. Basically whenever someone makes a request of a piece of software (hence the PublishingItem) a RequestItem is generated and associated to the PublishingItem. As with the PublishingItem, A RequestItem is stored in the **Item** table. Yet this is only half of the request. The other half is stored in the **SWP\_SoftwareRequest** table.

When a request is made the requires approval, or an administrator fulfils, denies, or asks for more information on the request, a Comment, labelled RequestComment, is generated. Any RequestItem can have none or many Comments, depending on how many notes are made during the process. Any Comment is stored in SWP\_RequestComment table.

When a RequestItem is created or updated, the Software Portal uses the stored procedure named **spSWP\_UpdateRequestTable** to insert or update records in the **SWP\_SoftwareRequest** table.

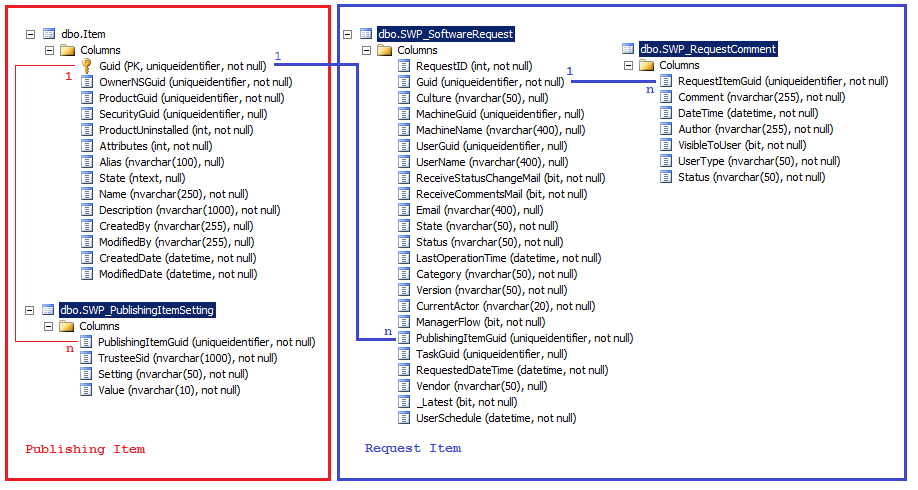
A RequestItem has 4 types of references. Like a PublishingItem, these references are found in the **ItemReference** table, as shown in this grid:

|  |  |  |  |
| --- | --- | --- | --- |
| Reference hint | Reference type | Parent Item | Child Item |
| SoftwareRequest to PublishingItem | Related child | Publishing Item | Request Item |
| SoftwareRequest to Task | Singleton child | Delivery Software ? | Request Item |
| SoftwareRequest to PortalUser | Singleton child | User Resource | Request Item |
| softwarerequest created resourcetarget | Related child | Target Resource | Request Item |

Please see **Figure 5 – Software Portal tables** that illustrates the associations within a RequestItem.

Legend:

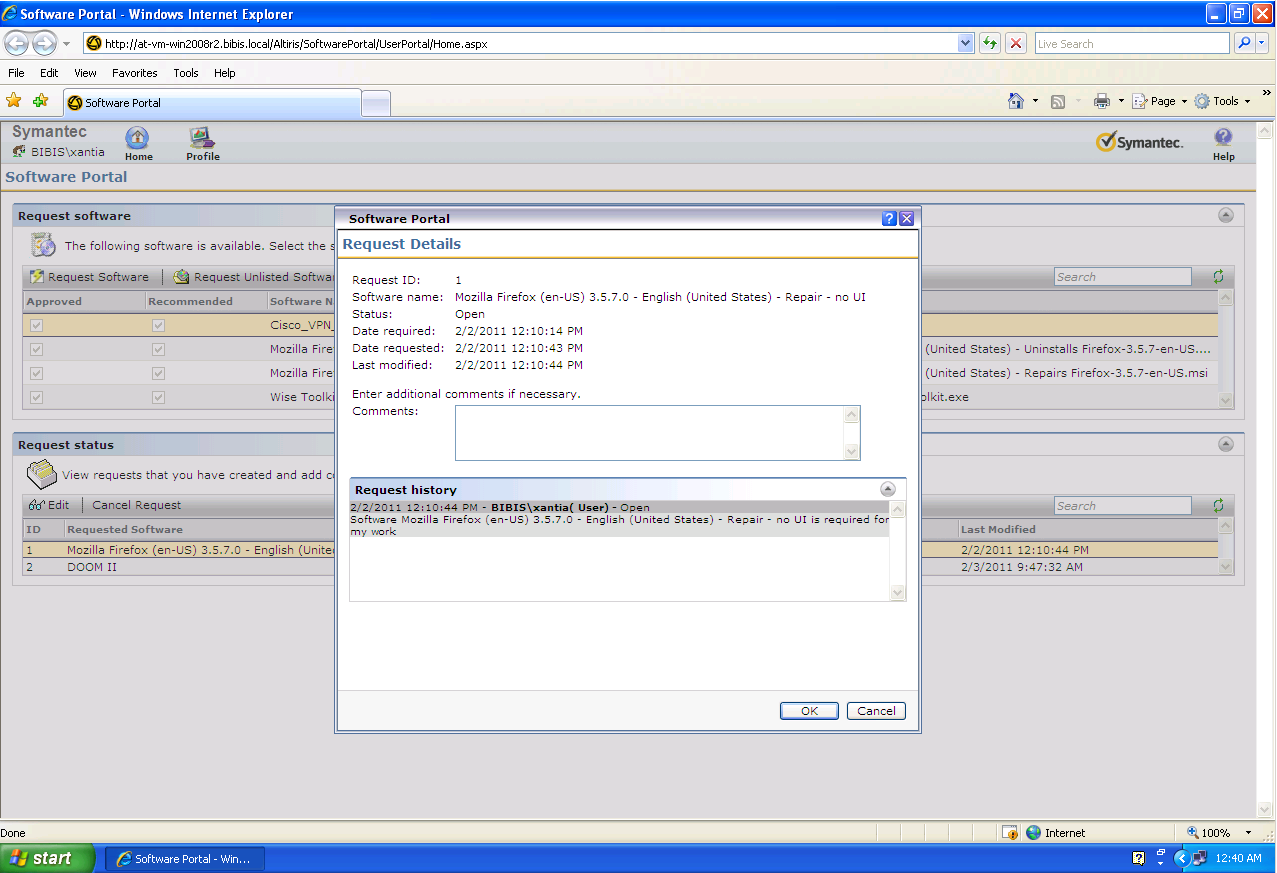
* 1 ------- n – This association means a one to many, or 1 to (number) associations or references.
* Red box: PublishingItem
* Blue box: RequestItem
* Red line: PublishingItem association
* Blue line: RequestItem association



**Figure 5 - Software Portal tables**

### Request Dialogs

The RequestComment object is created by using the Request Details dialog. Please see **Figure 6 – Request Details** for a screenshot of the dialog. Note that comments are not required, so no RequestComment association is required during the Approval process. In many cases the administrator will not comment when approving a request, only when denying. No one wants to receive a call because of a mysterious denial with no reason given, so it is highly recommended!



