

Agenda

- Overview
- Using Package Manager
- Package Manager Demo
- Call To Action
- Questions

Please hold questions until the end!

Image Servicing & Customization

Overview Goals

- Understand Windows Vista Servicing Technology and Terminology
- Understand how Package Manager is used
- Understand how to install:
 - Updates
 - Drivers
 - Language Packs
 - Optional Windows Features

Image Servicing & Customization

Definition of Terms

- Online
 - Operations on a running operating system.
- Offline
 - Operations on a Windows image that is not booted.
- Downlevel
 - Any Microsoft operating system prior to Windows Vista
- Mounted Image
 - An OS image that is not yet installed, often a “master installation”
- Applied Image
 - An OS that is already installed
- Host
 - The OS that you are booted in to, often a “technician computer”
- Target
 - The OS installation that you are modifying.

Image Servicing & Customization

More Terms Defined

- Servicing
 - Installing security updates or other Updates onto an operating system
- Customization
 - Configuring a system as permitted by your license agreement
- Windows Vista Servicing Stack
 - Is an OS Service also used by setup
 - How the OS installs, configures, and removes software packages.
 - Usable on downlevel operating systems, when manipulating Windows Vista images

Image Servicing & Customization

What is a Package?

- A package is a collection of software updates
 - Security and other OS updates
 - Optional Windows features
 - Inbox Drivers
 - Language Packs
- Released as a compressed .CAB
- Signed by Microsoft

Image Servicing & Customization

Windows XP vs. Windows Vista

- Windows XP used Update.exe
 - Update installer was in each update
- Windows Vista has a “Servicing Stack”
 - Installer is native to Windows Vista
 - Updates are now payload only

Image Servicing & Customization

Overview: Servicing Stack and Package Manager

- Servicing Stack Operations
 - Install and Uninstall packages
 - Configure settings in packages
 - List packages
- Key clients of the Servicing Stack:
 - Add/Remove Programs
 - Windows Update
 - Package Manager (Pkgmgr.exe)

Image Servicing & Customization

Overview: Online Servicing

- The running operating system is the one being serviced
- Examples of online servicing
 - End users downloading and installing updates from the Windows Update web site
 - Installing updates from an OEM Supplemental CD during Audit mode

The focus of this presentation is on Offline Servicing

Image Servicing & Customization

Overview: Offline Servicing

- Installing or configuring packages on an offline system
- Can operate against:
 - Applied image
 - Mounted image
- Uses the Host Servicing Stack
- Preferred Host platform is Windows PE 2.0

Image Servicing & Customization

Overview: Running the Servicing Stack downlevel

- Supported platforms for technician computer:
 - Windows XP SP2
 - Windows Server 2003 SP1
 - Windows Vista
 - Windows PE (1.6 and 2.0)
- Installed as part of the Windows Vista OPK
 - The folder containing the Servicing Stack can be copied to other systems

Image Servicing & Customization

Overview: Optional Windows Features

- The Servicing Stack is also used when:
 - Installing features during Windows Vista Setup
 - Turning features on/off when building custom deployments for corporate customers
 - Online: Add/Remove Programs
 - Offline: Package Manager (Pkgmgr.exe)

