MSI Application Packaging

Billy Beaudoin

June 16th 2008

- What you’ve learned:
  - Creating a Microsoft Installer with Installshield Admin Studio

- What I’m Teaching:
  - Packaging for Different Environments
  - Best Practices in Packaging
  - Group Policy Object Best Practices
  - What is an MSI, really?

What is an MSI?

- You were taught that an msi is:
  - A database containing information about all the actions to be performed during the installation.
- Microsoft says it is:
  - Microsoft Windows Installer is an installation and configuration service that reduces the total cost of ownership.
- I think we'll stick with definition #1.

Features?

- Configurable via Policy settings
  http://msdn.microsoft.com/en-us/library/aa372058(VS.85).aspx
- Rollback when encountering errors
- Administrative Installs
- Advertisement
- Self-Repair
- Supports Per-User and Per-Machine installs
- Patches/Transforms

Packaging for Different Environments

- Usage-based
  - Kiosks
  - Public Labs
  - Desktops
  - Laptops
  - Student Owned Computers
- Technology-based
  - AD vs. Novell
  - Remote Access vs. Console
  - Interactive vs. Automated Install

User Environments

- Goal: Build the installer once
  - Sorry, still won’t be 100% portable
- Considerations:
  - Permissions – Labs/Kiosks vs. Desktop/Laptop/SOC
  - Network – Wired, Wireless, None
  - Preferences – Persistence?
  - Storage space – user profile, network drive
  - Security
- If you ever successfully build the “perfect MSI” for one environment, it’ll be useless to all others.

- Labs
  - Permissions must be right for “user”
  - Launch conditions on vendor MSI’s often look for admin
  - user data must be redirected to network
- Laptops
  - No “Install on 1st use” (read: Office)
  - Cannot require network drives
- SOC
  - Downloaded, interactive installs
- So where the heck do you put the default save location?
  - Initially assume “user” level permissions on a desktop storing files on local drive. Make modifications as needed.

Novell vs. AD for MSI Distribution

- MSI App Objects with Zenworks
  - User initiated workstation assigned install confuses the Windows Installer service
  - Use separate installer MSI app object that is preinstalled w/ Start Menu or NAL shortcuts
- Distribution/Run Options -> Group Policy Preferences
- Launch Scripts vs. Advertised Shortcuts
- No Preinstall Schedule in AD, only at reboot
- NAL doesn’t support Advertised Shortcuts
  - If its in the NAL, don’t use Adv. Shortcuts in Start Menu

Technological Install Differences

- Remote Access vs. Console
  - Some vendor app installers are not terminal services aware (Aspen, Primavera)
- Interactive vs. Automated install
  - Interactive installs should be single-file or expandable installers with as simple a UI as possible
  - Automated installs (via script, GPO, or Zenworks) can be much messier
  - Big difference is in how transforms are dealt with

Best Practices in Packaging

- Validation
- Testing
- Types of Installs
  - Snapshot vs. Installation Monitoring
  - Scripted
  - Tweak vendor msi
- Best tool for the job
- Cleaner Snapshots
  - How to Clean
  - How to reduce captured noise
- Random Notes

Validation

- ISE’s – Orca, Installshield, or Tuner
  - Errors vs. Warnings vs. Info

- What you can ignore:
  - Error: Invalid file names
  - Warning: directory hardcoded to local drive (K:\)
  - Most Per-User vs. Per-machine warnings when doing GPO assigned apps
  - Also see Page 144 of Adminstudio book.

Testing

- install via double-click/script
- install via GPO
- uninstall via script/GPO
- 1st launch as "user“
- where does it save files?
- disconnected operation
- vista? x64? plain xp?
- Make sure to test w/o alwaysinstallelevated
- Does the second user get the same user experience as the first?

Types of Installs

- Multiple MSI’s can be better than a single large one
- Snapshotting MSI’s – Don’t Do it!
  - However, once in a great while is needed if vendor MSI is absolutely terrible, but cleaning process is much harder and errors have greater impact.
- Installation Monitoring > Snapshot most of the time.
- Scripted installs
  - Fine for lab-like environments where a reinstall is common
  - Good for permanent prerequisites
  - Bad for “user” machines where upgrades are necessary

Transforming Vendor MSI’s

- Don’t edit the vendor MSI unless you have to.
- Many vendors provide tools for creating transforms (Office, Autocad, Solidworks).
- Don’t include the licensing information if it is subject to change as a recache and reinstall of the MSI is required to change

Right Tool for the Job

- Tools I use: Package with Repackager then edit with Installshield, Orca
- Packagers
  - Installshield AdminStudio
  - Orca – MSI DB Editor
  - WiX – Scriptable interface for creating MSI's (also, MAKEMSI)
  - MSI Wrappers (ex: Windows Installer Wrapper Wizard)
  - Winstall LE (see TugZip.msi to see how bad it can be)
- Deployment
  - GPO
  - Zenworks
  - Scripts

