MECM - SCCM - Application Creation

MECM - SCCM - Application Creation

On this page:

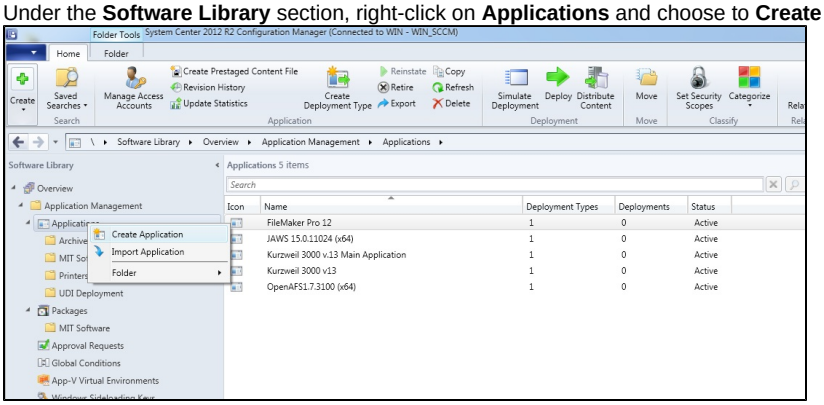
Creating an application from an MSI based installer
Creating an application from an EXE based Installer
Determining Uninstall Program
Testing procedures
See Also

Here we will outline the process for taking a MSI or EXE based installer and creating an application for deployment. The application based model in MECM is stateful. There is a client side evaluation of the installed state of the application which is reported back to the MECM server. The application is then made available or mandatorily installed based on the state of the client machine.

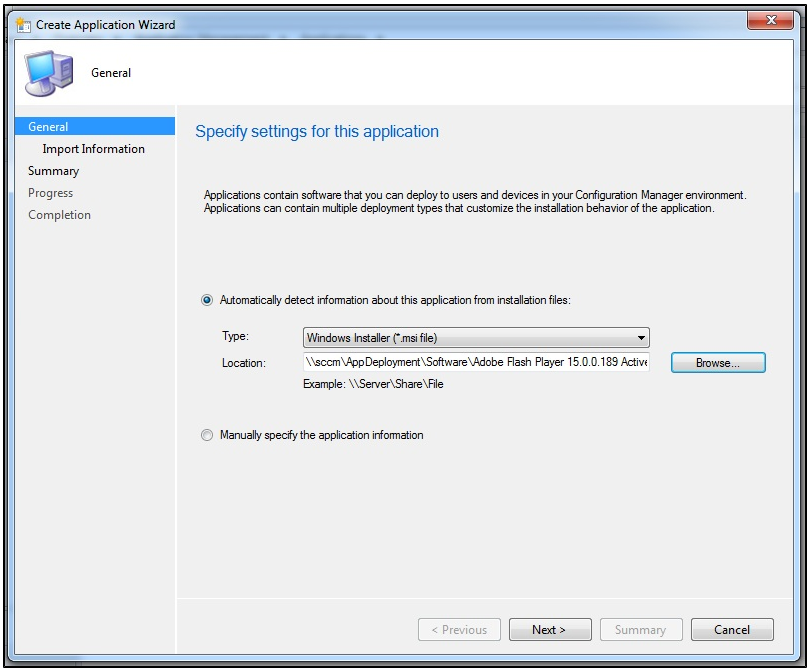
On this page:

Creating an application from an MSI based installer
Creating an application from an EXE based Installer
Determining Uninstall Program
Testing procedures
See Also

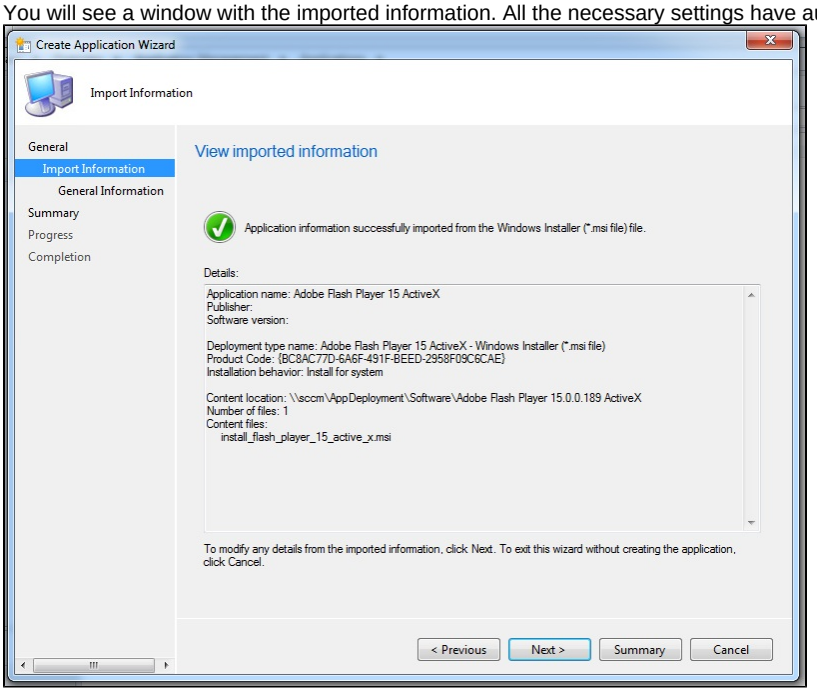
Creating an application from an MSI based installer

1. Under the **Software Library** section, right-click on **Applications** and choose to **Create Application**.


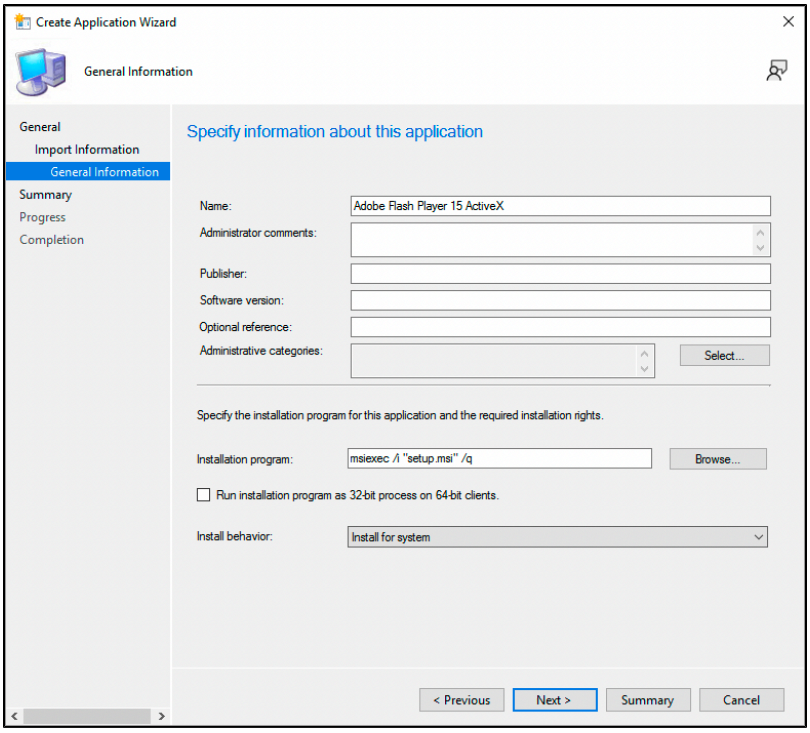
2. Browse to the msi file and select it. You should store the file in \\sprockets\uploads\yourfoldername\applicationfolder (each application should be in its own folder).



3. You will see a window with the imported information. All the necessary settings have automatically been configured.



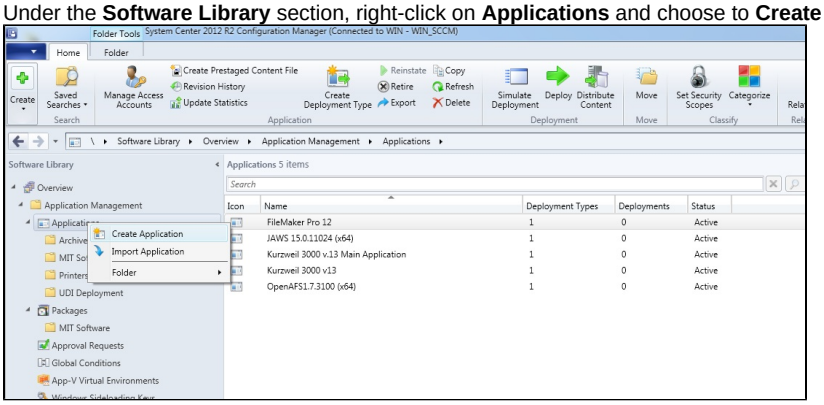
4. Click Next. You may optionally add additional information, or accept the defaults. By default, it will install for all users.



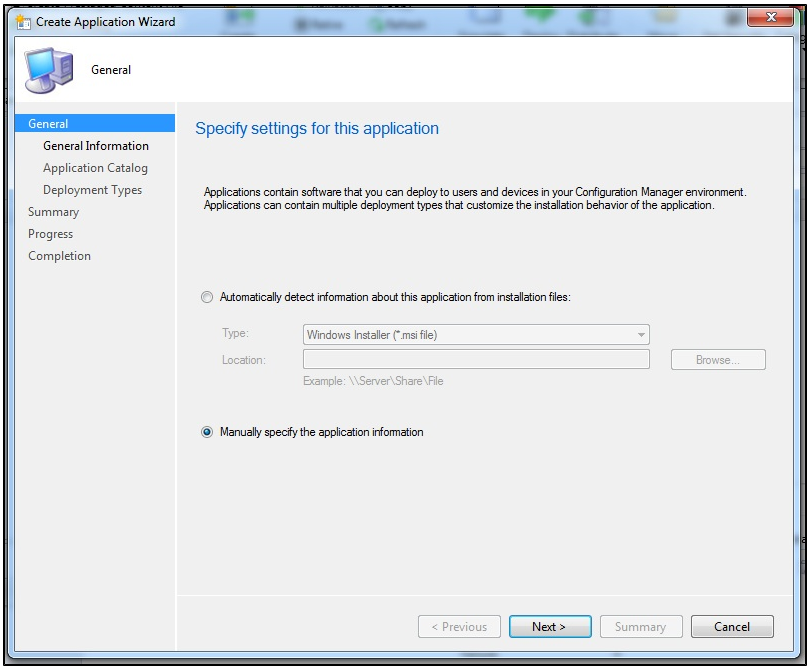
Creating an application from an EXE based Installer

Using an EXE based installer requires some manual configuration. This example will use the common freeware application Notepad++.

