



Extending Microsoft System Center Configuration Manager with Windows Embedded Device Manager 2011

Microsoft Corporation
February 2011

Windows® Embedded Device Manager 2011

Windows Embedded Device Manager 2011 extends the capabilities of Microsoft® System Center Configuration Manager to enable the centralized management of Windows Embedded devices such as thin client, point-of-service (POS), and digital signage devices.

Embedded devices are typically task-specific and come with unique deployment, maintenance, and service challenges. Embedded Device Manager provides an extensible management solution for embedded devices that enables embedded device collections, the deployment of packages to write filter-protected devices, and the imaging of embedded devices using OEM device imaging solutions.

This white paper introduces Embedded Device Manager and highlights some of its key features.

Contents

Introduction	333
What Is Embedded Device Manager?	333
Why Use Embedded Device Manager?	666
Common Usage Scenarios	776
What Comes with Embedded Device Manager?	777
Conclusion	999
Additional Resources	101010



Introduction

Embedded devices, such as thin clients, point-of-service (POS), and digital signage devices, are typically task-specific and come with unique deployment, maintenance, and service challenges.

Windows® Embedded Device Manager 2011 inventories embedded devices into embedded device collections unique to the embedded device type. Embedded Device Manager deploys packages to write filter-protected embedded devices that use the Write Filter management feature, and leverages OEM device imaging solutions to image embedded devices.

This white paper covers the following topics:

- What is Embedded Device Manager?
- Why use Embedded Device Manager?
- Common usage scenarios for Embedded Device Manager.
- What comes with Embedded Device Manager?

What Is Embedded Device Manager?

Embedded Device manager extends Microsoft® System Center Configuration Manger to manage embedded devices. