

**Microsoft Application Virtualization 4.5**

**Sequencing Guide**

White Paper Descriptor

This whitepaper is designed to provide administrators with guidance for sequencing applications to create virtual packages that can be delivered to the end user. This document discusses setting up the sequencer, sequencing best practices, an example of sequencing, important information related to updating packages, and finally examples of advanced OSD scripting.

Copyright © 2008 MICROSOFT CORPORATION

Contents

[1 Introduction to Application Sequencing 5](#_Toc208720215)

[2 Sequencer Workstation Configuration 6](#_Toc208720217)

[3 Recommended Best Practices for Sequencing 8](#_Toc208720219)

[4 Gauging Applications for Sequencing 10](#_Toc208720220)

[5 Sequencing Limitations 11](#_Toc208720223)

[6 Sequencing Phases 12](#_Toc208720224)

[6.1 Installation Phase 12](#_Toc208720225)

[6.2 Launch Phase 13](#_Toc208720226)

[6.3 Customization Phase 14](#_Toc208720227)

[6.4 Save Phase 14](#_Toc208720228)

[7 Sample Sequencing 15](#_Toc208720229)

[7.1 Package Information 15](#_Toc208720230)

[7.2 Advanced Options 16](#_Toc208720231)

[7.3 Monitor Installation 17](#_Toc208720232)

[7.4 Adding Files 18](#_Toc208720233)

[7.1 Configure Applications 18](#_Toc208720234)

[7.2 Launch Applications 19](#_Toc208720235)

[7.3 Package Deployment 20](#_Toc208720236)

[7.4 Saving the package 21](#_Toc208720237)

[8 Application Package Upgrade 22](#_Toc208720238)

[8.1 Updating a Package that will replace the existing package 22](#_Toc208720239)

[8.2 Updating a Package that will run at the same time as the existing package 23](#_Toc208720240)

[9 Dynamic Suite Composition 24](#_Toc208720241)

[10 Compression of the SFT 25](#_Toc208720242)

[11 Advanced OSD Scripting 25](#_Toc208720243)

[11.1 <SCRIPTBODY> Examples 27](#_Toc208720244)

[11.2 <HREF> Examples 28](#_Toc208720245)

[11.3 <PARAMETERS> Examples 28](#_Toc208720246)

[11.4 <WORKINGDIR> Examples – Specifying a Working Directory 29](#_Toc208720247)

[11.5 <REGISTRY> Examples – Registry Customizations 30](#_Toc208720248)

[11.6 Launching Local Applications 30](#_Toc208720249)

[11.7 Multiple Operating System Support 30](#_Toc208720250)

[11.8 Launching Application via Non-Default Provider Policy 31](#_Toc208720251)

[11.9 Running an Application as a CONSOLE application 31](#_Toc208720252)

[11.10 OSD XML Syntax Check 31](#_Toc208720253)

[12 Advanced Sequencing Techniques 33](#_Toc208720254)

[12.1 Sequencing Applications That Cannot Install to Q:\ 33](#_Toc208720255)

[12.2 Sequencing Web Based Applications 34](#_Toc208720261)

[12.3 Sequencing applications that require access to local devices and or resources 34](#_Toc208720262)

[12.4 Sequencer Log Files 34](#_Toc208720263)

[13 Finding Additional Information 36](#_Toc208720264)

[13.1 Error Codes and the Microsoft Knowledge Base 36](#_Toc208720265)

[13.2 App-V Team Blog 37](#_Toc208720266)

[13.3 App-V Related Technical Discussion Forums and Web Sites 37](#_Toc208720267)

[Appendix A 38](#_Toc208720268)

[Appendix B 46](#_Toc208720269)

1. Introduction to Application Sequencing

The Microsoft Application Virtualization (or App-V) Sequencer is a component of the App-V suite used to package your applications to be deployed to systems using the App-V client. Properly sequencing applications is the key to a successful App-V implementation. As such, it’s important to follow Microsoft’s recommended practices and be aware of the different options when sequencing. This document covers setting up the sequencer, sequencing best practices, an example of sequencing, important information related to updating packages, and finally examples of advanced OSD scripting.

1. Sequencer Workstation Configuration

Proper configuration of the sequencing station is imperative to ensure that applications will function properly when streamed to a client. Microsoft recommends the following configuration when sequencing:

* **Sequence on a machine that matches the Operating System and configuration level for the target clients**. Microsoft has clarified its support stance since 4.2. Sequencing on Windows XP and deploying to Windows Vista is not a supported scenario. If you choose to sequence on one Operating System and deploy to another then you do so at your own risk. In addition to the Operating System, you want to make sure your sequencer is at the same service pack and hot fix level of your deployed workstations.
* **If Microsoft Office is part of the base image of the client, then include it as part of the base image of the sequencer.** Many applications will install differently if they recognize that Microsoft Office is already installed on the machine. Thus, if an application is expected to integrate with Microsoft Office, it’s best to attempt sequencing on a machine with Office already installed and activated. This assumes that a Microsoft Office suite will be installed locally on all client PCs. In addition you may want to install any other programs that could be used by the application you are sequencing if they are not going to be a part of the sequence.
* **Create an ODBC DSN setting as part of the Sequencer base image.** If no ODBC DSN setting exists on the base Sequencer image and the application being packaged creates one, the entire registry key associated with ODBC settings will become virtualized. This will prohibit the packaged application from seeing any ODBC DSN settings that exist on the base client machine. If an ODBC entry already exists on the Sequencer machine, only the ODBC settings will become virtualized, and the ODBC settings on the Client will be merged with the ODBC settings in the package.
  + The following locations can be checked to determine ODBC information was captured:
    - Search for odbc.ini: It will be located in the VFS\%CSIDL\_WINDOWS% folder
    - HKLM\Software\ODBC\ODBC.INI\ODBC Data Sources
    - HKCU\%SFT\_SID%\Software\ODBC\ODBC.INI

For step by step instructions on how to setup a dummy ODBC entry see Appendix A.

* **Add a dummy printer device as part of the Sequencer base image**. Printers act in the same manner as ODBC settings. It is necessary to include a dummy printer device in the sequencer PC image. For step by step instruction on how to create a dummy printer device refer to Appendix B.
* **Setup your sequencer machine with multiple partitions.** It is recommended that the sequencer machine be configured with at least two primary partitions. The first partition C:, should have the operating system installed and should be formatted as NTFS. The second partition Q:, is used as the destination path for the application installation and should also be formatted as NTFS.
* **Temp Directory.** The sequencer uses the %TMP%, %TEMP%, and its own Scratch directory for temporary files. These locations should contain free disk space equivalent to the estimated installation size. The scratch directory is where the sequencer will temporarily store files generated during the sequencing process. You can check the location of the Scratch directory by launching the sequencer, clicking Options from the Tools menu, clicking the Paths tab, and then noting the Scratch Directory box. Placing the temp directories and the scratch directory on different hard drive spindles can improve performance during sequencing.
* **Sequence using Virtual PC.** Most applications will be sequenced more than once. This may be due to additional configuration changes or simply starting over to correct a mistake. The point is that you will be going back to your original configuration on the PC several times. To help facilitate this you may want to use a Virtual Machine. This will let you sequence an application and with a simple click of a button revert back to a clean state so you can continue sequencing with no down time. Additionally whenever you start a new sequence you will want to do so on a clean system.
* **Shutdown Other Programs.** Processes and scheduled tasks that normally run on your computer can slow down the sequencing process and cause irrelevant data to be gathered during sequencing. These programs should be shutdown before you begin sequencing. Some of these programs include:
  + - Windows Defender
    - Antivirus Software
    - Disk defragmentation software
    - Windows Search
    - Microsoft update
    - Any open Windows Explorer session
* Note: The sequencer workstation should be fully scanned for viruses and malware and then the anti-virus and anti-malware software should be disabled before creating a snapshot image of the sequencer workstation.

1. Recommended Best Practices for Sequencing

This section covers Microsoft’s recommended best practices for sequencing applications.

* It is recommended that you familiarize yourself with the installation and execution of the application prior to sequencing. Be sure to read all installation instructions associated with the application. It is also recommended that you learn how the application runs and the components of the application the user will need. To improve the process of sequencing an application, one should document step by step the installation and post-configuration procedures for the application. Step-by-step documentation will ensure that no unnecessary troubleshooting occurs during the sequencing process since no important steps will be skipped. Items to document include:
* What application components are needed and will be required to complete the installation of the application?
* What updates such as adding new files to the package need to be performed in the sequencer after the installation?
* What post-installation configuration steps need to take place in the sequencer?
* How do users commonly use this application immediately after its launch?
* Does this application do something that App-V currently does not support? If so, check the Microsoft Knowledge Base to see if there is a workaround available.
* When sequencing on Windows Vista ensure sure you have UAC enabled on the sequencing machine if the client machine you are deploying the application to will have UAC enabled as well.
* Always document the sequencing process step-by-step creating a “recipe” using a standardized template. Documenting the sequencing process step-by-step will allow you to hand the recipe to someone else in your organization and have them recreate the same package. Step-by-step documentation will ensure that no unnecessary troubleshooting occurs since no important steps will be skipped.
* Use the Comments field in the sequencer (Abstract Tag) any details about the package you may want to include. This will allow you to revisit the sequence later and have a record of this information.
* The installation drive on your sequencer should match the virtual drive on the client. In some cases this may not be possible (if say, packaging for multiple organizations), and you may be required to go back and edit the virtual registry or some individual files to point to the correct locations.
* Sequence to a unique, 8.3 directory name. This applies to both the Asset and Installation directories. (‘Q:\MYAPP’ is correct, ‘Q:\My Application’ is incorrect. Q:\MYAPP.001 is also correct.)
* Sequence to a folder in the root of the drive, not to a subdirectory. (‘Q:\MYAPP’ is correct; ‘Q:\’ is incorrect; ‘Q:\Temp\_Junk\MYFOLD’ is incorrect). If the suite has multiple parts, install each application in a subdirectory of the Asset Directory. For example, if a package contains a Line of Business Application along with the Oracle Client, use Q:\AppSuite as the Asset Directory; sequence the LOB application to Q:\AppSuite\LOB; and sequence the Oracle Client to Q:\AppSuite\OracleClient.
* Always use globally unique paths and Package names across the set of application sequencings. Do not install, for example, multiple Microsoft Office sequencings to the same Asset Directory name. Use a standardized naming scheme for the Asset Directory that can be incremented for new revisions, for example Q:\OFFXP.v1 or Q:\OFFXP.001. Failure to make these unique cause conflicts with your applications
* Configure and test the application in the Installation Phase. Completing the installation of an application often times requires performing several manual steps that are not part of the application installation process. These steps can involve configuring a connection to a ‘back-end’ database, copying updated files, etc. Do this configuration in the Installation Phase and run the application to make sure it works.
* Execute the application, multiple times if necessary, until the program is in a static state in the Installation Phase. For example, run the application multiple times to get past all registration and dialog box requests. Some applications perform different tasks on first launch, second launch, and sometimes subsequent launches. The multiple launches will make sure only the relevant application code into Feature Block 1 during the execution phase.
* Use the Application Wizard to launch each executable in a suite of applications. This will ensure that each application will have the required initial launch data on the App-V Client.
* Disable “Install on First Use”. Some applications have the option to “Install on First Use” for certain components. It is required that none of the components are sequenced with this option. It is necessary to choose either “Run from My Computer” (install this component) or “Not Available” (do not install this component). For application components that will not be used by any of the targeted users it is recommended that the components not be installed.
* Disable “Auto Update” features. Some applications have the ability to check a web site or a server for the latest application updates. This feature should be turned off, as version control should be performed via sequencing new versions.
* Operations made during the step six will be included in Feature Block 1. As a general rule when building feature block 1 make sure you execute the applications most common operations so that they are included in the initial streaming of the application and you have an accurate feature block 1. If this is not done then users will see delays as they start to use the application and will regard it as being slow if many of the features they use are not in feature block 1. Additionally if you are in an environment where bandwidth is limited then you want to have an accurate feature block 1 so that users are not constantly making calls to the server to download additional files in cache.
* There are several online articles relating to sequencing best practices you should examine as well:
  + Microsoft Support: <http://support.microsoft.com/kb/932137>
  + App-V Team Blog:

[http://blogs.technet.com/App-V/archive/2007/07/12/sequencing-best-practices.aspx](http://blogs.technet.com/softgrid/archive/2007/07/12/sequencing-best-practices.aspx)

1. Gauging Applications for Sequencing

All applications are different and therefore no application will require the same amount of time to sequence. However, sequencing estimates can be put into four different categories based on the complexity of the application, size (in both size on disk and number of files), and finally reliance on resources outside of the virtual application. Another item to take special note of before sequencing begins: nothing will slow down the sequencing process more than not having access to someone who inherently understands the full functionality of the application.

This section includes approximate sequencing times. However, these are very general in nature. Every application will be different, and the times are present only to assist in *estimating* the time required for a project.

|  |  |  |
| --- | --- | --- |
| Application Type | Description | Time Scale |
| Simple | These applications are normally small ones. An example of an application in this category would be WinZip or Adobe Reader. These applications are very straightforward and normally small in size (usually under 100MB). Very little if any modifications are needed. | **Typical sequencing time less than 1 hour** |
| Moderate | These applications might require slight changes while sequencing to function correctly. Or in some cases they may require no changes but have a larger install that takes more time. And in rare occurrences you will encounter both. Changes you might encounter in these packages would be on the order of making changes to the registry, altering the .osd file to launch with additional parameters and scripts, or finally there maybe additional applications needed to install together as a suite so they can function together. This is probably the most common application type you will run into. | **Typical Sequencing time between 1-4 hours** |
| Complex | These are large applications or applications that take four or more hours to install, significant amounts of customization to function in the virtual environment, or both. Packages like this will normally be 3-4 GB in size and may require compression to get the package under the 4GB App-V limit. Other hurdles you may encounter are the applications reliance on files being in a specific place and functions hard coded to that install. These applications may require you to manually edit batch and command files to point to resources in the virtual environment. If this is the case it is highly recommended utilizing a program that can scan multiple files and make several changes at once. You also may be required to install a device driver separately since drivers cannot be virtualized. Applications of this complexity can be sequenced however it is imperative that before you begin that all the pieces are in place. All knowledgeable resources should be engaged and available, sequencing hardware should be better than average, and finally sequencing applications such as these should be done by an experienced sequencer who has experience with the App-V product. | **Typical Sequencing time between 4-8 hours but could be longer depending on the size and number of files** |

1. Sequencing Limitations

Sometimes there are applications that cannot or should not be sequenced. Also there are certain limitations with App-V. Here is a short list of application functions and limitations of the software.

|  |  |
| --- | --- |
| Limitation | Description |
| Applications that when sequenced are over 4GB in size | * By utilizing compression within the sequencer it’s possible to take applications that are larger than 4GB and get them below the limit. * However after compression the application must be smaller than 4GB. * If the application is too large then the sequencer will not save the application. It will however attempt to compress the file. |
| Applications that start services at boot time | * App-V requires a logged in user to initiate the launch of an application. |
| Applications that require device drivers | * App-V cannot virtualize drivers. It is possible to bypass this issue and install the driver locally on the target computer. * Some user-mode device drivers can be virtualized. |
| Applications that are required by several applications for information or access | * For example a program that will initiate a command and launch another program. Normally you would include both programs in the same suite however if this application launches or initiates commands in several applications it may not be feasible for you to include all of the applications in the same suite. * This is especially true if one of the reasons you are deploying App-V is to avoid application conflicts. Always remember the virtual “bubble” can see the OS and what’s installed on it but the OS cannot see the “bubble” and interact with it. On the same note remember that one “bubble” cannot see another. |
| Applications that are a part of the OS | * Such as Internet Explorer |
| Applications that use COM+ | * Because COM+ is dynamic and happens at runtime there’s no way for the sequencer to capture this information. |
| COM DLL surrogate virtualization | * i.e. DLL’s that run in Dllhost.exe |

1. Sequencing Phases

There are four phases to sequencing a package; Installation, Launch, Customization, and Save. In this section I’ll give a high level overview of each phase.

* 1. Installation Phase

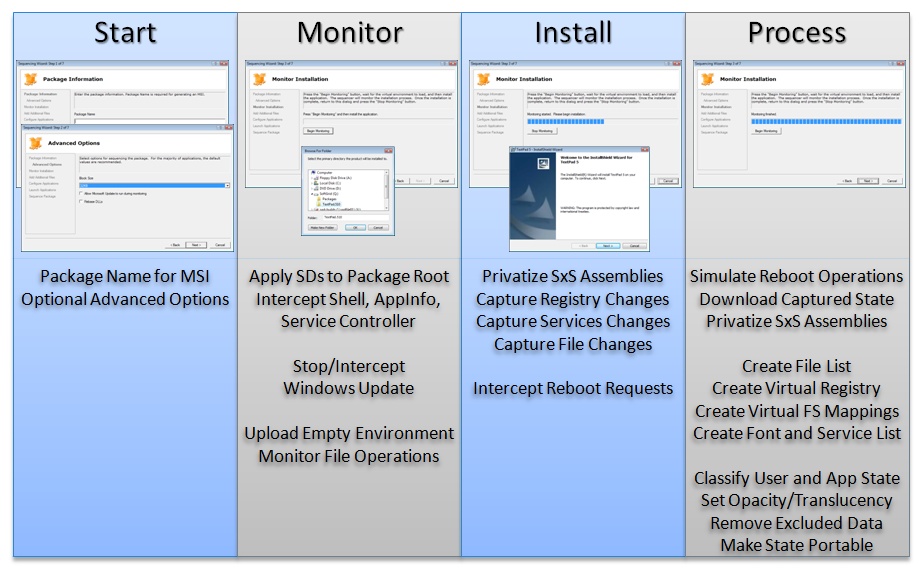


Figure 1: A walkthrough and brief description the different steps of the installation phase

* 1. Launch Phase

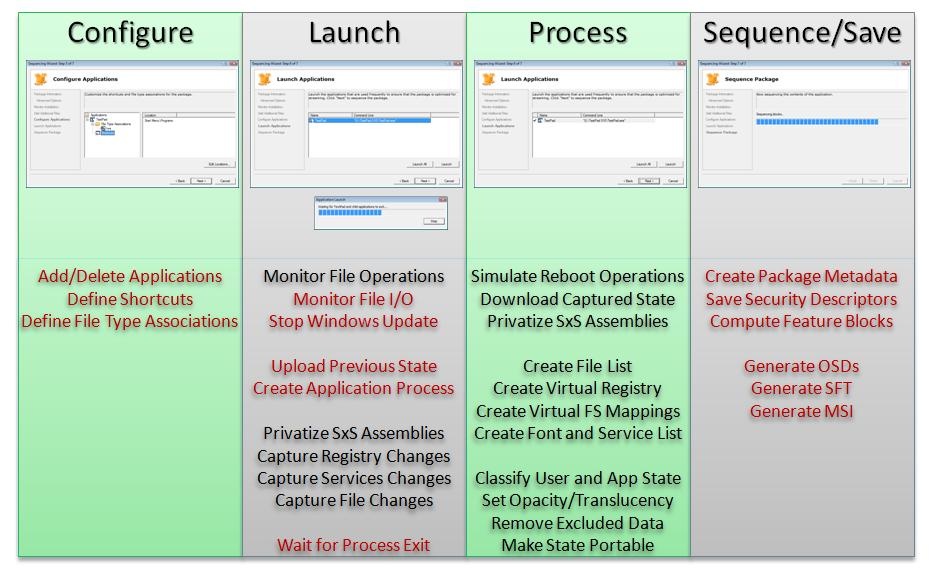
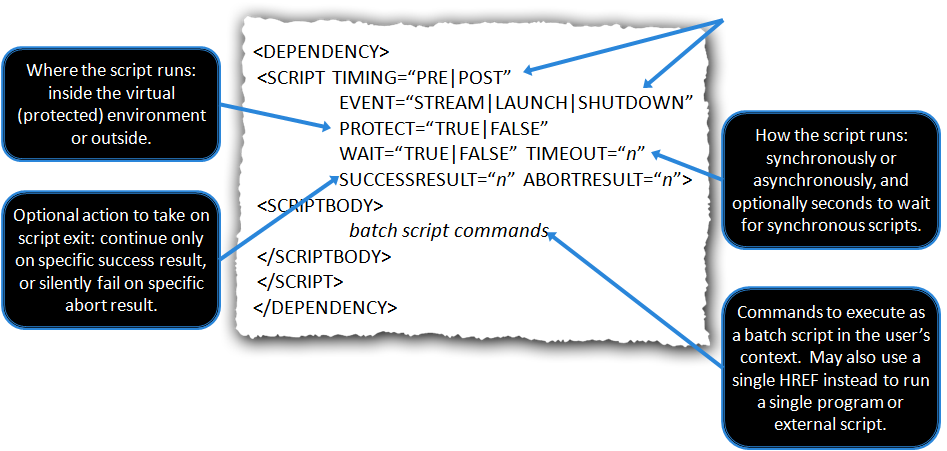


Figure 2: A walkthrough and brief description of the different steps of the launch phase

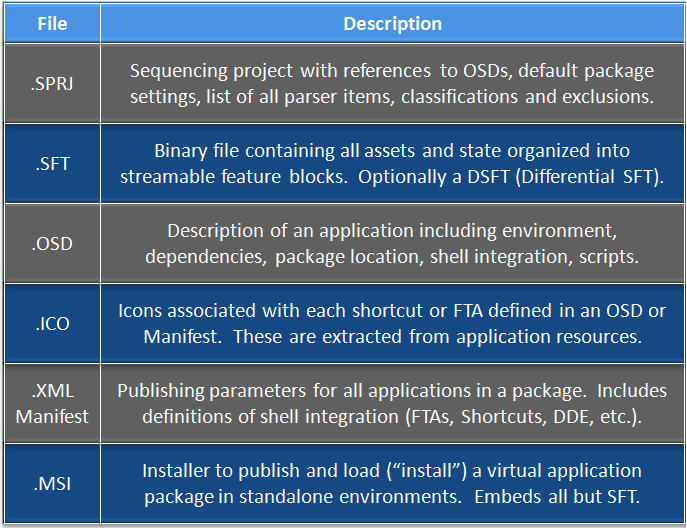
* 1. Customization Phase

During this phase you can make customizations to your package in the OSD files. The following is a brief example of what is in the OSD file. A more detailed description can be found in the “Advanced OSD Scripting” section later in this document.



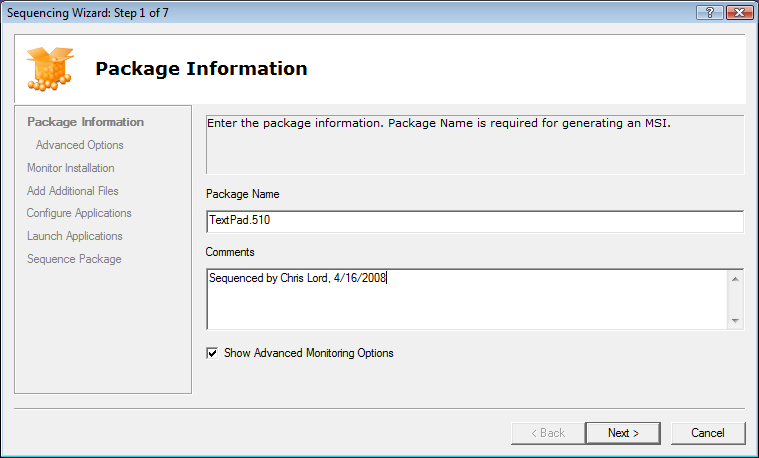
* 1. Save Phase

When you save your package you will have several files. The following is a list of those files and description of each of them.



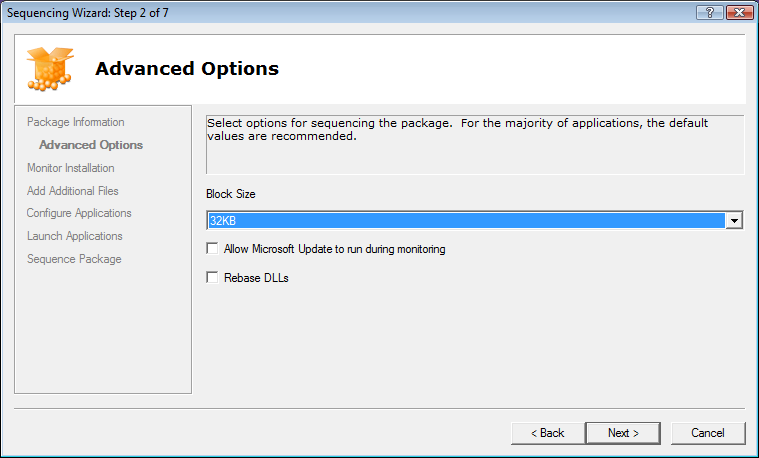
1. Sample Sequencing

This section describes some of the key points to remember during sequencing. To begin click file and select “new package”.

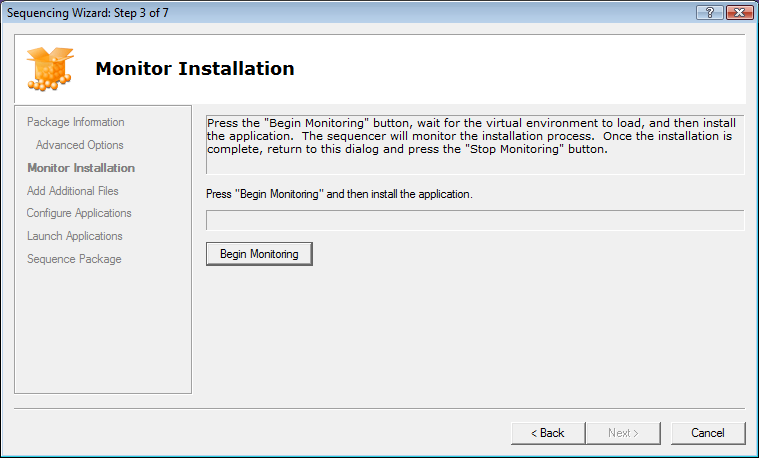
* 1. Package Information

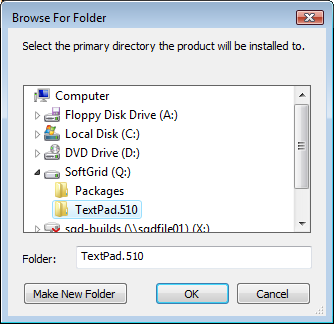
|  |  |
| --- | --- |
| Package Information Component | Description |
| Package Name | The Package Name must be unique. Microsoft uses the following naming convention: <Appname>\_<AppVendor>\_<Version>\_<MNT | VFS>, where MNT indicates the application was installed to Q: and VFS indicates the application was installed to C: |
| Comments | Indicates the person who performed the sequence, the date the Sequence was created and the version of the Sequencer. This information is recorded in the ABSTRACT element of the OSD file, and can be used to understand the pedigree of the sequence |
| Advanced Monitoring  Checkbox | Checking this box will display the Advanced Options dialog seen in 7.2 below. |

* 1. Advanced Options



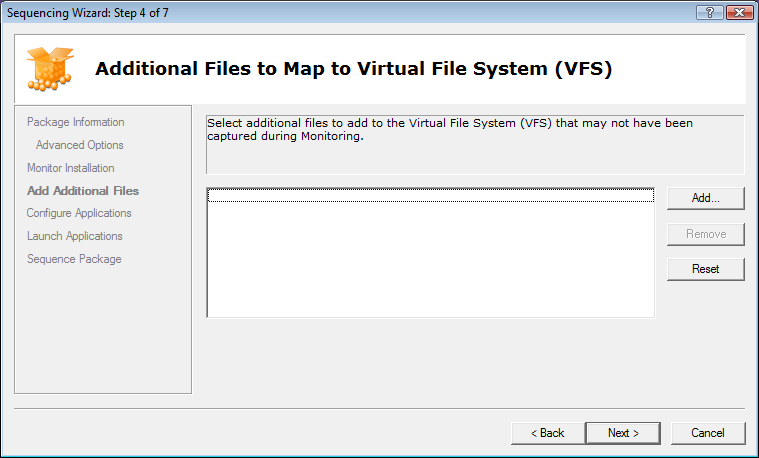
|  |  |
| --- | --- |
| Advanced Options Component | Description |
| Block Size | Specify the size of packets that are sent down that are not part of Feature Block 1. The recommended size is 32 KB |
| Allow Microsoft  Update | Allows Microsoft Update to be run as part of the sequencing process. Typically this is used to let Microsoft Update to update Microsoft Office during the package update process. However it is important to remember that the sequencer cannot capture OS components |
| Rebase DLL’s | Results in DLLs being remapped to a contiguous address space in RAM to increase efficiency. This option is only relevant for applications being deployed to a Terminal Server environment. Some applications may not support this feature and therefore rebasing is disabled by default. |

* 1. Monitor Installation



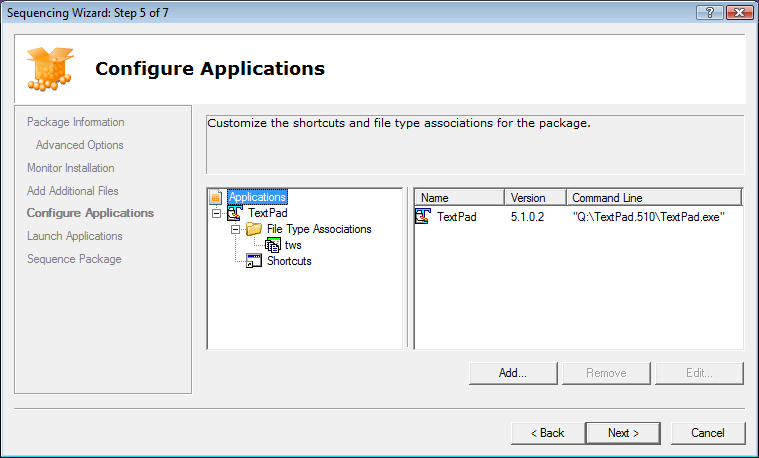
|  |  |
| --- | --- |
| Monitor Installation Component | Description |
| Begin Monitoring | Click the Begin Monitoring button then install your application. When done Click the Stop Monitoring Button |
| Create installation Folder | After you click “Begin Monitoring” you will be prompted to create the installation folder at this time. In version 4.2 this was done when you clicked “Stop Monitoring” |

* 1. Adding Files



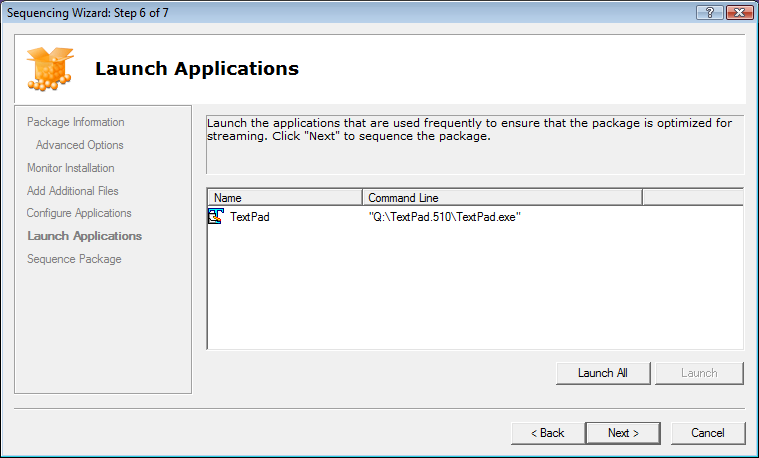
|  |  |
| --- | --- |
| Additional Files  Component | Description |
| Add Files | At this stage you can add additional files to the VFS manually |

* 1. Configure Applications



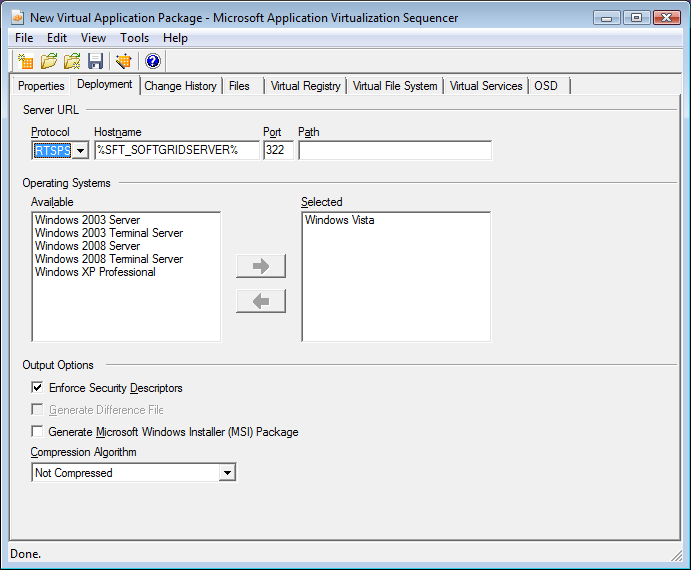
|  |  |
| --- | --- |
| Configure Apps Component | Description |
| Edit Applications | At this point you can add or remove icons for applications. Additionally you can edit the file type associations and the icon locations on the client. |

* 1. Launch Applications



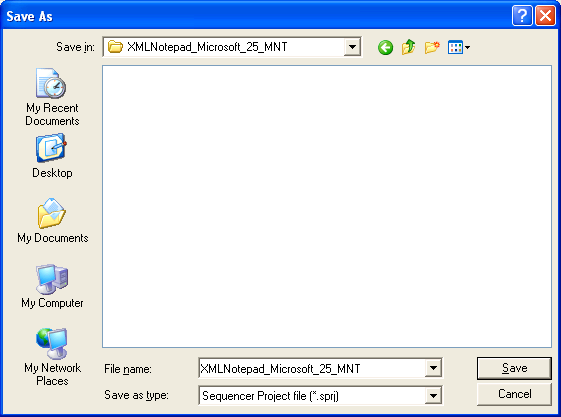
|  |  |
| --- | --- |
| Launch Applications Component | Description |
| Launch | Here you will launch your application to ensure functionality. Additionally any files that are executed during this phase will be tagged as feature block 1. Any remaining files will be tagged as feature block 2. If you don’t launch any application then all files will be a part of feature block 1. |

* 1. Package Deployment



|  |  |
| --- | --- |
| Package Deployment | Description |
| Protocol | Select from the following protocols: RTSP, RTSPS, HTTP, HTTPS, and FILE |
| Hostname | Define the hostname of your APP-V server. Leave the default %SFT\_SOFTGRIDSERVER% if you will be defining this variable on your client machines. |
| Port | This will be defined when you select the protocol that you want. |
| Path | This is the relative path of the content of the App-V server. |
| Operating Systems | Select the Operating Systems that you want the application to run on. |
| Output Options | 1. Enforce Security Descriptors - When this option is checked, the sequencer captures security descriptors at packaging time and the client enforces them on the file system drive at runtime. 2. Generate MSI Package – Select to generate a MSI package for standalone deployment scenarios. |
| Compression Algorithm | Select ZLIB if you need to compress the size of your package. Note that this only compresses the package itself. After it is streamed to the client it will expand to its full size. |

* 1. Saving the package



|  |  |
| --- | --- |
| Package Save | Description |
| Save | At this stage you’re ready to save your package. One easy method of ensuring that the filename and folder you save in are unique is to use the same PACKAGE NAME that you specified in the Package Information wizard. |

1. Application Package Upgrade

Throughout the lifecycle of an application, applications will need to be updated on occasion. It is important to understand the options associated with updating an application, and when to use the different options.

* 1. Updating a Package that will replace the existing package

Sometimes, a package needs to be updated and replace the existing package. In this case, follow these steps when sequencing and subsequently publishing in production:

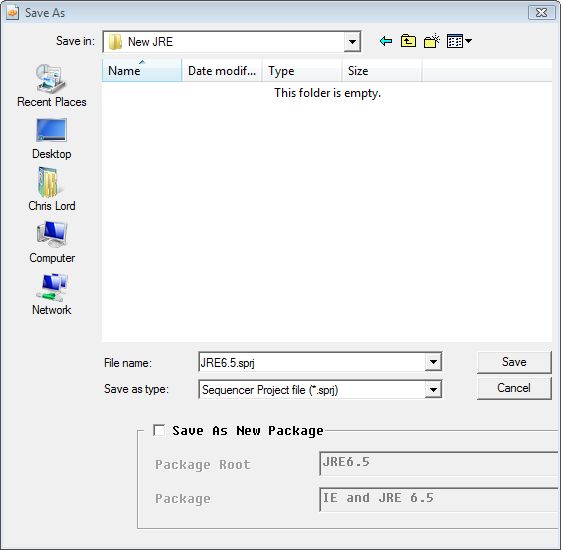
* Copy the package to the Sequencer.
* Open the package for Package Upgrade
* Select the sequencing drive (usually Q: ).
* Run the Installation Wizard and make the appropriate changes to the package.
* After the Installation, launch all shortcuts in the same manner as during the original sequence.
* Save the package. You have three options when saving the package. Since we are replacing an existing package you will select option 1:

1. Save - makes new SFT/OSDs, remove the old files from the directory from where the package was unpacked
2. Save As - Select a new folder, makes new SFT/OSD
3. Save As, New Package - Makes new package including a new GUID & Package Root. This will allow you to publish the application side by side with the original.

* 1. Updating a Package that will run at the same time as the existing package

Sometimes, a package needs to be updated, but the business community still wants access to the existing package. In this scenario, follow these steps:

1. Copy the package to the Sequencer.
2. Open the package for Package Upgrade
3. Select Q: to decode the package to.
4. Run the installation wizard and make the appropriate changes to the package.
5. After the Installation Wizard, run the Application Wizard. Modify either the Name or the Version for all of the shortcuts, as these must be unique for each shortcut. Launch all shortcuts in the same manner as during the original sequence.
6. Select “Save As” to save the package. Check the “Save As New Package” checkbox to create a new GUID and a new package root. Remember the package root must be 8.3 compliant and unique.



1. Dynamic Suite Composition

In previous versions of App-V if you wanted virtual applications to talk to one another it was necessary to include them in the same package. In version 4.5 it is now possible to allow two virtual applications to talk to one another by making one dependent upon another. The other major benefit of this is it allows you to update packages separately. If you had a plug-in that was used by twenty different applications in the past you would have had to package the plug-in with every suite and if you needed to update it all twenty packages would need to be updated. Now with DSC you sequence the plug-in separately and only need to update it once.

Let’s use the example of Microsoft Office and Snagit (a plug-in for office).

Here are the sequence of events:

1. Sequence Microsoft Office, create a package, then save it.
2. Go back to a clean sequencer and create your Snagit package. However before you start monitoring make sure you install Office.
3. With Notepad.exe open the OSD file for the Snagit package and copy the line that begins with codebase and copy the following bits
   1. HREF
   2. GUID
   3. SYSGUARDFILE
   4. MANDATORY - this you will type yourself. If MANDATORY=”TRUE” then the plugin will be required to run the main application so access must be given to it to allow the main application to run.

<CODEBASE HREF="RTSPS://%SFT\_SOFTGRIDSERVER%:322/Snagit\_Techsmith\_9\_MNT.sft" GUID="E4DE8DA8-87BB-4200-BFE1-8F88E4863034" SYSGUARDFILE="snagit.001\osguard.cp" MANDATORY=”TRUE"/>

1. Next open the OSD file for each of your Office apps that you want to use Snagit. Above the line that says </VIRTUALENV> type the following
   1. <DEPENDENCIES>
   2. Paste the code from #3 above
   3. </DEPENDENCIES>
2. So in the end what you typed would look something like this:

<VIRTUALENV TERMINATECHILDREN="FALSE">

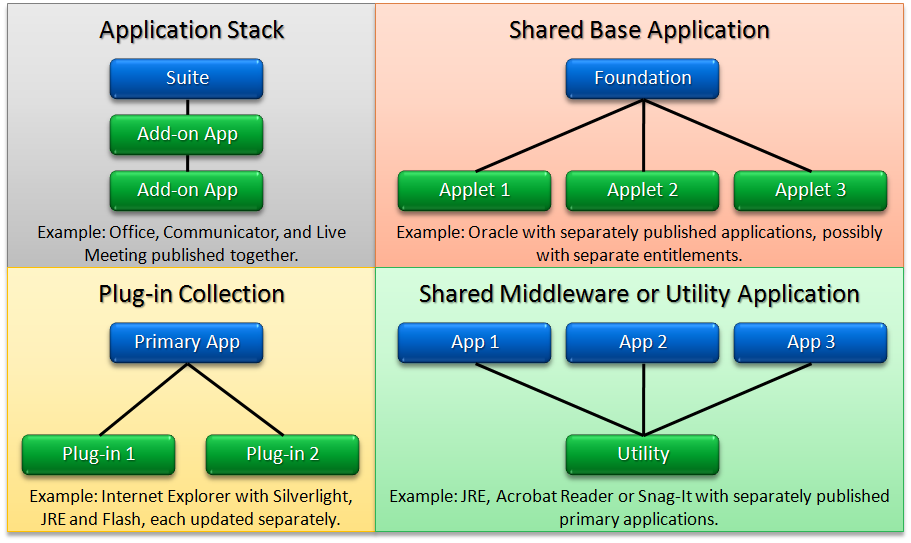
<ENVLIST/>

<DEPENDANCIES>

<CODEBASE HREF="RTSPS://%SFT\_SOFTGRIDSERVER%:322/Snagit\_Techsmith\_9\_MNT.sft" GUID="E4DE8DA8-87BB-4200-BFE1-8F88E4863034" SYSGUARDFILE="snagit.001\osguard.cp" MANDATORY=”TRUE"/>

</DEPENDANCIES>

</VIRTUALENV>



The above diagram shows four different scenarios on how you might use DSC in your environment.

1. Compression of the SFT

In some scenarios you may decide to add compression to the package that you are sequencing. If the application is compressed this can significantly reduce the size of the SFT file. This compression reduces the amount of data that will be distributed from the App-V Server to the App-V Client. The compression will impact the end client, as the client PC will need to decompress the SFT and put the data into the client’s cache. The decompression algorithm is CPU intensive. The default setting is “No Compression” and should be left unless the target LAN for streaming is excessively bandwidth limited.

1. Advanced OSD Scripting

In some instances it may be necessary to make modifications to the application to get it to function properly in the App-V environment. One of the most common methods is making modifications to the OSD file to allow certain actions to be performed at a specified time during the execution of the application.

Before we talk about scripting here is a generalized look at a typical OSD file.

<**SOFTPKG** GUID="*application GUID*" NAME="*application name*" VERSION="*version string*">

<**IMPLEMENTATION**>

<**CODEBASE** HREF="*location*“ GUID="*package*" FILENAME="*application*“

PARAMETERS=“*parameters*” SYSGUARDFILE="*state file*"/>

<OS VALUE="*platform*"/>...<OS VALUE="*platform*"/>

<WORKINGDIR>*optional working directory*</WORKINGDIR>

<**VIRTUALENV** *virtual environment configuration options*>

<ENVLIST>*optional variables*</ENVLIST> <REGISTRY>*optional values*</REGISTRY>

<POLICIES>*optional policies*</POLICIES>

<DEPENDENCIES>

<CODEBASE ... MANDATORY="TRUE|FALSE"/>...<CODEBASE ... MANDATORY="TRUE|FALSE"/>

</DEPENDENCIES>

</**VIRTUALENV**>

</**IMPLEMENTATION**>

<**DEPENDENCY**>

<CLIENTVERSION VERSION="4.5.0.0"/> <SCRIPT>*optional scripts*</SCRIPT>

</**DEPENDENCY**>

<**MGMT\_SHORTCUTLIST**><SHORTCUT *definition*>…<SHORTCUT …></**MGMT\_SHORTCUTLIST**>

<**MGMT\_FILEASSOCIATIONS**>

<PROGIDLIST>*shell integration*</PROGIDLIST><FILEEXTENSIONLIST>*FTAs*</FILEEXTENSIONLIST>

</**MGMT\_FILEASSOCIATIONS**>

</SOFTPKG>

When creating a script it is important to understand when scripting events occur so you know when you will want your script executed. The following information is helpful to understand how the virtual environment is constructed and torn down and where the various scripting event triggers occur:

* Virtual Application Launch / Shutdown Progression
  + User clicks on program shortcut for a virtual application
  + **Pre-Stream script runs (outside VE)**
  + Page streams FB1
  + **Post-Stream script runs (outside VE)**
  + Virtual environment (VE) is loaded
  + **Pre-Launch script runs (inside or outside VE**)
  + The virtual application launches
  + **Post-Launch script runs (inside or outside VE)**
  + If background streaming is enabled for the application,   
    FB2 will stream into the client cache in the background.
  + User exits the virtual application
  + The virtual environment is unloaded
  + **Post-shutdown script runs (outside VE)**

Note that the client does not support a **SHUTDOWN** event with Timing of **PRE**.

The **PROTECT** attribute specifies if the script runs inside or outside of the virtual environment.

The **WAIT** attribute specifies if the client waits for the script to finish before proceeding to the next phase.

Scripts can be written in any language but their language must be installed locally on the client machine. For example if they are using PERL, the PERL runtime must be locally installed on the client machine.

There are two kinds of scripts **<SCRIPTBODY>** and **<HREF>**:

* HREF – Normally used when executing a single command
* SCRPITBODY – Used when needing more complex operations and/or multiple commands to be run.
  1. <SCRIPTBODY> Examples

**Example 1 – ABORTRESULT and SUCCESSRESULT**

If ABORTRESULT is specified, the client will check the script's exit code when it completes. If SUCCESSRESULT is specified (and if ABORTRESULT is not or it does not match the script's exit code), the client will check the script's exit code to see if it matches the value in SUCCESSRESULT.

<DEPENDENCY> <SCRIPT TIMING="PRE" EVENT="LAUNCH" PROTECT="TRUE" WAIT="TRUE" TIMEOUT="10" SUCCESSRESULT="1" ABORTRESULT="0">

<SCRIPTBODY>@ if not %COMPUTERNAME% == “SpecialComputer” exit 1 @ exit 0

</SCRIPTBODY>

</SCRIPT>

</DEPENDENCY>

A more detailed explanation of these commands can be found at:

<http://support.microsoft.com/kb/930973/en-us>

**Example 2 – Pre-Launching Command Prompt**

<DEPENDENCY>

<SCRIPT TIMING="PRE" EVENT="LAUNCH" WAIT="TRUE" PROTECT="TRUE">

**<SCRIPTBODY>cmd.exe</SCRIPTBODY>**

</SCRIPT>

</DEPENDENCY>

**Example 3 – Embedding Batch File Commands within OSD File**

<DEPENDENCY>

<SCRIPT EVENT="LAUNCH" TIMING="PRE" PROTECT="TRUE" WAIT="TRUE">

**<SCRIPTBODY>**

**net use k: \\\\w2k-pdc\\netlogon\n**

**CALL k:\\usr-w2k.cmd\n**

**\\\\sfc-App-V\\shr\editini.exe c:\\word\\word.ini "File Locations" TempPath c:\\temp\n**

**</SCRIPTBODY>**

</SCRIPT>

</DEPENDENCY>

**Example 4 – Pre-Launching EXE from within the Sequence**

<DEPENDENCY>

<SCRIPT TIMING="PRE" EVENT="LAUNCH" WAIT="TRUE" PROTECT="TRUE">

**<SCRIPTBODY>%SFT\_MNT%\\OfficeXP\\Office10\\proflwiz.exe</SCRIPTBODY>**

</SCRIPT>

</DEPENDENCY>

**Example 5 – Pre-Launching a Data File (e.g. AVI file) from within the Sequence**

<DEPENDENCY>

<SCRIPT TIMING="PRE" EVENT="LAUNCH" WAIT="TRUE" PROTECT="TRUE">

**<SCRIPTBODY>%SFT\_MNT%\\App\\Data\\Sequencing.avi</SCRIPTBODY>**

</SCRIPT>

</DEPENDENCY>

**NOTE:** In this example, the sequencing.avi launches the locally installed Media Player.

**NOTE:** A data file (e.g. sequencing.avi) cannot be specified within the FILENAME entry of the CODEBASE tag as shown below.