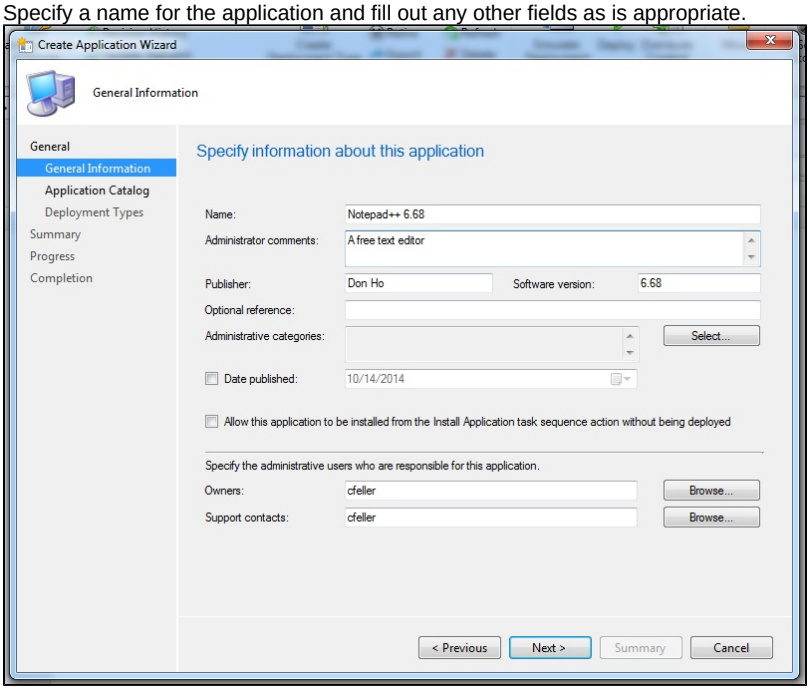
1. Under the **Software Library** section, right-click on **Applications** and choose to **Create Application**.



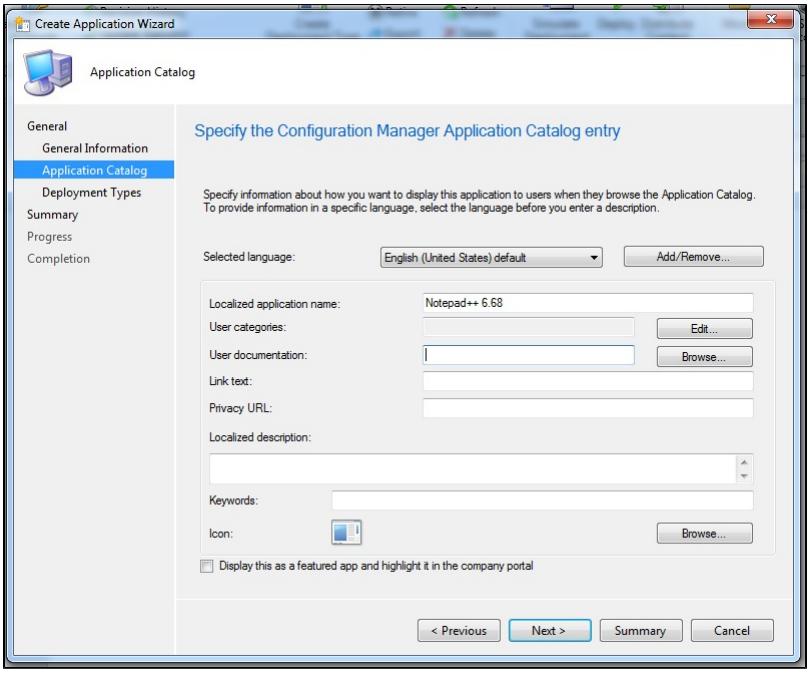
2. Select **Manually specify the application information** and click Next.