**Figure 6 - Request Details**

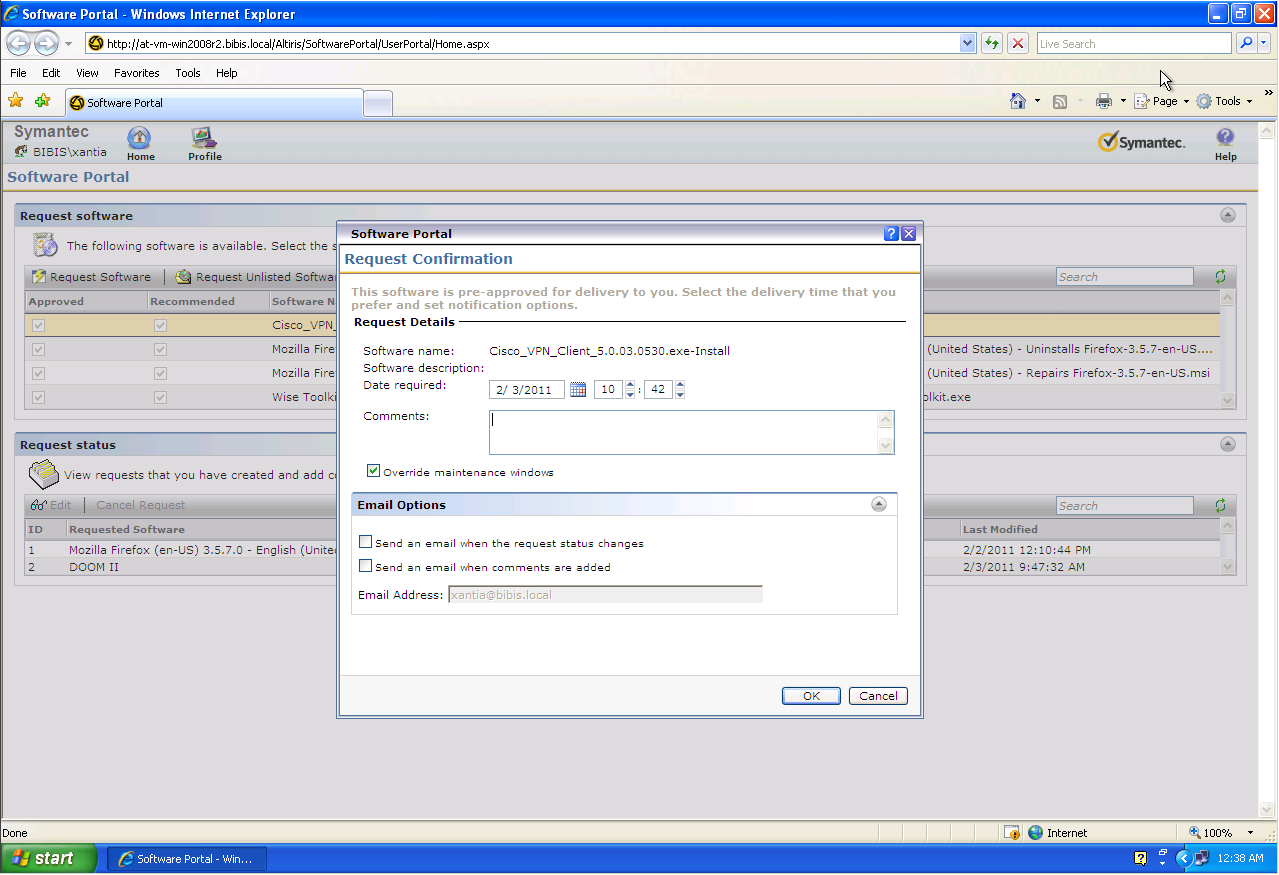
When requesting Software that is available but requires approval, the dialog Provides a number of options for the User to fill in. Certain inputs are strictly data driven, meaning they do not affect how the Portal handles the request. They are separated in this list:

Data only:

1. Date required – This is a notification only. This will not change how the Administrator receives the request, but is added so that users can provide a needed-by date as an FYI.
2. Comment

Functional options:

1. Override maintenance windows – If a user removes the check from this option, it will affect how quickly software is delivered, if maintenance windows are in use in the environment. It is recommended to steer users away from this as most users will want software as soon as possible if they’ve gone through the trouble to request it.
2. Send an email when the request status changes
3. Send an email when comments are added
4. Email Address: - This option is typically greyed out as the User will have already supplied an Email address when they initially setup their User Profile on first Portal load.