<CODEBASE

**FILENAME="App\Data\Sequencing.avi"**

SYSGUARDFILE="App\osguard.cp"

HREF="rtsp://App-V:554/app.sft"

GUID="A2A6D95E-8898-41C0-8020-797B23A1E917"

SIZE="33363442"/>

**Example 6 – Pre-Launching a Control Panel Applet within the Sequence**

<DEPENDENCY>

<SCRIPT TIMING="PRE" EVENT="LAUNCH" WAIT="TRUE" PROTECT="TRUE">

**<SCRIPTBODY>%SFT\_MNT%\\App\\CPL\\SYSDM.CPL</SCRIPTBODY>**

</SCRIPT>

</DEPENDENCY>

**Example 6a – Pre-Launching a Control Panel Applet within the Sequence**

<DEPENDENCY>

<SCRIPT TIMING="PRE" EVENT="LAUNCH" WAIT="TRUE" PROTECT="TRUE">

<SCRIPTBODY>

**"C:\\Program Files\\Common Files\\System\\Mapi\\1033\\mlcfg32.cpl"**

</SCRIPTBODY>

</SCRIPT>

</DEPENDENCY>

* 1. <HREF> Examples

**Example 1 – Accessing local application**

<DEPENDENCY>

<SCRIPT TIMING="PRE" EVENT="LAUNCH" WAIT="TRUE" PROTECT="TRUE">

**<HREF>c:\winnt\system32\cmd.exe</HREF>**

</SCRIPT>

</DEPENDENCY>

**Example 1a – Accessing Batch File via UNC**

<DEPENDENCY>

<SCRIPT TIMING="PRE" EVENT="LAUNCH" WAIT="TRUE" PROTECT="TRUE">

**<HREF>\\SRV\_NAME\NT\_SHR\PRE\_SCRIPT.CMD</HREF>**

</SCRIPT>

</DEPENDENCY>

**NOTE:** HREF tag cannot be used with %SFT\_MNT% or to launch EXE from within virtual environment.

* 1. <PARAMETERS> Examples

**NOTE:** The PARAMETERS tag will not support environment variables (e.g. %SFT\_MNT%).

**Example 1 – Generic Example**

<CODEBASE

**FILENAME="\ABCD\EXECUTE\Z\_RUN.EXE"**

**PARAMETERS="FORM=MRF01 LANG=ENG ZENV=X\_POINT\_CHAR"**

SYSGUARDFILE="ACBS\osguard.cp"

HREF="rtsp://hercules:554/ACBS\_V2.sft"

GUID="BD9AF94E-4A8E-415D-B57F-54A454911FAA"

SIZE="648019522"/>

**Example 2 – Launching Data File via a Relative Path**

<CODEBASE

**FILENAME="App\WordEditor\wordpad.exe"**

**PARAMETERS=" ..\Data\cli.doc"**

SYSGUARDFILE="App\osguard.cp"

HREF="rtsp://App-V:554/app.sft"

GUID="A2A6D95E-8898-41C0-8020-797B23A1E917"

SIZE="33363442"/>

**NOTE:** ‘..\Data\cli.doc’ is located in the following location within the sequence:

%SFT\_MNT%\App\Data\cli.doc

* 1. <WORKINGDIR> Examples – Specifying a Working Directory

**Example 1**

<IMPLEMENTATION>

**<WORKINGDIR>C:\temp</WORKINGDIR>**

<CODEBASE HREF="rtsp://ds1-soft1:554/np\_ecj.sft?Customer=Sequencer"

FILENAME="np\_ecj\notepad.exe" PARAMETERS="cjm.txt"

GUID="D47C0A23-32CF-11D6-990A-00A0CC6413D5"/>

<OS VALUE="WinXP"/>

</IMPLENENTATION>

**Example 2**

<IMPLEMENTATION>

**<WORKINGDIR>C:\program files\internet explorer</WORKINGDIR>**

<CODEBASE HREF="rtsp://ds1-soft1:554/np\_ecj.sft?Customer=Sequencer"

FILENAME="np\_ecj\notepad.exe" PARAMETERS="cjm.txt"

GUID="D47C0A23-32CF-11D6-990A-00A0CC6413D5"/>

<OS VALUE="WinXP"/>

</IMPLENENTATION>

**Example 3 – Example with Variable**

<IMPLEMENTATION>

**<WORKINGDIR>%SFT\_MNT%\App\Data</WORKINGDIR>**

<CODEBASE HREF="rtsp://ds1-soft1:554/np\_ecj.sft?Customer=Sequencer"

FILENAME="np\_ecj\notepad.exe" PARAMETERS="cjm.txt"

GUID="D47C0A23-32CF-11D6-990A-00A0CC6413D5"/>

<OS VALUE="WinXP"/>

</IMPLENENTATION>

* 1. <REGISTRY> Examples – Registry Customizations

<IMPLEMENTATION>

<CODEBASE … />

<VIRTUALENV>

**<REGISTRY>**

**<REGKEY HIVE="HKLM" KEY="Software\Test">**

**<REGVALUE REGTYPE="REG\_SZ" NAME="string">Holas</REGVALUE>**

**<REGVALUE REGTYPE="REG\_DWORD" NAME="dword">5051</REGVALUE>**

**<REGVALUE REGTYPE="REG\_BINARY" NAME="hex">50,51,52</REGVALUE>**

**<REGVALUE REGTYPE="REG\_SZ">Value of Default Key</REGVALUE>**

**</REGKEY>**

**</REGISTRY>**

</VIRTUALENV>

</IMPLENENTATION>

* 1. Launching Local Applications

**NOTE:** The PARAMETERS tag will not support environment variables (e.g. %SFT\_MNT%).

**Example 1 – Launching Application that is local (not part of the Sequence)**

<CODEBASE

**FILENAME="C:\Program Files\Internet Explorer\iexplore.exe"**

SYSGUARDFILE="App\osguard.cp"

HREF="rtsp://App-V:554/app.sft"

GUID="A2A6D95E-8898-41C0-8020-797B23A1E917"

SIZE="33363442"/>

<VM VALUE="Win32">

<SUBSYSTEM VALUE=”Windows”/>

</VM>

<CODEBASE

**FILENAME=" C:\Program Files\Windows Media Player\wmplayer.exe"**

**PARAMETERS=" q:\app\data\sequencing.avi"**

SYSGUARDFILE="App\osguard.cp"

HREF="rtsp://App-V:554/app.sft"

GUID="A2A6D95E-8898-41C0-8020-797B23A1E917"

SIZE="33363442"/>

<VM VALUE="Win32">

<SUBSYSTEM VALUE=”Windows”/>

</VM>

**NOTE:** C:\Program Files\Windows Media Player\wmplayer.exe is a local application.

* 1. Multiple Operating System Support

<VIRTUALENV>

<ENVLIST>

<ENVIRONMENT VARIABLE="PATH">%PATH%;%SFT\_MNT%\vs2005\Microsoft SDK\bin\.;</ENVIRONMENT>

<ENVIRONMENT VARIABLE="W2K3DDK">%SFT\_MNT%\vs2005\winddk\3790~1.183</ENVIRONMENT>

<ENVIRONMENT VARIABLE="W2K3SP1DDK">%SFT\_MNT%\vs2005\winddk\3790~1.181</ENVIRONMENT>

</ENVLIST>

</VIRTUALENV>

* 1. Launching Application via Non-Default Provider Policy

<CODEBASE

FILENAME="Photoshp\Photoshop.exe"

SYSGUARDFILE="Photoshp\osguard.cp"

HREF="rtsp://sgserver:554/adobephotoshop70.sft**?Customer=Sales**"

GUID="FE72DDCD-6E80-4ABE-8E82-8208A134CB34"

SIZE="178299260"/>

* 1. Running an Application as a CONSOLE application

<OS VALUE="Win2K"/>

<OS VALUE="Win2KSvr"/>

<OS VALUE="Win2KTS"/>

<OS VALUE="Win2003Svr"/>

<OS VALUE="Win2003TS"/>

<OS VALUE="WinXP"/>

<VIRTUALENV>

<ENVLIST/>

</VIRTUALENV>

**<VM VALUE="Win16">** Can also be “Win32”􀃅 **WARNING! VM Values are case sensitive; WIN16 does not work.**

**<SUBSYSTEM VALUE="console"/>**

**</VM>**

</IMPLEMENTATION>

* 1. OSD XML Syntax Check

To check the XML syntax of an OSD file, change the .OSD file extension to .XML and drop the .XML file into an Internet Explorer Window.

Note: XML file editors such as XML Notepad can also be used to check XML file syntax

Example XML file syntax display: <?xml version="1.0" standalone="no" ?>

**-** <SOFTPKG GUID="**26998703-328D-41DE-A256-5AFACD27ED08**" NAME="**WinZip**"

VERSION="**90**">

**-** <IMPLEMENTATION>

<CODEBASE

HREF="**rtsp://ntlabsg01:554/Winzip\_Winzip\_901\_MNT/Winzip\_Winzip\_901\_**

**MNT.sft**" GUID="**A9800C2A-E9ED-4711-B90C-808D6DD0AABC**"

FILENAME="**Winzip.v1\WINZIP32.EXE**" PARAMETERS=""

SYSGUARDFILE="**Winzip.v1\osguard.cp**" SIZE="**6225393**" />

<OS VALUE="**Win2K**" />

<OS VALUE="**Win2KSvr**" />

<OS VALUE="**Win2KTS**" />

<OS VALUE="**Win2003Svr**" />

<OS VALUE="**Win2003TS**" />

<OS VALUE="**WinXP**" />

**-** <VM VALUE="**Win32**">

<SUBSYSTEM VALUE="**windows**" />

</VM>

**-** <VIRTUALENV>

<ENVLIST />

</VIRTUALENV>

</IMPLEMENTATION>

**-** <DEPENDENCY>

<CLIENTVERSION VERSION="**2.0.5.0**" />

</DEPENDENCY>

<SUITE NAME="**Winzip\_Winzip\_901\_MNT**" />

<TITLE />

<ABSTRACT>**Win XP Sp1, IE 6.0 Seq Build 300**</ABSTRACT>

**-** <MGMT\_SHORTCUTLIST>

<SHORTCUT LOCATION="**%CSIDL\_STARTMENU%**" WORKINGDIR=""

DISPLAY="**WinZip**" ICON="**%SFT\_MIME\_SOURCE%/WinZip90.ico**" />

<SHORTCUT LOCATION="**%CSIDL\_DESKTOPDIRECTORY%**" WORKINGDIR=""

DISPLAY="**WinZip**" ICON="**%SFT\_MIME\_SOURCE%/WinZip90.ico**" />

</MGMT\_SHORTCUTLIST>

</SOFTPKG>

1. Advanced Sequencing Techniques

This section will briefly cover some sequencing challenges and how to overcome them. It is also recommended to be aware of the logs created when sequencing, a list of the log files related to the sequencer process and their purpose will be .