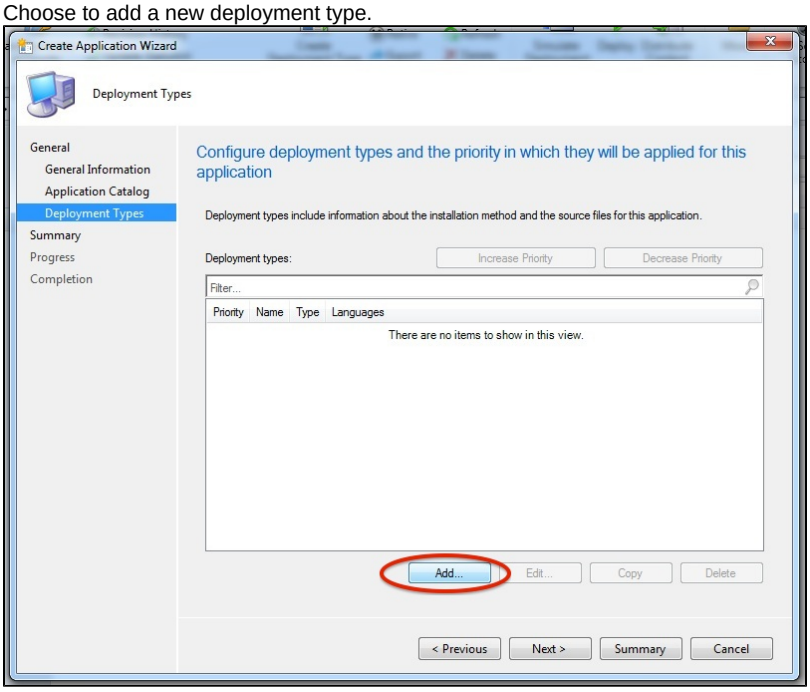
3. Specify a name for the application and fill out any other fields as is appropriate.



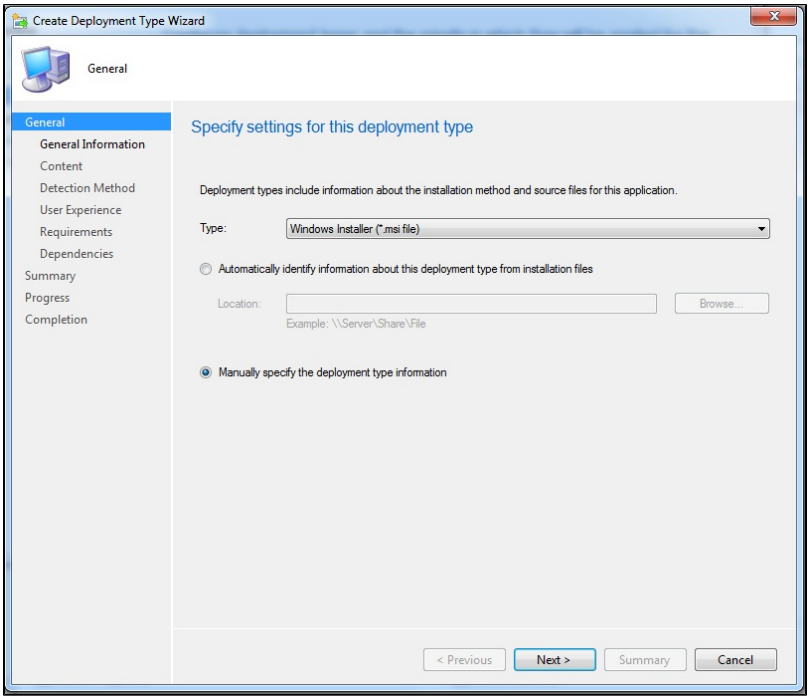
4. The next window lets you specify the attributes for the application catalog. You can leave the default settings for now.