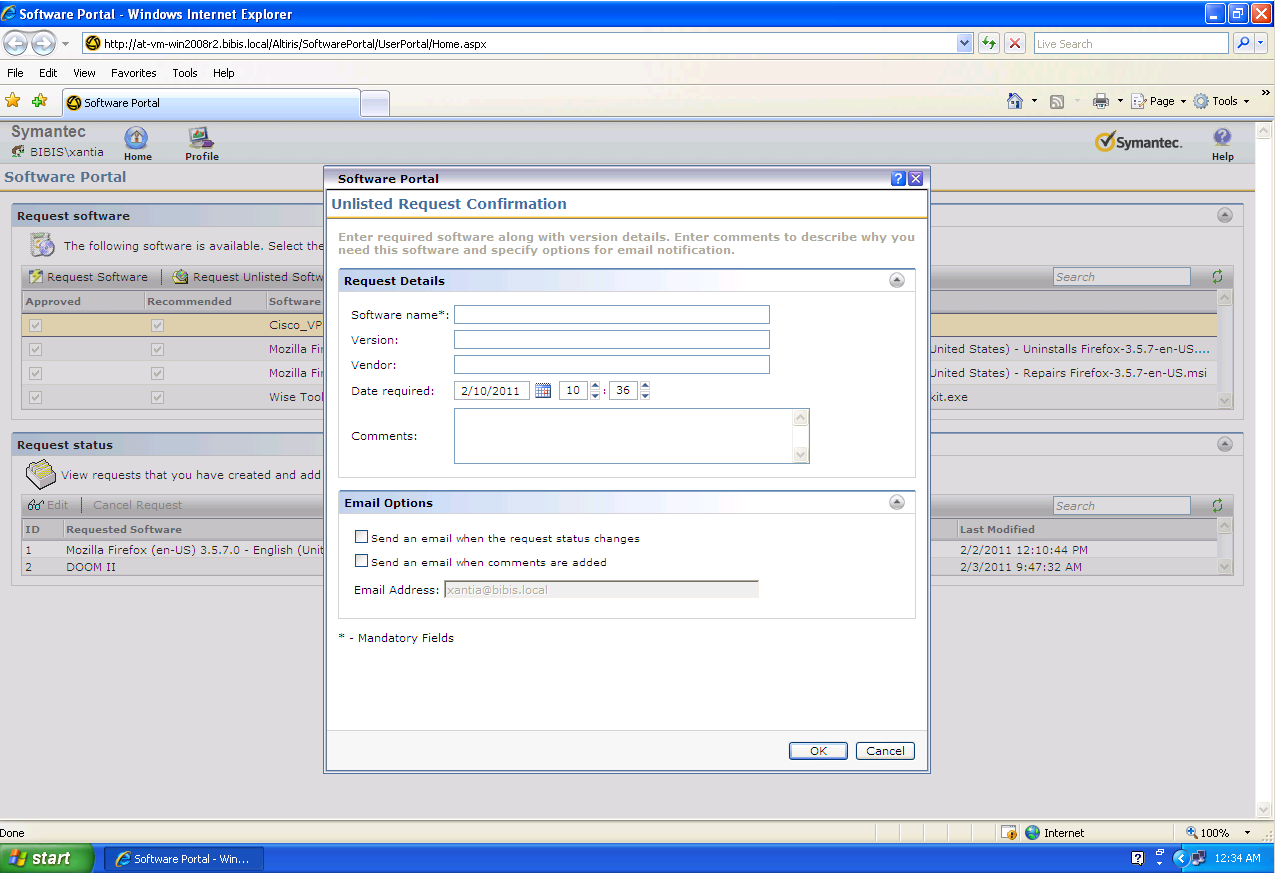


**Figure 7 – Listed Software Request**

The next dialog, found in **Figure 8 – Unlisted Software Request**, has turned out to be a bit controversial. Most administrators do not want to worry about providing possibly unmanaged software to users. The button, therefore, gives End-users potentially unrealistic expectations on how unlisted software requests are handled. Previously, under the **User Resources** section, a method for removing this button is provided (with a disclaimer, be careful!).

In the Unlisted Software Request dialog, most of the fields are data-only. The email checkboxes are the only functional ones. It is this open-ended mechanism that can be problematic for Administrators as these requests appear as the User inputted them. On the flipside it does allow users to request software that is not available to them in the Software Portal. If the SLA of IT or the Helpdesk includes this level of service, then it is built-in to the Portal.

The administrators will receive the request much like any request, only the fulfilment of this request will need to be done manually by creating a Software Resource and publishing it, or pushing it out via a Quick Delivery or MSD Policy.



**Figure 8 – Unlisted Software Request**

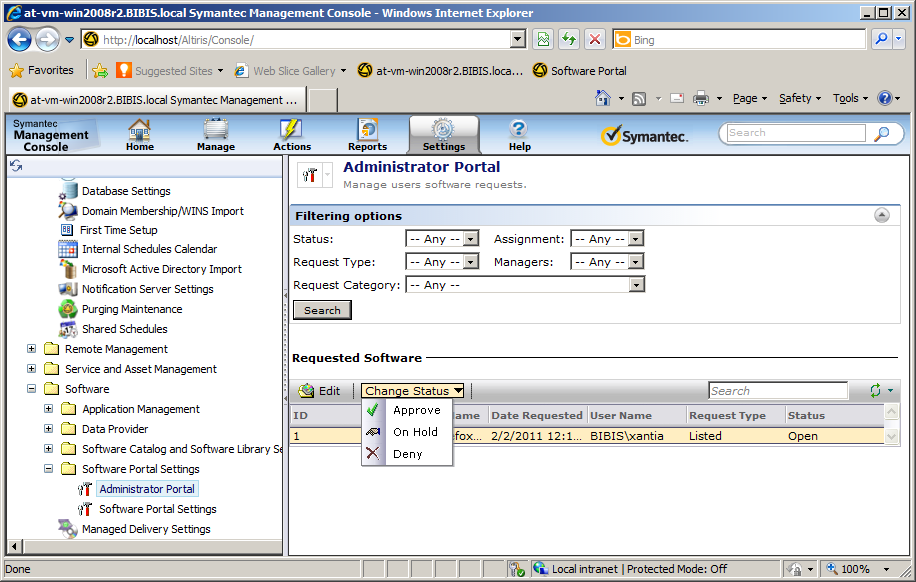
## Software Approval

When a RequestItem that requires approval is generated, the Portal Administrators and possibly Portal Managers are sent the request. The Manager or Administrator can then do one of several things:

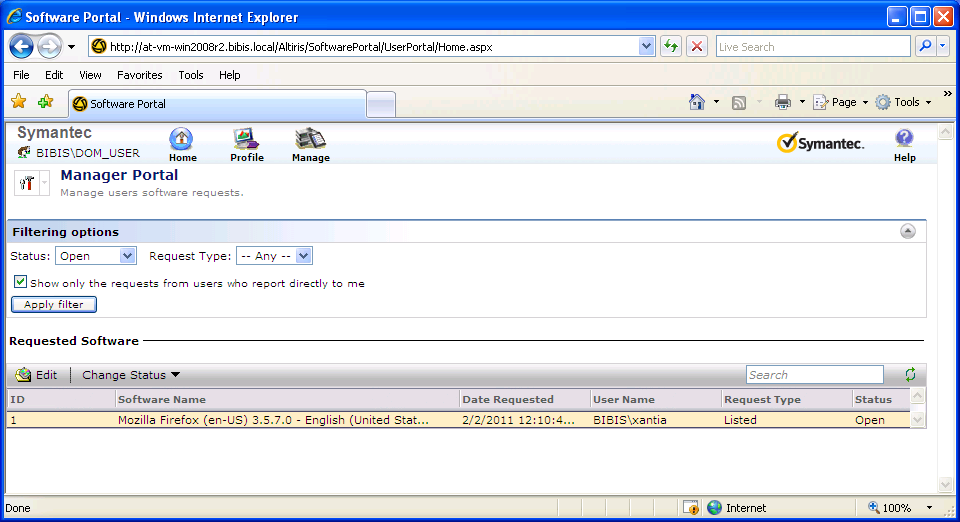
* Approve
* Deny
* Place on Hold – This will leave the request unanswered, and the end-user can see that the request is on hold. It is recommended to add a comment as to why an item is on Hold. For example:
* Commented – Portal Admin added comment: “IT is in the process of acquiring more licenses for the application you have requested. When obtained, this Software will be approved and delivered to your computer.”

### Portals

What is the difference between the Administrator and the Manager Portals? Please see **Figure 9 – Administrator Portal Page**, and **Figure 10 – Manager Portal Page** for screenshots of the Portals.