* 1. Sequencing Applications That Cannot Install to Q:\

### At some point while sequencing applications you will most likely run across an application that will not let you install to Q:\. This by itself does not present a problem when sequencing the application. In fact only one minor modification will need to be made.

### When you hit the “Start Monitoring” button you will browse to the root of Q:\.

### Make a directory for your application using the 8.3 naming convention and select this directory as the asset directory

### Select Ok and continue sequencing as normal.

### There is a reason that this technique is at the end of the guide. There are pro’s and con’s when sequencing with this method.

* 1. Sequencing Web Based Applications

Internet Explorer is NOT supported in a sequenced application. However you can sequence plug-ins or ActiveX controls for Internet Explorer.

This pulls the locally installed application into the virtual environment with the additions you have specified. In this case it was a web plug-in. This lets you have a clean and secure Internet Explorer that you can lock down on a users machine and then allow them to use the plug-ins that you define.

* 1. Sequencing applications that require access to local devices and or resources

In many ways you sequence these applications no different than you would any other. The applications inside the virtual environment can see the local resources on the OS and can access them. Devices are another matter. App-V cannot virtualize drivers. If you can install the driver separately from the application then the application will fall through to the OS to look for the driver and see that it is installed.

* 1. Sequencer Log Files
* ***Sft-seq-log.txt***  
  Description: The main log file for logging on the sequencer.  
  Path: ProgramFiles%\Microsoft Application Virtualization Sequencer\Logs\Sft-seq-log.txt
* ***Sftrbt.txt***  
  Description: Stores the actions taken during the reboot processing.  
  Path: %ProgramFiles%\Microsoft Application Virtualization Sequencer \App-V Sequencer\Logs\Sftrbt.txt
* ***SFTCallBack.txt***  
  Description: Records all process starts and stops seen by SystemGuard.  
  Path: %ProgramFiles%\Microsoft Application Virtualization Sequencer \App-V Sequencer\Logs\SFTCallBack.txt
* ***Filter.log***  
  Description: A binary log that stores file activity during sequencing.  
  Path: %ProgramFiles%\Microsoft Application Virtualization Sequencer \App-V Sequencer\Logs\Filter.1.log
* ***Files.txt***  
  Description: Logs all files in the VFS. It must be enabled by setting   
  HKLM\Software\Softricity\App-V  
  Sequencer\CurrentVersion\Configuration\FileManifest to “1”. (REG\_DWORD)  
  Path: %ProgramFiles%\Microsoft Application Virtualization Sequencer\App-V Sequencer\Logs\File.txt

1. Finding Additional Information
   1. Error Codes and the Microsoft Knowledge Base

Microsoft has completed the process of migrating data off of the old Softricity knowledge base. The knowledgebase contains useful information regarding planning, implementation and troubleshooting App-V. A link to the Microsoft knowledgebase follows:

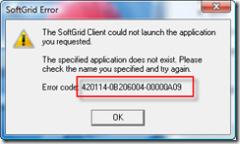
<http://support.microsoft.com>

*Error Code Formats*: When searching the knowledge base for error codes, be aware that the first 12 digits are unique to the version of the App-V client software. The ending 8 digits of the error code remain constant between all client versions.

When you search for an error code, enter the last 8 digits such as 000DF004.

e.g xxxxxx - xxxxxxxx - xxxxxxxx

or

[](http://blogs.technet.com/blogfiles/virtualworld/WindowsLiveWriter/Sowhatdoalltheseerrorsactuallymean_1470B/image_1.png)

The First block of 6 characters are around the Version build of the platform/client. Basically when you see these its information based and will vary depending on;

* Major Version
* Minor Version
* Patch Version
* Build

The Next Block of 8 Characters Identify

* File ID
* Line
* Info

The Last Block of 8 Characters show;

* Problem Code

The Last 10 digits of the error code are constant between the SoftGrid Builds. Because the last 10 digits are consistent we tend to use these for identify and searching our errors as regardless of the Version, Builds, etc we can use this knowledge to identify problems and issues as these codes are reused between modules .

In addition, Justin Zarb from Premiere Field Engineering in the UK has written an excellent article relating to the App-V error codes and what they mean. The article can be found at:

<http://blogs.technet.com/virtualworld/archive/2007/07/27/so-what-do-all-these-errors-actually-mean.aspx>

* 1. App-V Team Blog

This is probably the single greatest repository of App-V related information on the web. This should be your first stop for anyone interested in learning about App-V and Sequencing. The blog is updated frequently so keep checking it for more information. Additionally look through the archives for valuable information as well.

* App-V Team Blog - [http://blogs.technet.com/App-V/default.aspx](http://blogs.technet.com/softgrid/default.aspx)
* RSS Feed - [http://blogs.technet.com/App-V/rss.xml](http://blogs.technet.com/softgrid/rss.xml)
  1. App-V Related Technical Discussion Forums and Web Sites

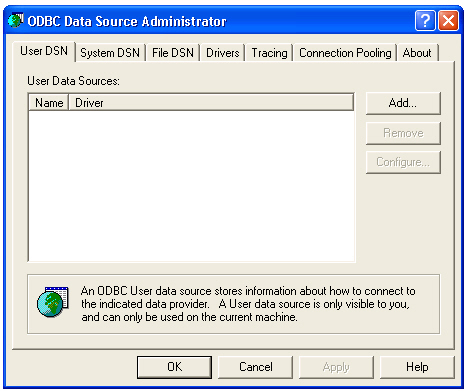
Here is a list of non-Microsoft sites that contain information relating to the App-V product and sequencing. Some of these sites include useful App-V related technical discussion forums. While these sites contain valuable information about App-V it should be noted that these sites are not affiliated with Microsoft in any way and cannot verify any of the information contained within.

* <http://www.softgridguru.com/>
* <http://www.brianmadden.com>
* <http://blogs.technet.com/virtualworld>
* <http://forums.microsoft.com/TechNet/default.aspx?ForumGroupID=497&SiteID=17>

# Appendix A

Step by step instructions for setting up dummy ODBC entries.

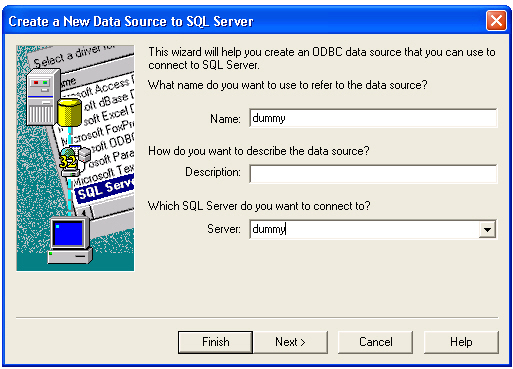
1. Open the Administrative Tools 🡪 Data Sources (ODBC) applet.
2. Select the User DSN tab.



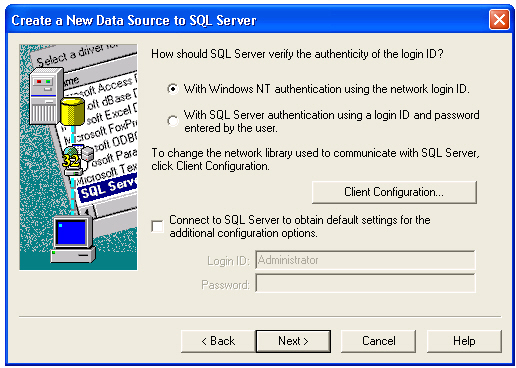
1. Click Add…

Seq SS2.pct

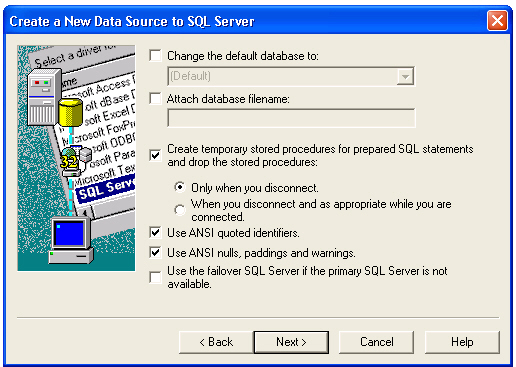
1. For “Select a driver…”, select SQL Server and click Finish.



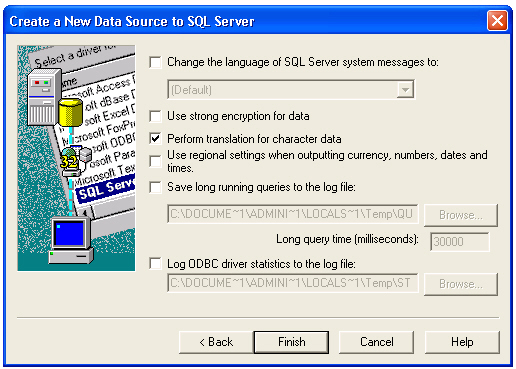
1. For Name, enter “dummy”.
2. For Description, enter “Required for application sequencing”.
3. For Server, enter “dummy”.
4. Click Next.



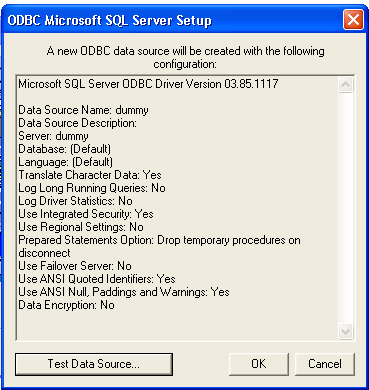
1. Deselect the “Connect to SQL Server to obtain settings…” checkbox.
2. Click Next.