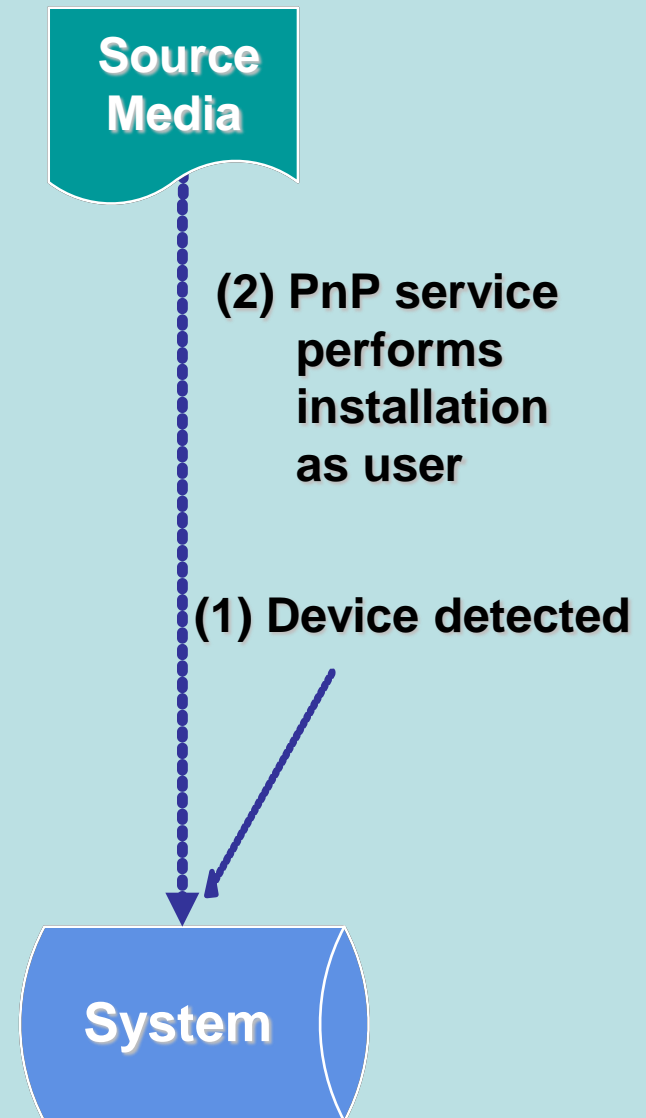
Image Servicing & Customization

Overview: Language Packs

- Language Packs (LP) are:
 - Update packages
 - A Collection of language resources for Windows features
 - Edition Neutral
 - Installed by Servicing Stack
 - Online: Language Pack Setup (Control Panel)
 - Offline: Package Manager (Pkgmgr.exe)
 - Offline installation requires intlcfg.exe to complete configuration

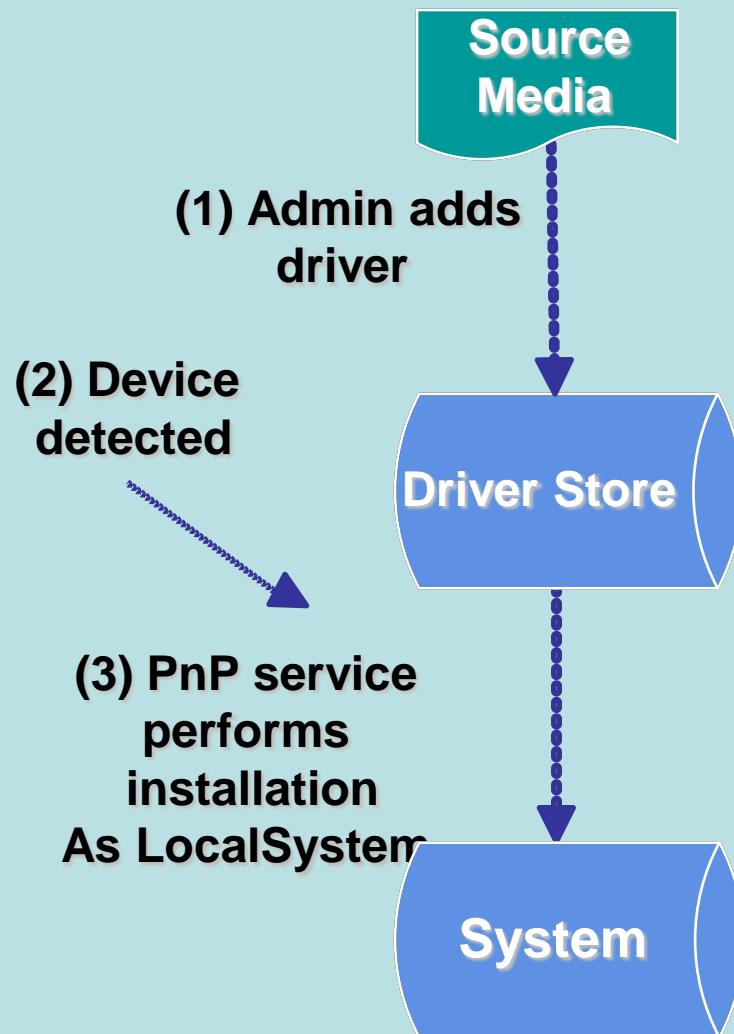
Windows XP Driver Installation

- Windows XP
 - PnP installs run in user's context
 - User can directly modify system state



Windows Vista Driver Installation

- Windows Vista
 - PnP runs in LocalSystem context
 - User cannot directly modify system state
 - Drivers are staged in a “Driver Store”



Third-Party Drivers and Package Manager

- Package Manager injects third-party drivers into the Driver Store
- Third-party drivers do not need to conform to regular packaging format
- PnP ranking chooses the best driver match based on the best fit for the hardware, regardless of whether the driver is inbox or third-party

Image Servicing & Customization

Overview Summary

- Servicing stack vs Update.exe
- Package Manager installs and configures packages on an offline image
- One technology for all packages:
 - Update Packages, Windows Features, Language Packs, and Inbox Drivers
- Drivers
 - Staged in Driver Store
 - Package Manager can inject third-party drivers into Driver Store

Key Facts about Package Manager

- Pkgmgr.exe is a Windows application with minimal user interface and runs in the background
- Requirements
 - Mounted or Applied Image to modify
- Best Practices
 - “start /w pkgmgr /o:“c:\;c:\Windows” /n:unattend.xml
 - “echo %errorlevel%”
 - Don’t run simultaneous instances of pkgmgr.exe

Offline Details

- Package Manager:
 - Does not mount or apply the image
 - Cannot modify remote images across the network; images must be on the local system.
- Some packages may require a reboot to complete configuration.
- All critical update actions occur before system boot
- Offline servicing is platform independent
 - A 32-bit technician computer can be used to service an offline 64-bit mounted image

Key Command-Line Options of pkgmgr.exe

- **/o** – “Offline”

- Syntax: **/o:***“BootDrive;PathToWindows”*

- **/n** – “uNattend”

- Example:

```
start /w pkgmgr /o:“c:\;c:\Windows”  
/n:unattend.xml
```

- Gotchas

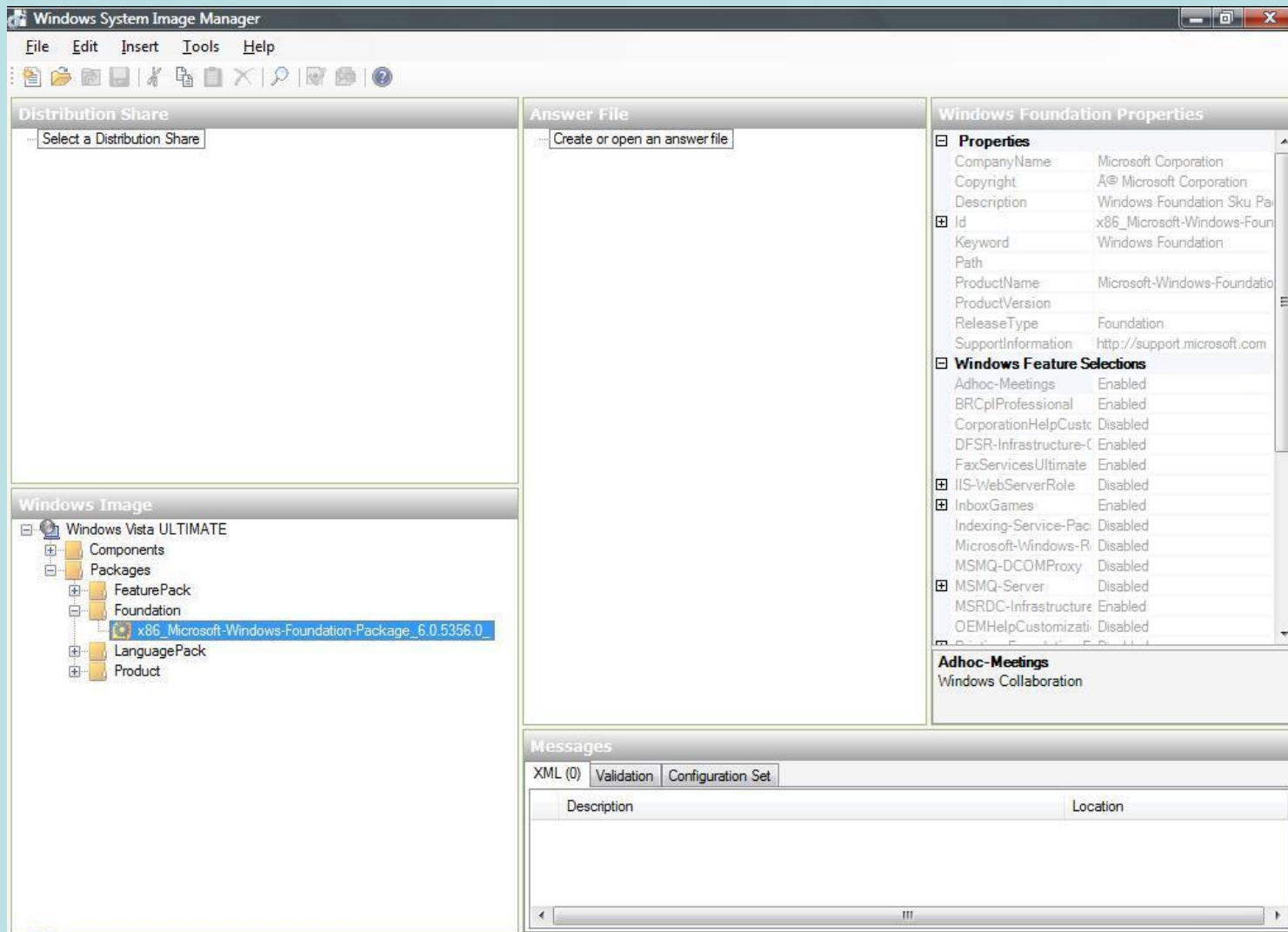
- The value of *BootDrive* must end with “\”