Tools from Microsoft

- Windows SDK Components for Windows Installer Developers - http://msdn.microsoft.com/en-us/library/aa370834(VS.85).aspx
  - Orca – MSI Direct DB Edit
  - MSIZap – Removes MSI information for 1 or all products
  - Wilogutl – MSI Log Analyzer
  - Msimsp – Create Patches (.msp)
  - Msitran – Create Transforms (.mst)
- Platform SDK – Scripts in samples/sysmgmt/msi/Scripts
  - WiDiffDB.vbs – Diff between 2 MSI's
  - WiRunSQL.vbs – Run SQL statements against MSI DB
  - WiStream.vbs – embed streams (ex: .cab file) in MSI
- Microsoft Cabinet Software Development Kit - http://support.microsoft.com/kb/310618
- MSI Cleanup Utility - http://support.microsoft.com/kb/290301

Installshield AdminStudio

- Different Versions
  - http://www.acresso.com/products/licensing/adminstudio.htm?link_id=rightnav
  - Standard – No custom action/transform editor, greatly reduced MSI testing
  - Professional – Better Testing (user permissions, etc), Installscript -> MSI, vbs custom actions
  - Enterprise – Citrix support, Central App catalog, automated testing
- Repackager
- What is Installshield bad at?

Repackager

- When do you use the installation monitoring vs. snapshotting?
  - Installation Monitoring works well when there is a single installer
  - Snapshot works well for apps that have a bunch of chained installers (ArcGIS or Office), have no executable installer (WinSCP, Eclipse, VPython), or does install-time compiling (Adams)
  - Always exclude reg keys/directories at the highest level (ex: HKCU or WindowsFolder)
  - You still have to clean the msi even with Installation monitoring
- What is the repackager bad at?
  - Snapshotting certain huge apps -> non-linear growth of time to construct file/component tables
  - Adding to the path – it tends to add everything in the path to the path
  - Multiple snapshots – You have to clear out snapshot directories in order to do a follow-on snapshot

InstallScript

- Repackager can do “Installscript Scan” which turns an isscript MSI into a normal MSI.
  - Only works with Installation Monitoring or Single-Step Snapshot
- isscript.msi – Same Product Code for all versions. Do not ever assign via GPO.
- If Installscript Scan doesn’t work:
  - Property ISSETUPDRIVEN=1
  - InstallExecuteSequence OnCheckSilentInstall, set Condition=0

What is Installshield bad at?

- Direct DB editor is flaky on large apps
- Shortcuts are almost always advertised
- Tweaking the installation conditions of vendor provided MSI's is difficult
- If the installer used registry keys to set file associations, it does not consolidate them into the appriate table (extension, MIME, etc).
- Does not reassign the Keypath for a directory if the File initially used is deleted from MSI.
- Figuring out the parent feature when adding new files

Orca

- Direct Database Editor
- When to use?
  - Removing launch conditions
  - Un-advertising shortcuts
  - Changing install levels for features
  - Looking at transforms
  - Searching the Database
  - Validation
- Issues:
  - Crashes on Vista
  - Save Transformed as… drops any streamed .cab files

How to have a cleaner snapshot?

- Put the CD in ahead of time
- Do a test install so you know about any surprises
- Look on CD for prereqs and install them first
- Delete any uninstall shortcuts/reg keys
- Do you do a test launch in the snapshot?
  - Sorry, it depends on the app
- Windows updates, Installshield updates
- Reboot right before beginning the snapshot

- For the love of God: Apps go under C:\Program Files

Cleaning your MSI

- Reboot Example!
  - Start Snappshot, Reboot, Close Snapshot
  - Remember to Update exclusion lists!!!
  - Keeping the Exclusion editor updated can save lots of time.
- HKCU\Software\Microsoft\Explorer <- MRU
- HKLM\Services\ <- mostly reboot trash
- HKLM\Software\Microsoft\Cryptography\RNG
- DHCP Renews/Firewall Epoch
- Log Files

Fixing Things

- PATH
- De-Advertising Shortcuts
- Swapping out the Username
- UI
  - Desktop Icons usually are bad
  - Default save location should be under user profile (use transforms for network location)

Random Notes

- Save the project before hitting “Build” when using Repackager to create an msi.
  - Allows the ability to go back and choose building an isolated msi
  - Sometimes you can run out of space/memory when building a large MSI and the repackager might crash
- If MSI or Transform references a network location via the DrLocator table (ex: to figure out if K:\ exists) the network location must be there during the assignment of MSI in GPO

Group Policy Object Best Practices

- One Application per GPO
- Filtering
- LUA
- PreReqs
- Group Policy Preferences
- Upgrades

Filtering

- Use Groups – Authenticated Users is bad
- WMI Filters
  - You can filter on way more than just the OS
  - http://techies.ncsu.edu/wiki/Group_Policy_WMI_Filters
- Link in at highest level for licensing/sanity/etc.