**Figure 9 - Administrator Portal Page**

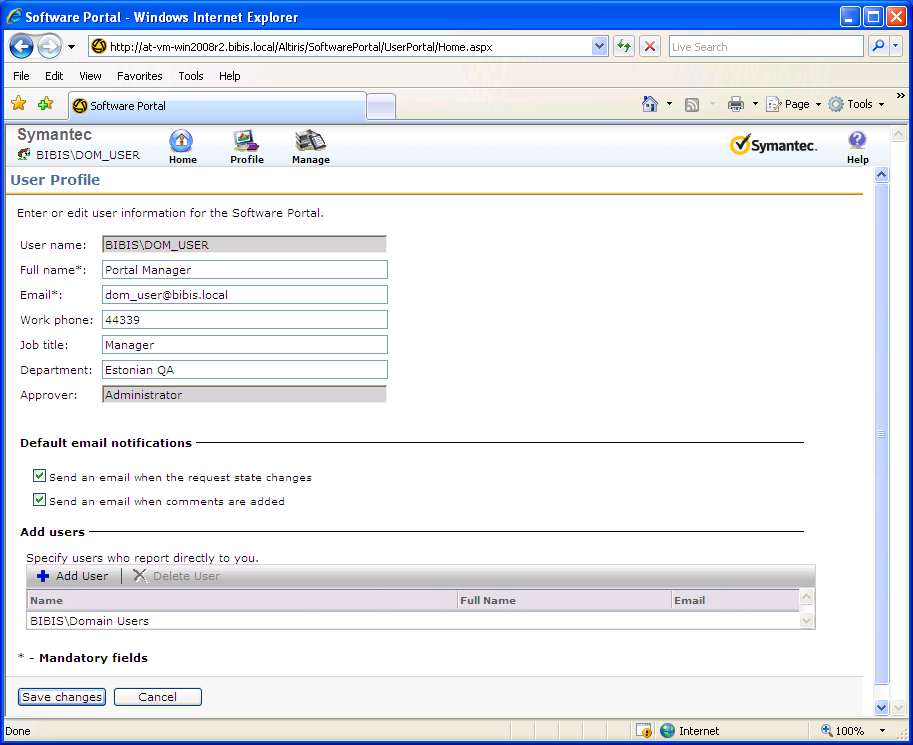


**Figure 10 - Manager Portal Page**

The Administrator Portal can be accessed through the Symantec Management Console (SMC) and gives the Admin full access to all Software Requests currently in the system. This portal is found at AdminPortal\ViewSoftwareRequest.aspx. Any action available for a Request can be done by the Admin as he or she has full privileges. It is NS Role and Scope security that defines who has access to the Administrator Portal. By default the Application Identity and all Symantec Administrators have rights.

The Manager Portal is only accessed from the actual Software Portal. When a user who has Manager rights loads the Software Portal, they will get three “tabs” available as opposed to the usual two. The Manage tab provides access to requests that Manager has access to. This portal is found at ManagerPortal\ViewSoftwareRequest.aspx.

The Portal Manager can only edit requests from his subordinates. These subordinates are defined on the User Profile Page under the section "Add users" (this section is visible only for Portal Managers and Administrators). See Figure **Figure 11 - User Profile Page** for a screenshot.



**Figure 11 - User Profile Page** (UserProfile.aspx)

The “Add User” button allows the usage of both NT Local users and groups, and AD Domain users and groups. If AD is setup correctly, the Managers can add those they have access to at this UI. This will automatically route requests for any users in the list, or users who belong to groups in the list to the Manager Portal, allowing the Manager to fulfil the request.

If a subordinate is a domain group it means that ALL users from this group and from ALL children groups (recursively) are subordinates, and this Manager will be able to access and fulfil all requests.

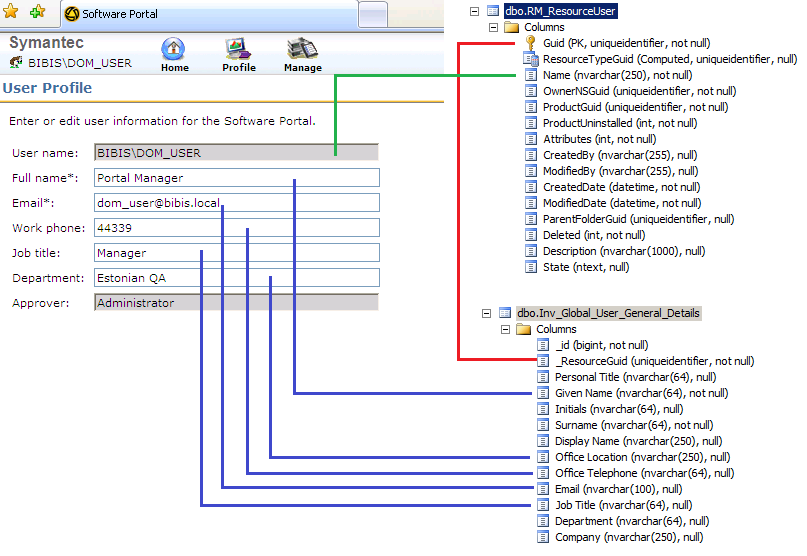
### User Resources

Any Portal User, including the Portal Administrator and all Portal Managers create or update a User Resource in the SMP. This Resource is used as the Portal identity for any user that needs access to the Software Portal. This “Link” to existing User Resources within SMP, or the creation of a new Resource, occurs when any users opens the Software Portal for the first time and is prompted to full out the User Profile page.

See **Figure 12 - Resource User data mapping** to see how the data maps:

Legend:

* Red Line: This links the two halves of the User Resource together.
* Green Line: Data for the primary User Resource table
* Blue line: Data for the auxiliary User table that contains additional details



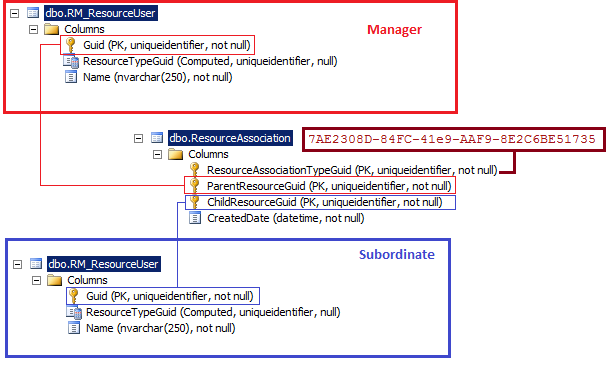
**Figure 12 - Resource User data mapping**

**Warning!** The data the user enters in the User Profile page is added or created as part of that User’s User Resource in the Symantec Management Platform. This may allow a user, as one IT manager told me, to name himself “Captain Kangaroo”, with a job title of “Admiral”, in the “Oceanography” department.

When a user first accesses the Software Portal, the following process is used to either promote the existing User Resource to a Portal User, or create a User Resource to give that user access to the Portal:

1. The User accesses the Portal for the first time and is prompted for credentials. They enter their Domain credentials.
2. The Software Portal attempts to find the User Resource within the SMP by the user's login name, namely **Name**.domain. The Stored Procedure used is: "**spSWP\_GetPortalUserGuidFromNT**".
3. The SWP will create a new User Resource if one is not found.
4. If values already exist in the target tables for existing User Resources, it will skip those fields. Otherwise, the SWP will insert the User Resource fields from the User Profile Page and commit it (see mapping in **Figure 12**).
5. Lastly, it takes the subordinates a Manager or Administrator may have added and saves the Parent Resource Associations for them. See **Figure 13 – Relation between Portal Manager and Subordinates**.

If a Manager of Administrator fills out the Users, additional steps are taken to create or use existing User Resources for their subordinates. This is true for any User or Group. Relation between Portal Manager and his subordinates is implemented as a Parent Resource Association between the User Resources with Guid "7AE2308D-84FC-41e9-AAF9-8E2C6BE51735", as shown in **Figure 13**.