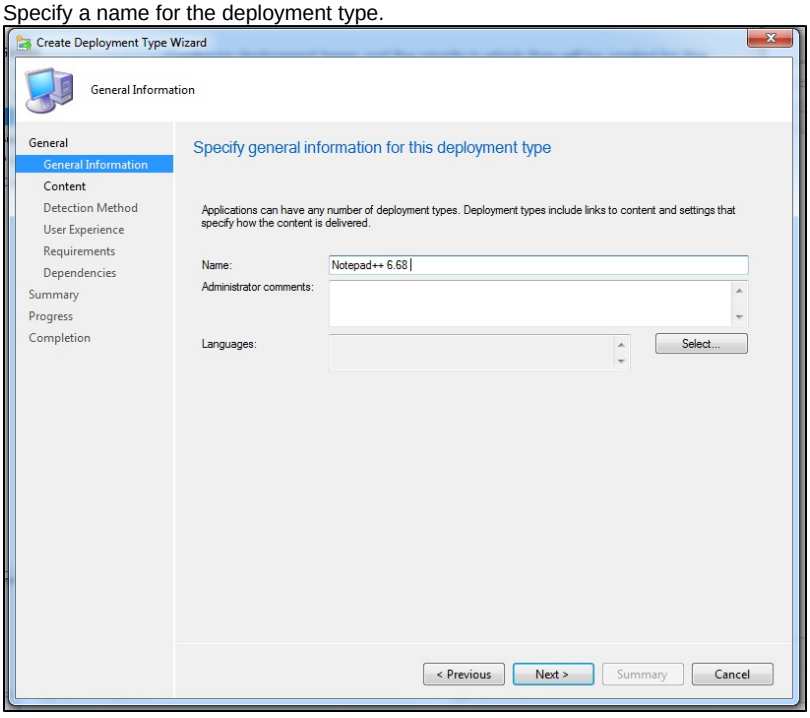
5. Choose to add a new deployment type.