Limited User Account

- Setting Permissions via GPO
  - File
  - Registry
  - Services
- Tools for figuring it out
  - Process Monitor
  - LUA Buglight

Prerequisites

- Determine if the Prerequisite is tied only to the particular application.
  - If so, include it in the snapshot
  - If not, install it via “run once” style scripting
- Isscript.msi is terrible
- Do not include VBA, DirectX, MDAC, or anything else like them in a snapshot.

Group Policy Preferences

- What is it?
  - RSAT-Only
  - http://www.wolftech.ncsu.edu/support/support/Active_Directory/Documentation#Group_Policy_Preferences
- What you can use it for:
  - Licensing – no NCSU info in MSI at all, means no reinstalling msi to update licensing info
  - Add things to path based off group memberships
  - Set reg keys per OU
  - Modify licensing based off group memberships
  - Distribute stuff to K:\ w/o advertised shortcut (if K: is mapped prior to user GPO running)

Upgrades

- Options
  - Assign new MSI to same GPO
  - New GPO/group with upgrade set
  - Uninstall / Reinstall
- Uninstall when falls out of scope is very useful

Details on MSI Specifics

- Logging
- Tables:
  - Property
  - Sequencing
  - Feature{Component}
  - File, Directory, Registry, Shortcut, MIME, Extension
- KeyPaths
- Custom Actions
- Transforms

Logging

- System Key: [HKLM\SOFTWARE\Policies\Microsoft\Windows\Installer]
- Value Name: Logging
- Data Type: REG_SZ (String Value)
- voicewarmup = Verbose output, Out-of-disk-space messages, Status messages, Initial UI parameters, all Error messages, non-fatal Warnings, start up of Actions, action-specific Records, out-of-Memory or fatal exit information, User requests, terminal Properties

- Saves MSIXXXX.log in %TEMP%

Properties

- http://msdn.microsoft.com/en-us/library/aa370905(VS.85).aspx
- Property Table
  - ALLUSERS = 0 Per User; 1 Per Machine; 2 Try Per Machine first
  - ARP* - Configure Add/Remove Programs
  - INSTALLLEVEL – Controls which features get installed
  - REBOOT = Force/Suppress/ReallySuppress
  - TRANSFORMS = Path to transforms
  - REINSTALLMODE = omus (reinstall files, registry, and shortcuts); v – recache msi
  - SOURCELIST – Specify any number of network locations (only used for new installs/advertisements)
- User and Computer Variables
  - ComputerName, LogonUser, UserSID, AdminUser, System Folders

Codes

- GUIDs – Unique 8-4-4-4-12 number in UPPERCASE hexidecimal {XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}
- GUIDGEN – Windows App included in Visual Studio to generate GUIDs.
- Product Code
  - GUID that specifies the Product. Major versions should change Product Code. In Property Table.
- Package Code
  - GUID that specifies the Package. Any change to Package should change the Package Code.
- Upgrade Code
  - GUID that specifies the Product line. All Packages/Products for a given application should have the same Upgrade Code. In Upgrade Table. May have multiple Upgrade Codes.

- TIP!
  - Upgrading an Application via Group Policy assignment requires that the upgrade code for the old application be in the Upgrade Table for the new application’s msi. The old app will be uninstalled and the new one installed at next reboot.
- Trick!
  - You can use this for “replacement” of an application rather than just upgrades. By adding the upgrade code from SpyBot to a new Ad-Aware msi, you can “upgrade” it.

Sequencing

- {Admin, Advt, Install}{Execute, UI}Sequence
  - Admin – Used when doing admin install of the msi
  - Advt – Used when advertising the msi
  - Install – Used when installing the msi
  - Execute – Required Actions
  - UI – Interactive-only Actions
  - AdvtUISequence is not valid
- Conditions – Supports <, >, <=, >=, =, <>, NOT, AND, OR, XOR http://msdn.microsoft.com/en-us/library/aa368012(VS.85).aspx
  - Can use Properties, Env. Variables, Feature/Components
- Sequence Number – Executed in order. Negative only execute if terminated. Nulls are never executed.

MSI Construction

- Feature – Logical portion of the product
- Component – Atomic piece of the installer (a file, a directory, reg keys from a single hive, a mime type registration, etc.)
- FeatureComponent – Maps atomic pieces to their feature
- Attributes – bitmask that controls which whether atomic unit is favored local, favored source, shared, permanent
- Conditions

- Notes
  - Features have a parent/child relationships
  - A component can be mapped to multiple features

KeyPaths

- Keypath is what Windows Installer looks at to see if a repair is required. A component keypath can be a directory, a file or a registry value.