**Figure 13 - Relation between Portal Manager and Subordinates**

### Filtering the Admin or Manager Portals

Depending on the size of the organization, and the availability of Managers for the Software Portal, filtering may be crucial to properly administering the various Software Requests that may exist.

A Portal Administrator can filter out requests by the following criteria:

* Status: Any (default), Open, Approve, On Hold, Deny
* Request type: Any (default), Listed, Unlisted
* Request category: Any (default), Approved Software, Software Requiring Approval, Approved MDP, MDP Requiring Approval, All Approved, All Requiring Approval
* Assignment: Any (default), Admin, Manager
* Managers: Any (default), [List of ALL Portal Managers – uses the **spSWP\_GetAllManagers** stored procedure]

A Portal Manager can filter out requests from his subordinates by:

* Status: Any, Approved, Denied, Open (default), On Hold
* Request type: Any (default), Listed, Unlisted
* Show only requests from users who report directly to this manager or show requests from subordinates of subordinates of this manager (recursively), if such relationships exist.

The first step in the process when an administrator or manager of the Software Portal logs onto the Administrator or Manager portals, respectively, is to filter out requests according to the selected Status, Request type, Request category, and Assignment options. For the Administrator portal, the stored procedure used is quite simple, namely **spSWP\_Admin\_GetFiltereRequest**. This stored procedure executes the following type of SQL query:

SELECT req.\*, i.Name

FROM SWP\_SoftwareRequest req, Item i @WhereQuery

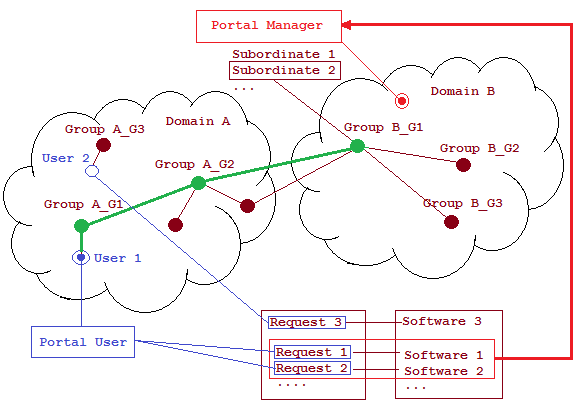
The @WhereQuery point is a single parameter of the stored procedure, and is a WHERE clause that the Software Portal constructs from the Web.AdminPortal.ViewSoftwareRequest.aspx page according to the selected filter options.

In the case of a Portal Manager the stored procedure **spSWP\_Manager\_GetFilteredRequest** is used. Its input parameters are request's state, status and category (see **SWP\_SoftwreRequest** table, State, Status, Category fields). In this way the Portal is using the Manager’s own assignments to populate what requests he or she sees. This leads into the next step.

The Manager Portal then filters out requests based on what users or subordinates the Manager has assigned to him or her. This step is optional for Portal Administrator (only if a manager is selected in the filter). Of course for a Manager the filter happens automatically. This filtering is implemented inside the Software Portal’s .NET assemblies within the GAC, and is common for both a Portal Manager and the Portal Administrator.

The SWP then uses an Algorithm with input that is the set of request items filtered for in the first step, and is executed as follows:

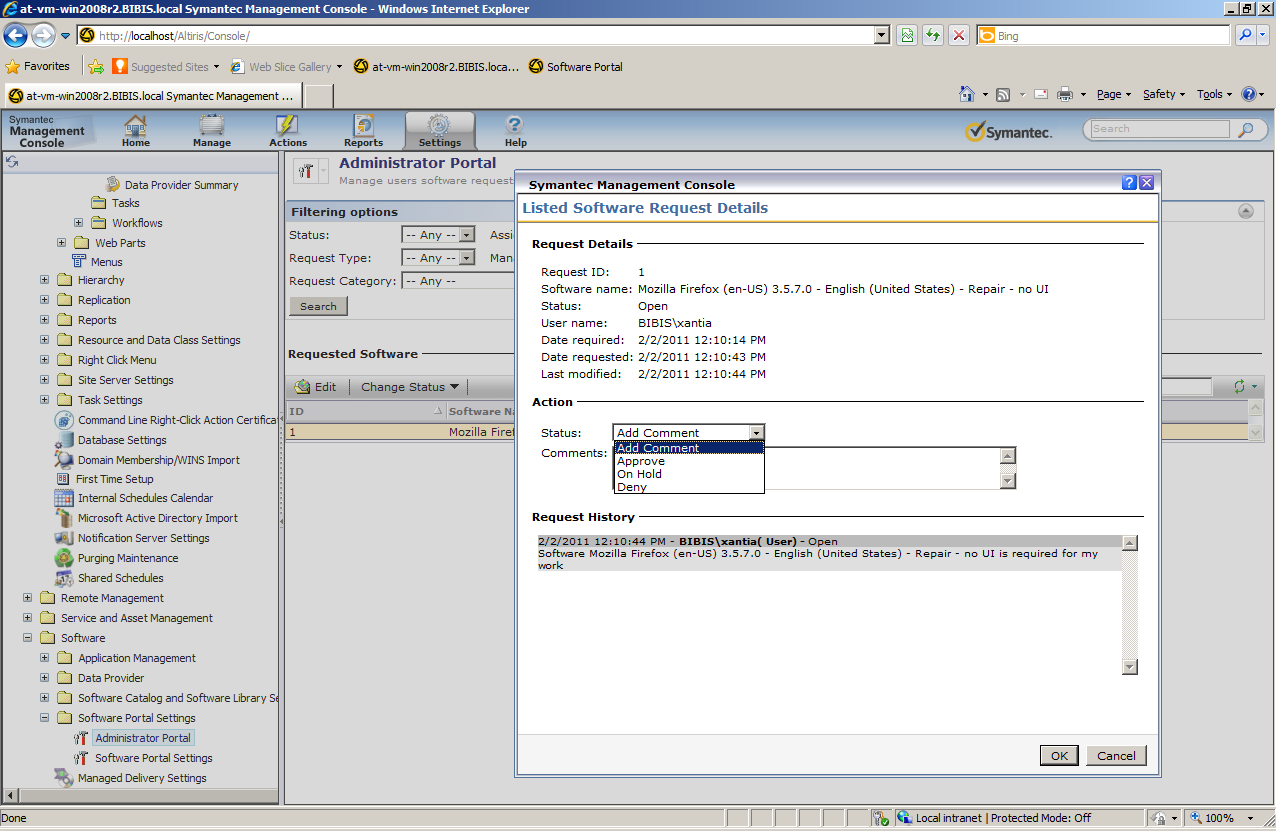
1. Get unique requesters (map of Resource User Guid to Resource User Name) from all requests. This is done in the Code and is not available via SQL or other visible methods.
2. Next the SWP gets ALL Managers if one is selected in the Administrator Portal, or automatically selects the Manager who is logged into the Manager Portal. Also show requests from subordinates of subordinates. The SWP uses the stored procedure **spSWP\_GetAllManagers** to accomplish this.
3. The SWP now Matches requesters that are subordinates of the Portal Manager:
   1. First, the SWP gets ALL subordinates of the Portal Manager. The stored procedure **spSWP\_GetUsersByManager** is used. This procedure returns all subordinates analysing only the Notification Server’s database. The main obstacle here is that a subordinate of the Portal Manager may be An AD or local group, but a requester is always a user. The SWP needs to know - is the requester a member of a group that is the Portal Manager's subordinate or not? For example - We have a group "G1" in domain "Dom1", and this group is a subordinate of the Portal Manager. Requester X is member of the group "G1". The Portal Manager can manage requests from user X. This is a simplest case.
   2. Second, the SWP Iterates through all the subordinates. If a subordinate is an AD user - try to find a matched requester (duplicate). If matched - remove the option “remember this requester” and remove it from the unique requesters map. If the subordinate is an AD group - remember it for future analysis.
   3. Third, the SWP Iterates through all the subordinates that are AD groups. It finds requesters that are direct or indirect members of the subordinates groups. To optimize this matching, the SWP does it in "bidirectional" and "simultaneously (asynchronously) in multithreads” (This is new in 7.0 MR4 and 7.1). Bidirectional means that the SWP first gets a map of all parent groups (including the primary group) for all users and the requester. Following that it processes the group - subordinate and find it in the requester's parent groups. It accomplishes this recursively for all child groups of the subordinate group. The algorithm has been optimized for better performance, in that it remembers all analysed groups to avoid repeating and cyclic dependencies. It stops analysing if no more requesters are found, and it analyses simultaneously in several threads. This process only uses a .NET API, with no database stored procedures in use.
   4. Lastly, the SWP does it recursively for all sub-managers that are subordinates of the Portal Manager, if applicable.
4. The next step done is to exclude requests from the initial set where the requester is not a subordinate of the Portal Manager. See Figure 14 for a graphical representation of this process.