6. Choose to manually specify the new deployment type.

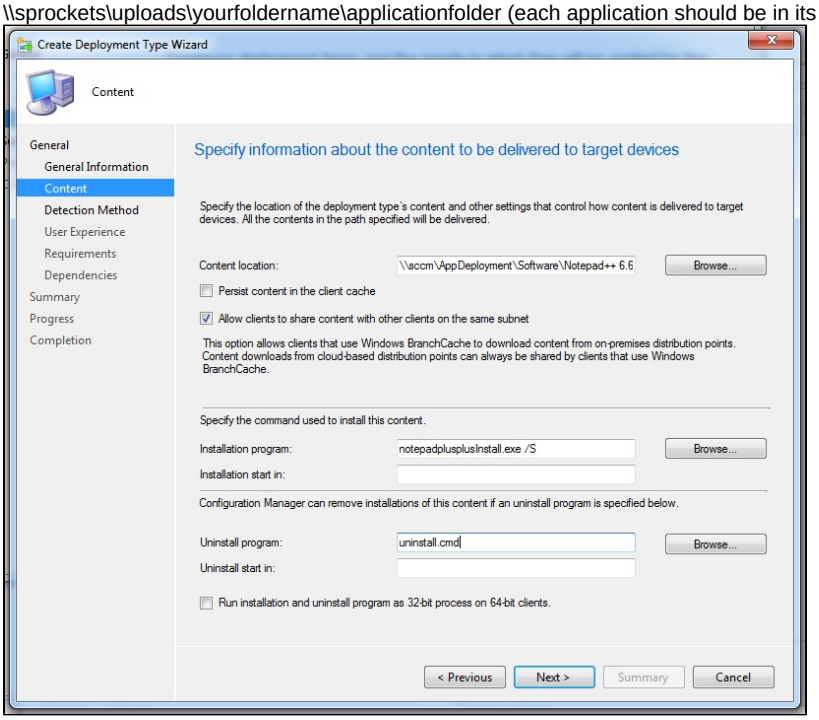


7. Specify a name for the deployment type.

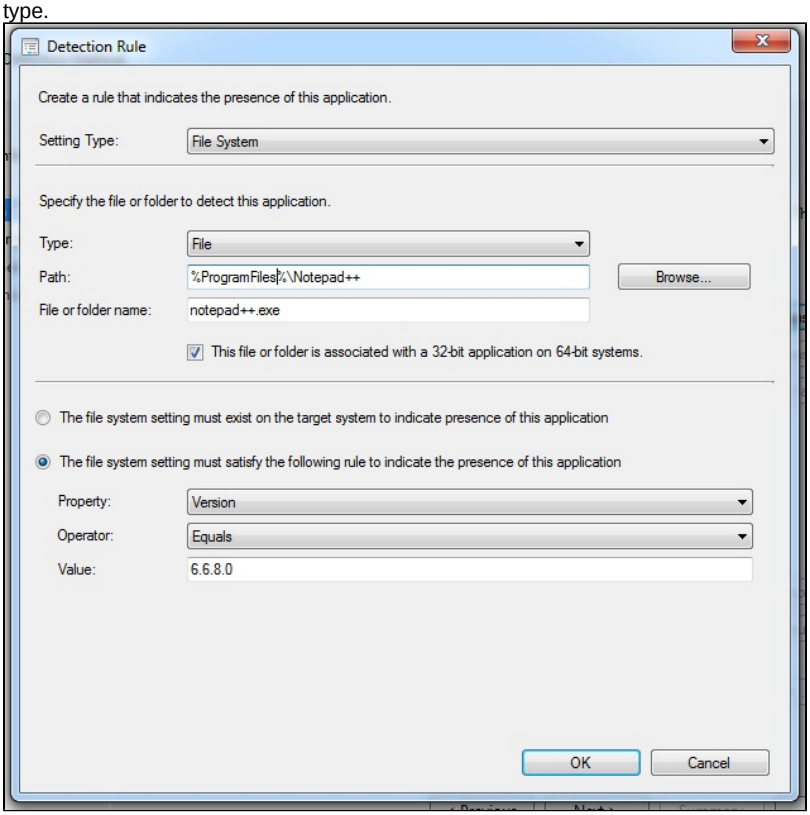


8. Choose to browse to the to Content Location. This is where you're source files are located and should be a UNC path. Since this is not an MSI based install, you'll need to manually specify the setup program e.g. "setup.exe." You should store the file in

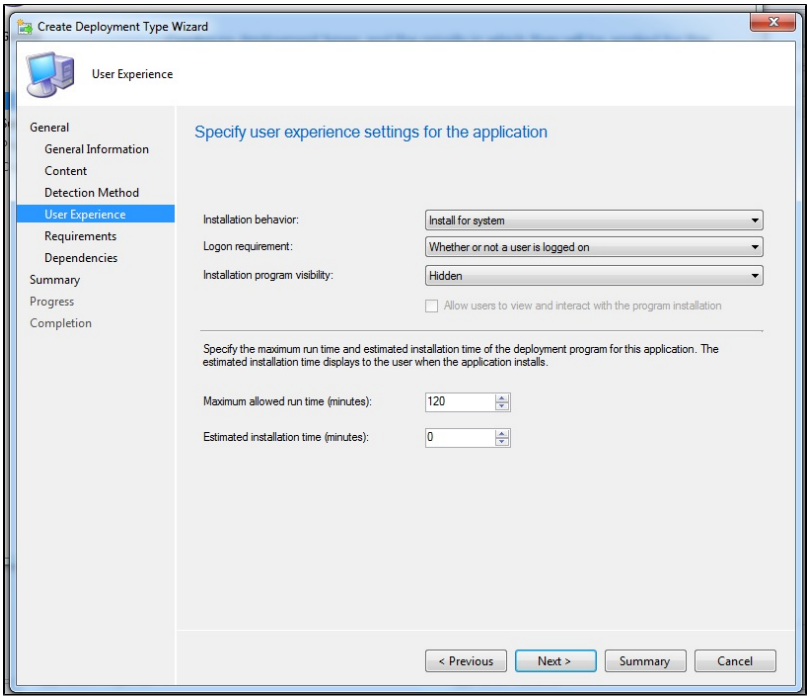
\\sprockets\uploads\yourfoldername\applicationfolder (each application should be in its own folder).