- Do not use KeyPaths that are subject to change
  - Can cause unintended repairs (as in every time)
  - Can backrev files/reg keys that are supposed to be updated

Custom Actions

- Types of Custom Actions
  - http://msdn.microsoft.com/en-us/library/aa368066(VS.85).aspx
- Common types:
  - Type 6 – vbscript embedded in a stream
  - Type 35 – Directory set with formatted text
  - Type 51 – Set Property with formatted text
- Cannot use the wscript object with Type 6 Custom Action
- Examples of Type 6 Scripts – Firefox 2.0.0.6
  - Distribute files to network drive at first launch
  - Write out a config file w/ user-specific information

Transforms

- The Good
  - Change default save locations
  - Multiple configurations for common app
  - Many vendors provide tools to generate them
  - Can have Binary streams
- The Bad – Confusing for Interactive Installations
- The Ugly
  - Cannot have embedded Media in the transform
  - Cannot override product/package codes
- Tuner – Response Transforms!
- Orca – Better for viewing transforms
  - Save Transformed as… drops any streamed .cab files

Prefixes

- Transforms
  - : - transform is embedded in msi
  - @ - look at same folder as msi for transforms
- [] – Properties in File/Directory/Registry/Environment/Shortcut Tables
- Registry
  - #x - REG_BINARY
  - #% - REG_EXPAND_SZ
  - # - REG_DWORD
  - [~] - REG_MULTI_SZ and append/prepend
- Environment
  - * - Denotes System Env variable
  - [~] – Append/Prepend (ex: [~];c:\syb12\bin)
  - Removal: !, on install; -, on uninstall; !- on {un}install

Advertisement

- Using Group Policy:
  - Add/Remove Programs – Install on demand
  - User Assignment of Software Packages
  - This doesn't end up working too good… ☹
- Advertised Shortcuts
  - Benefits: Per-User settings, launch scripting via Custom Actions
  - Issues: "Launch conditions" includes all types of installs
- Recommendation:
  - Only use Advertised Shortcuts when you need them.
  - Note: Installshield will automatically make most shortcuts advertised when creating the MSI via snapshot.

Self-Repair

- The Good
  - Looks for the KeyPath of a Component in the MSI, and if its not there or different, it repairs. Includes loading COM objects.
- The Bad
  - Unintentionally Forcing Registry Values
    - Do not use KeyPaths that are subject to change
  - Chained Product Repair Looping
    - http://msdn.microsoft.com/en-us/library/aa371546(VS.85).aspx
    - Often Caused by snapshotting multiple MSI's that install the same files, usually a prerequisite (VBA, MDAC)

Putting it all together

- Goal: Create a Thunderbird MSI with NCSU-specific configuration that deploys via GPO
  - Capture using Repackager\Installation Monitoring
  - Clean it using Repackager and Installshield Editor
  - Add type 6 Custom Action (vbscript) via Installshield Editor or Orca
    - create the profiles.ini, prefs.js, bookmarks.html
    - under AppDataFolder
    - In Binary table
  - Entry in InstallExecuteSequence table with a
    condition of UserSID <> “S-1-5-18”
  - Make the Advertised Shortcut use a per-user keypath
  - Make a Transform if you need it to go elsewhere

Questions!

- How do you handle non-roaming environment w/ profile on the network?
  - Script knows not to overwrite or file-based keypath
- Which tables do you edit to add the custom action?
  - Binary, CustomAction, InstallExecuteSequence

Websites

- AppDeploy - http://www.appdeploy.com/
- Sysinternals - http://technet.microsoft.com/en-us/sysinternals/
- Eval Admin Studio - http://www.acresso.com/downloads/downloads_4886.htm
- LUA Buglight - http://blogs.msdn.com/aaron_margosis/archive/2006/08/07/LuaBuglight.aspx
- MSI Examples - http://msdn.microsoft.com/en-us/library/aa372837(VS.85).aspx
- WiX - http://wix.sourceforge.net/
- WiX Tutorial - http://www.tramontana.co.hu/wix/
- MAKEMSI - http://dennisbareis.com/makemsi.htm
- MSI Basics - http://makemsi-manual.dennisbareis.com/windows_installer_basics.htm
- MSI Reference - http://msdn.microsoft.com/en-us/library/aa372860(VS.85).aspx
- Installsite - http://www.installsite.org/
- Altiris Packaging site - http://juice.altiris.com/packaging
- MSI Blogger - http://msiblogger.com/

Questions?