**Figure 14 - Filter Portal Manager Subordinate**

For ease in administering Portal requests, a Change Status control allows very quick approval or disapproval. The Administrator or Manager can also use the Edit control if some of the default settings or a comment needs to be added. See Figure 15 for a screenshot of the edit window.

The change of status is saved in the **SWP\_SoftwareRequest** table. If a comment has been added, it is saved in the **SWP\_RequestComment** table as new record (see Figure 5 - Software Portal tables). If you are not using MR4 of 7.0 or later, please see the Known Issues section for a limitation when using the Change Status control.



**Figure 15 - Edit Request**

## Quick Delivery or MSD fulfilment

When a Software Request is approved automatically and a user requests such software, or if an Administrator of Manager approves a request, a new Client Software Delivery Task is created. This is for listed software published directly from the Software Resource. For a Managed Software Delivery Policy, the Policy itself is used. The following two processes detail how Portal requests, via Listed Software or an MSD Policy respectively, are handled by the Software Portal on the Notification Server.

### Listed Software

The Software Portal uses a Manager class to process any incoming requests for Software. The logic used by this process is detailed here:

1. The SWP obtains the Software Request. This can contain multiple approvals if the Request required approval and the approval was executed against multiple requests for the same software.
2. Next, the SWP gets a Publishing Item associated with the selected Request Item. The association are unique because a request item references to one publishing item (see Figure 5).
3. Once the Publishing Item is obtained, the SWp gets ALL Task GUIDs for the Publishing Item. To do this, it uses the stored procedure **spSWP\_GetTaskGuidByPublishingUtemGuid**. This Task GUID is stored in the **SWP\_SoftwareRequest** table, TaskGuid field (see Figure 5 - Software Portal tables).
4. The SWP looks at the resulting list of tasks and tries to find a Task with the same command line and package as the listed software just requested.
5. The SWP will create a new Task if an existing one is not found. The Task type is Altiris.SoftwareManagement.Tasks.DeliverySoftwareEx. The parent folder for these tasks is defined in Folder.cofig XML (Software Management Solution). The folder name is "Software Request Item Folder", the GUID is "0C682F39-25BF-4b57-99A3-0C0116D933FC", and the attributes are “no delete” and “hidden”.
6. The SWP then either Schedules the existing task, or schedules the new task. The scheduling is done via Task Server, using both TaskSchedule and TaskExecutionHelper classes of the Task Server .NET assemblies. The schedule time is set at now or, if the user set a schedule in their request, the user’s schedule time. Computers for this task are taken from the Request Item’s Machine GUID or GUIDs.
7. If the Administrator sets the listed software as already approved, when the User Requests it the Quick Delivery task will be executed using the User’s Schedule, or if none is specified the Quick Delivery Task will be executed at the Current Date time +5 minutes.
8. If there are multiple users who request the same publishing item (same resources and same command line) then for every user in the Quick Delivery Task a separate scheduled task will be created within the same QD.

### Managed Software Delivery Policy

The basic same method is called as for listed software, a Software Resource:

1. The SWP obtains the Software Request. This can contain multiple approvals if the Request required approval and the approval was executed against multiple requests for the same Managed Software Delivery Policy.
2. Next, the SWP gets a Publishing Item associated with the selected Request Item, in this case which is a MSD Policy. The associations are unique because a request item references to one publishing item (see Figure 5).
3. The SWP then gets a Resource Target using the **spSWP\_GetResourceTagetGuidForPublishing** stored procedure.
4. Regardless of what targets may already exist on the MSD Policy, the SWP will add the computers the requests were made from to the MSD Policy.
5. The Portal will then execute a Save of the MSD Policy.
6. Lastly, the SWP will Schedule an ASAP Update Client Configuration Task, set at Date time Now +5 minutes.

## Populating the Portal

Having configured a group of users to have access to certain Software, how does the Portal know what software to display when a user logs onto it? While the eventual target of the Software will be the Computer the user is logged onto, the computer makes no difference to what is populated on the Portal (assuming that the Software Management Solution Agent is properly installed, if it is not the Portal will throw a message indicating such). It is the User who populates the Portal, and the computer is only involved when the user requests Software.

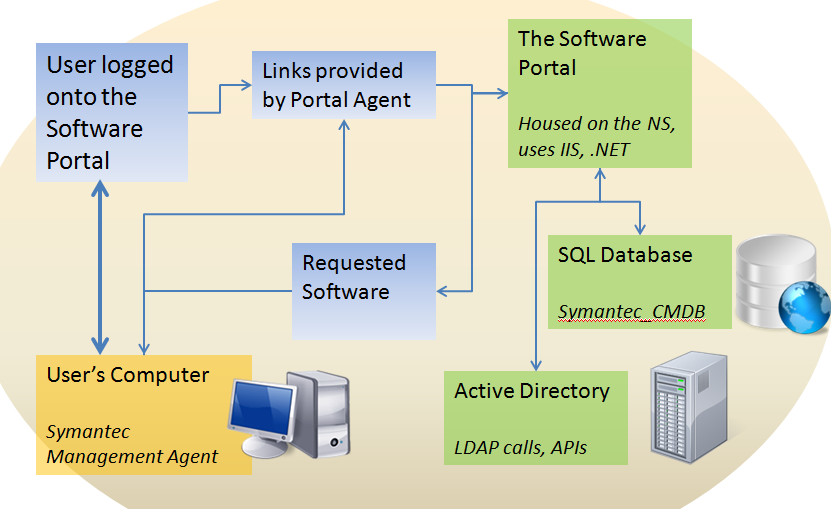
To determine what software is published and available for the user, the Software Portal must collect all the Groups that user is a member of, either directly or indirectly. This means Groups that have nested groups within add to the complexity of this collection of data. This also applies to trusts within a multi-domain environment. To accomplish this, the SWP collects the Security IDs that the user is a part of.