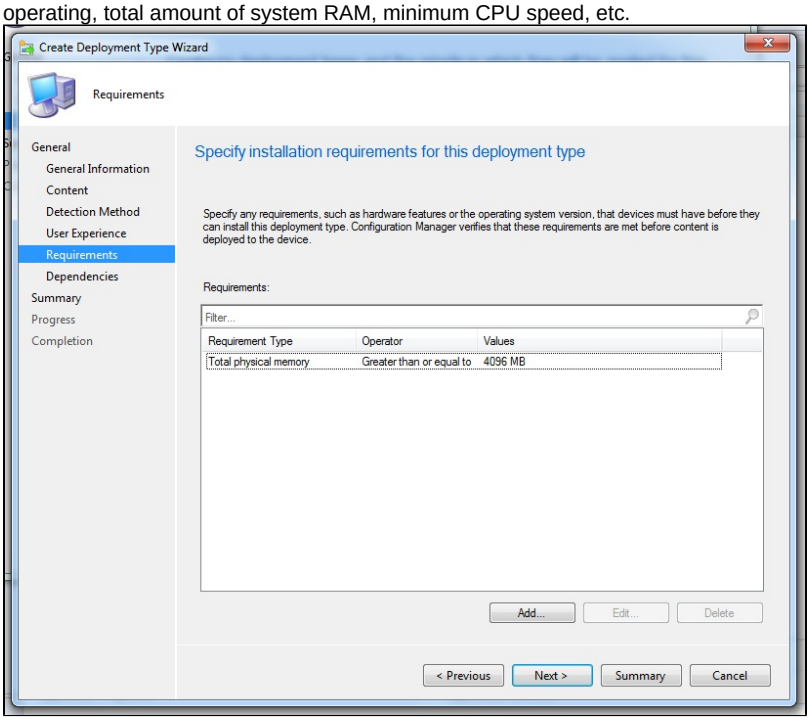
9. You'll need to specify a means to detect whether the application is installed. This can be in the form of an MSI product code (some exe installers are wrappers for MSI), a registry entry, or file located on the system. This example will use a file based detection rule. Note: The option "**This file or folder is associated with a 32-bit application on 64-bit systems"** maybe be necessary for 32-bit applications. This is because 32-bit applications will install to the "C:\Programs Files (x86)" on 64-bit systems instead of using "C:\Program Files". Using this setting along with the **%Program Files%** environment variable will ensure that the correct path will be selected, regardless of architecture type.



10. Next you'll need to select user experience settings. Typically it's recommended to select **Install for system**, **Whether or not user is logged on**, and **Hidden**. This will install the application for all users and in a silent manner.



11. You may choose to select certain requirements of the client on the next window. For example, you may want to specify a certain operating, total amount of system RAM, minimum CPU speed, etc.



12. You can choose to select software dependencies to install if necessary. The final window is a summary of all the settings for this deployment type. Click **Next** when you're done and then click **Close**.
13. You'll be back at the deployment types windows in your Application Wizard. Click **Next** to continue and then click **Next** at the summary. Click **Close** when you're finished.

Determining Uninstall Program

Each application may have a different method for uninstalling the application. As mentioned earlier, some exe setups are actually wrappers for MSI installers. The MSI uninstall strings can be found in the registry:

HKEY_LOCAL_MACHINE \ SOFTWARE \ Microsoft \ Windows \ CurrentVersion \ Uninstall {application GUID}

You should be able to figure out the application by looking at the DisplayName for each GUID. When you find the correct GUID, the correct uninstall string is "msiexec /x ??{GUID} /q"

For other exe based installers, you may need to check with the developer to determine the correct method for uninstalling the application via command line.

Testing procedures

The following scenarios are possible with a rollout of software and need to be tested for ahead of time based on your department setup.

- Test when application being deployed is not installed on machine (clean install).
- Test when application is being upgraded on machine.
- Test when application is already in use (worst case scenario).

After installation, test the following:

- Did the machine reboot? It shouldn't. If a reboot is required, considering putting it in the software center.
- Does the application open and work as intended? Try opening / saving some kind of document depending on program.

See Also

- Microsoft Endpoint Configuration Manager (MECM) Landing Page