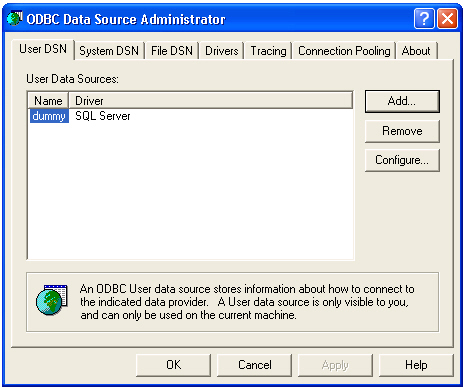
1. Click Next.



1. Click Finish.



1. Click OK. Note: Do not click “Test Data Source…” since this test would fail.



* **Create an ODBC System DSN setting as part of the Sequencer base image as follows:**

1. Select the System DSN tab.

Seq SS1.pct

1. Click Add…

Seq SS2.pct

1. For “Select a driver…”, select SQL Server and click Finish.

Seq SS3.pct

1. For Name, enter “dummy”.
2. For Description, enter “Required for application sequencing”.
3. For Server, enter “dummy”.
4. Click Next.

Seq SS4.pct

1. Deselect the “Connect to SQL Server to obtain settings…” checkbox   
   and click Next.

Seq SS5.pct

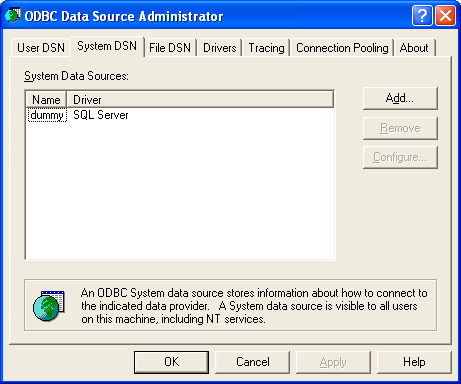
1. Click Next.

Seq SS6.pct

1. Click Finish.

Seq SS7.pct

1. Click OK. Note: Do not click “Test Data Source…” since this test would fail.

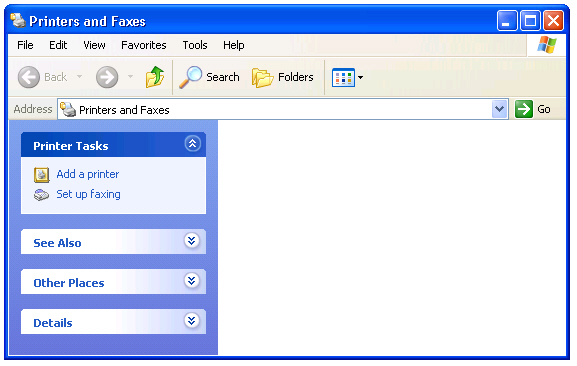


1. Click OK to close the ODBC applet.

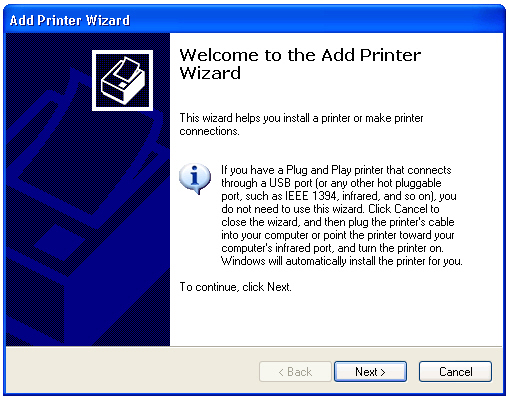
# Appendix B

Step by step instructions for setting up a dummy printer.

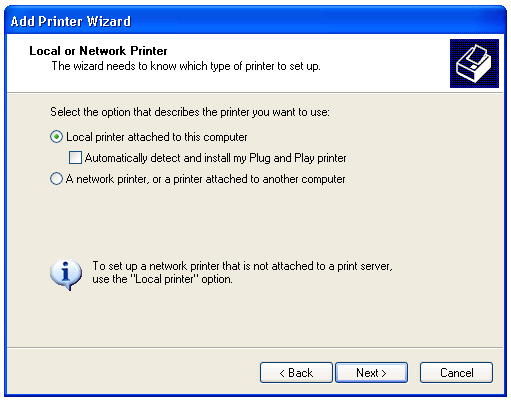
1. Open the Control Panel 🡪 Printers and Faxes applet.



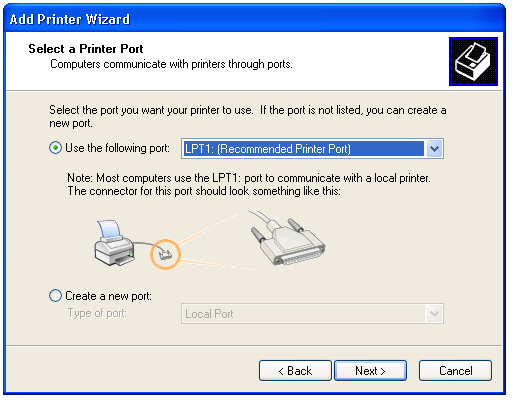
1. Select “Add a Printer”.



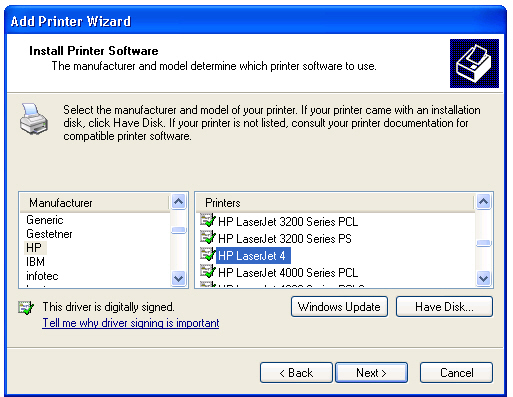
1. Click Next.



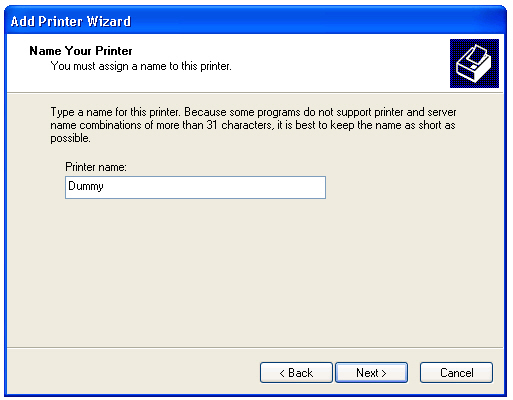
1. Select “Local printer attached to this computer”.
2. Deselect the “Automatically detect and install my Plug and Play printer” checkbox.
3. Click Next.



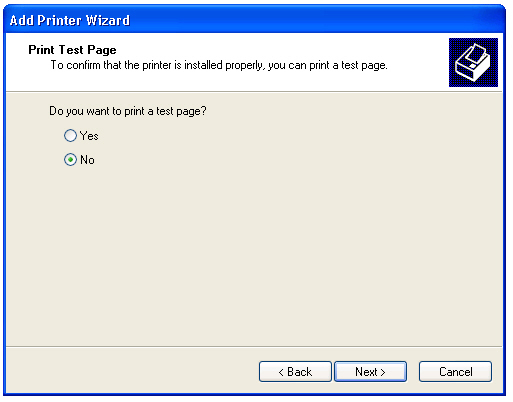
1. Click Next.



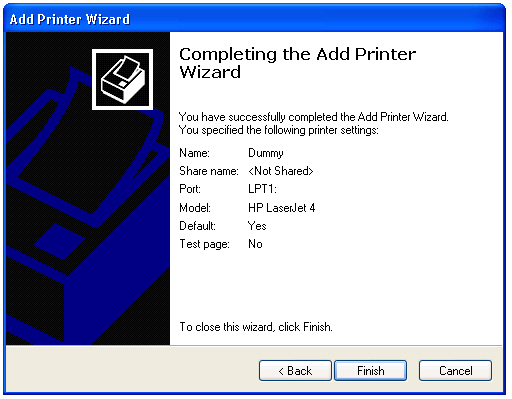
1. For Manufacturer, select “HP”.
2. For Printers, “HP LaserJet 4”.
3. Click Next.



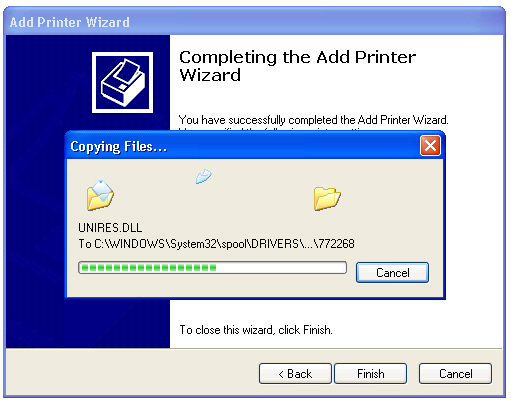
1. For Printer Name, enter “Dummy”.
2. Click Next.



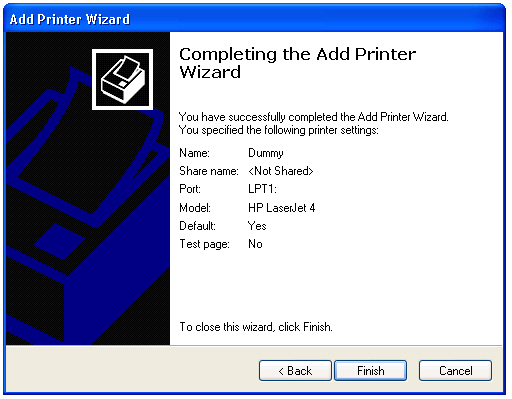
1. For “Do you want to print a test page?”, select No and click Next.



1. Click Finish.



1. Wait for the printer driver installation to complete.



1. Close the Printers and Faxes applet.