

SCCM Application Management Overview

Deploying and managing software is one of the most important things that you will do as a SCCM administrator. It is important that you setup your lab and practice the examples that will be given in this section.

So let's get started. There are two methods to create software in Configuration Manager

- Packages
- Applications
- Packages are the older method of creating software.
- Applications are software that fully support the newer SCCM Application model.

Advantages of Applications over Packages.



- Applications – Have the ability to Supersede old software, or automatically install prerequisites. So what do you mean by supersede? For example, if you have deployed version 1604 7-zip and there is a newer version available. You can configure a new application to either uninstall the older version or just update the current version to the latest version. This is called Supersedence.
- The easiest software to deploy is the MSI version of the software, which stands for Microsoft Installer. The MSI version of software is optimized for the Application Model.
- Packages – Still have their uses, but packages can't handle all the new functionality of the Application model. For example, packages have no supersede function.
- You can create packages from MSI's or EXE's
- MSI's automatically detect a lot of the installation information right from the install file. Such as the install and uninstall command line. Whereas for the EXE, you will need to get the command line information from the vendor or other sources.

Creating Applications


- **First, you'll want to create your source folder**


Create a folder for each version of the software you intend to deploy

In our example, I have created a folder for version 1604 and another folder for version 1900. That way, you avoid confusion, and you have lessened the chances of making a mistake and deploying the wrong version to the client.

<< 09_Application Management > Software > 7-zip_msi	
Name	Date modified
 7z 1604	5/14/2020 2:15 PM
 7z 1900	5/14/2020 2:15 PM

- You can take this one step further if you plan on deploying software that is 32 and 64 bit, by creating separate folders for those as well.

<< Software > 7-zip_msi > 7z 1604 > 7z 1604 x32	
Name	Date modified
 7z1604 x32.msi	5/13/2020 5:41 PM

<< Software > 7-zip_msi > 7z 1604 > 7z 1604 x64	
Name	Date modified
 7z1604-x64.msi	5/13/2020 5:20 PM

- **Creating a shared folder**

After you've created your source folder structure in order to create applications, we will need to share the folder called software. In this example from C: on the server right click on the software folder, then click properties, then click sharing then click advanced sharing then click share this folder then click permissions then click full control then click OK then click OK again then click close. Now when creating applications, you'll have access to your software folder.

- In this section, we will be discussing the different components like
 - Deployment types
 - Requirements
 - Detection Methods
 - Dependencies
 - Supersedence
 - Task sequence options
 - Install and uninstall command lines and parameters

Instead of just telling you about each of these components. I would rather take an explain as we go approach. The best way to understand the use of these components is by creating Applications. So, in the next lecture, we'll start with the basics. I'll show you how to create an application from an MSI using 32-bit and 64-bit versions of the software.

So, I'll see you in the next lecture.