Once all the SIDs are gathered, those SIDs are compared against all Software Resource, and thus command lines thereof, and Managed Software Delivery Policies to see which ones the user has access to. Since there are two possible options for each PublishedItem, it is possible that a User, through different Group assignments, will have more than one return per PublishedItem. Specifically this is when, through one group, the user has rights to request software that requires approval, and through another group the user has rights to request the software immediately, as in already approved.

It is the SIDs of the Groups and the User that makes all the connections via the Symantec Database.

This analysis of user Groups is done directly in .NET on the Notification Server. The SWP, which is housed on the SMP, calls an Active Directory Service API to fetch all SIDs associated with the user. .NET also uses SQL via the stored procedure **spSWP\_GetSoftwarePublishingItem** to match all the SIDs fetched via AD to PublishingItems, or what Software that user can request.

See Figure 16 for a diagram representing the Portal Request Process.



*TechTip:* ***GetRemoteUserMembership*** *is the primary method on the ContextUtil class that is used to get the SIDs for the user accessing the Software Portal.*

The process or analysis of obtaining the SIDs, and subsequently the PublishedItems, uses the following Algorithm:

1. The Software Portal obtains the Active Directory entry of the user accessing the Portal.
2. If the User is local, then the SWP analyses only the local context. .For the .NET API this means that the name of the local machine is used for the corresponding LDAP queries. The machine name is taken from the user context that is entered as the credentials when launching the Software Portal (it is possible the User is logged on as a user that is automatically passed to the SWP, so no prompt was given. In this case it’s the user’s logged on context).
3. If the user is a Domain User, The SWP must first analyse the domain relations (including relations between multiple domains), and then analyse its local context (as with step 2) because the domain user may also be a member of a local group.
4. Next, the SWP Analyses relations between groups, meaning that it obtains a SID for every group, whether local or domain, which the User belongs to.
5. The SWP executes a Parent group search that is a recursive process. First, the SWP looks for groups that the user belongs directly to, and then for each matched group it looks for groups that include this group, etc. For the purpose of optimization and to prevent the analysis of cyclic dependencies, the SWP remembers groups already analysed and does not analyse them a second time.
   1. To find all Parent groups the type of LDAP queries that are used are, for example: "Select all the groups that have <user> member"
   2. The LDAP queries the SWP executes are within a single domain.
6. There can be trusted relationships between domains. This means that the groups of one domain may be the Parent groups of another domain, and in this case trusted Domain group’s analysis are required.
   1. The first analysed domain is the user’s domain.
   2. After the user relations analysis are completed within the user’s domain, the SWP searches for all domains that have trusted relations with the current user domain.
7. Relations between groups, which are found in any trusted domains, are also analysed.
8. Trusted domain analysis is a recursive process. For optimization purposes and to prevent the analysis of cyclic dependencies, the SWP remembers relationships between all the domains.
9. The analysis process ends when all domains and groups within trusted domains are discovered and all SIDs have been obtained.

### Tables in Use

As SMP is heavily database driven, knowing some basic schema details can help when troubleshooting issues. The following tips stem from the question: What tables are used to determine what software is available?

A method exists (by name: SearchSoftwarePublishingItems) that uses 3 parameters in order to obtain Software for a User who has loaded the Software Portal.

1. SearchValue = what to search (by default it is %)
2. OnlyRecommended = search for only recommended software, toggled by the Recommended button on the Portal
3. TrusteeSids = search for software for current SIDS only

This method calls the stored procedure spSWP\_GetSoftwarePublishingItem, which obtains publishing items for the current SIDs.

This stored procedure uses the view vSWP\_PublishingItem and the SWP\_PublishingItemSetting table to determine which software is available.

## Known Issues

The 7.1 Portal has been heavily optimized based on feedback learned during the use of 7.0.

For the 7.0 SP2 MR4 release, many of the issues we encountered originally are resolved. At the time of this article, there is only one item that is under investigation. This involves untrusted domains, where the user has rights in the user’s domains, but other domains in the structure are not trusted. This causes the Portal to fail for that user.

Issues for MR3/MR2:

1. Before MR4 an issue existed when approving multiple requests with the same publishing item. In this case, if the administrator or manager approves more than 1 request (multiple selections) from different clients who wants same software, the Quick Delivery task for every user will run more than 1 time.
2. Poor Portal Performance – Many optimizations were made in MR4. MR3 contains some performance optimizations, but the real heavy hitters were released with MR4.

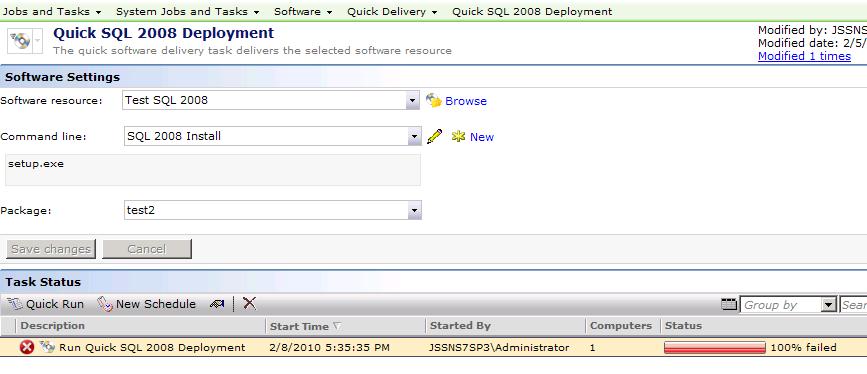
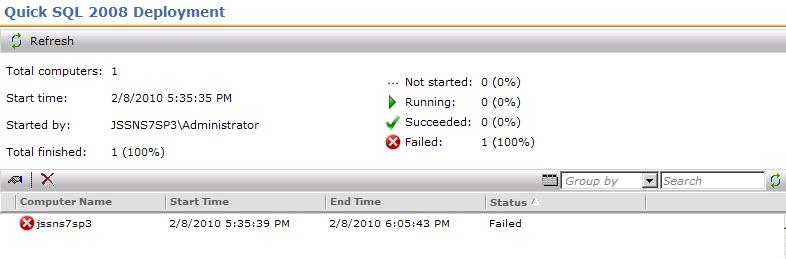
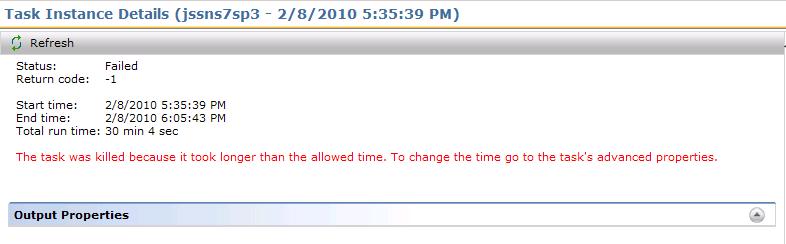
# Tracking Deployments

After you’ve setup your Quick or Managed Deliveries, now you need the ability to track the deployments. This can be accomplished in two basic ways: Task Server Status grids, and the Software Management Reports. Both methods provide insight into the degree of success for a quick deployment or a Managed deployment.