- Do not put a “\” after *PathToWindows*

Using Unattend.xml with Package Manager

- Package Manager uses the <servicing> section
- Everything you can do manually you can do using Unattend.xml
- Use Windows SIM to create and validate the syntax of the Unattend.xml file.

Configuring Optional Components



See “Summary of Allowable Customizations” in the OPK User’s Guide

Diagnosing Failures

- Package Manager automatically logs key actions in plain-text in %windir%\logs\cbs\cbs.log on the technician computer
 - Same format as the setupact.log created during OS setup
- Use the /l switch to create a binary trace log when creating an incident report with Product Support
- We are actively improving the scope and content of the logs

Driver Injection

- Use Unattend.xml to point to your driver directory location
- All drivers in that location are imported into the driver store
- Boot critical drivers are “reflected” or installed so that the image can boot.
- Other drivers are installed by PnP after the installation is booted and online
 - Driver Ranking determines best match

Example: Driver Install in Unattend.xml

The screenshot displays the Windows System Image Manager interface. The main window is titled "Untitled* - Windows System Image Manager" and contains several panes:

- Distribution Share:** A dropdown menu with the text "Select a Distribution Share".
- Answer File:** A tree view showing the configuration structure. The selected path is "x86_Microsoft-Windows-PnpCustomizationsNonWinPE" > "DriverPaths" > "PathAndCredentials[Key='VideoDrivers']".
- Properties:** A table showing the properties for the selected element. The "Path" property is highlighted.
- Windows Image:** A list of components for the selected architecture (x86), including various Microsoft-Windows-* components and "DriverPaths".
- Messages:** A pane at the bottom showing the status of the configuration set, with "Configuration Set (0)" selected.

The "PathAndCredentials[Key='VideoDrivers'] Properties" pane contains the following information:

Properties	
AppliedConfigurationPass	2.offlineServicing
Component	Microsoft-Windows-PnpCustomizationsNonWinPE
KeyName	Key
Path	DriverPaths/PathAndCredentials[@keyValue="VideoD

Settings	
Action	Default Action
Key	VideoDrivers
Path	\\myserver\share\videodivers

Path	

The "Messages" pane at the bottom shows the following table:

Description	Location

To confirm that drivers were added successfully, look under %windir%\inf for oem*.inf files

Call to Action

- Use Unattend.xml to configure component settings
 - Windows SIM makes this easier
- Use “start /w” and “%errorlevel%” when using pkgmgr.exe
- Installing a .CAB is the same for both update packages and language packs

Questions?