## Task Server Statuses

The status grid and drilldowns provided by Task Server are not available for Managed Delivery. The main reason for this is that the Tasks executed during a Managed Delivery are not initiated from the server, but are client-initiated tasks. This leaves Quick Delivery as the consumer of this feature.

To demonstrate how this works, I’ll use the test deployment of SQL 2008 previously configured. In this example I want to check to see how the Quick Delivery Task operated.

1. In the Symantec Management Console, browse under Manage > Jobs and Tasks.
2. Browse down through Jobs and Tasks > System Jobs and Tasks > Software > Quick Delivery.
3. When you select your Task, in this example SQL 2008, you’ll receive the following screen:  
   
4. As seen, this deployment failed.
5. Double-click on the scheduled instances to drill down into this instance.
6. This gives you a more in-depth view of statuses. If this rollout had multiple target computers, you’d have the ability to see status per resource. See this screenshot for an example:  
   
7. To receive all status information for a particular execution attempt, drill down by double-clicking on the row desired.
8. The following screenshot demonstrates this screen:  
   
9. As you can see, the Task was terminated based on the execution time exceeding the Task timeout value set under the Advanced Settings.

For tasks, this grid system allows you to view the status at multiple layers, with greater granularity the further you drill down.

## Provided Reports

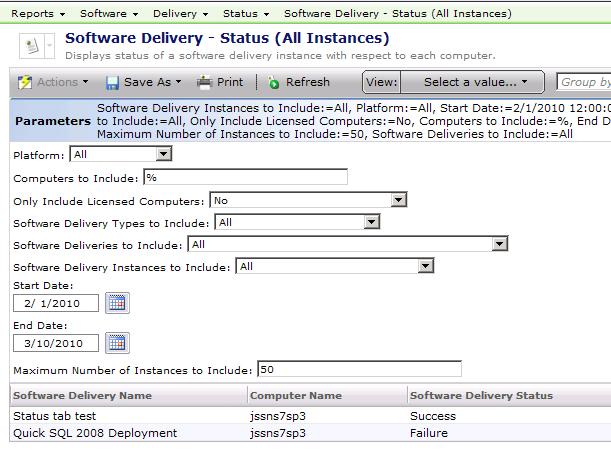
Reports offer flexibility in how you want to view the results of a managed or quick deployment. The following reports are available for use in tracking deployments.

For Task Server Quick Delivery reports, they are found in the Symantec Management Console under Reports > All Reports > Software > Delivery (some reports contained in the root) > and Status.

*NOTE: These are for Quick Delivery Tasks only:*

* Software Delivery – Download Status
* Software Delivery – Download Summary
* Software Delivery – Execution Attempts
* Software Delivery – Execution Failures
* Software Delivery – Execution Status
* Software Delivery – Execution Summary
* Status > Quick Delivery Details by Task
* Status > Quick Delivery Details by Computer
* Status > Software Delivery – Status (All Instances)
* Status > Software Delivery – Summary (All Executions)

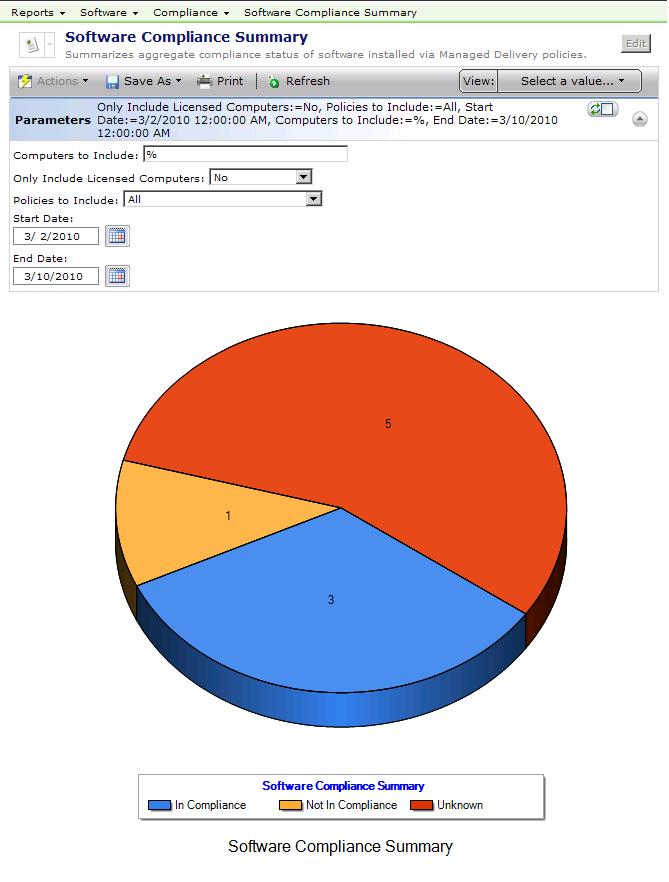
Obviously some reports are better than others for giving you an idea how compliant your target systems are. For my example I will use the Software Delivery – Execution Status in tracking the previous SQL rollout.

1. Load the report Status > Software Delivery – Status (All Instances).
2. Change the date range to start at the date you initially configured and rolled out your task.
3. If necessary, choose a specific Task to track. In my example I only had 2 and left the field as All.
4. Click Refresh.
5. See the following screenshot for the results:  
   
6. Manipulate the parameters as needed to tune your results. Most of the reports listed above have the same parameters.

For Managed Software Delivery Policies, the Reports are found in the Symantec Management Console under Reports > All Reports > Software > Compliance. The following reports are available:

* Software Compliance by Computer
* Software Compliance by Managed Delivery Policy
* Software Compliance Detailed Summary
* Software Compliance Details by Computer
* Software Compliance Details by Managed Delivery Policy
* Software Compliance Remediation Summary
* Software Compliance Status
* Software Compliance Summary

These reports fit the nature of a Managed Delivery Policy. For example browse to the report: Software Compliance Summary. I’ve included all MSDs I’ve deployed in the example shown in the screenshot below:



Other reports provide more specific details for any given MSD. Review them all to find the one that gives you the best representation of the data you require.

# Conclusion

I hope this best practices article helps ease you into the use of Software Management on the Symantec Management Platform 7.0. As new versions are released, some of this information may become out of date. Keep this in mind if you come across a screen you see or a step that doesn’t fit. I don’t anticipate large changes between now and even the 7.1 release.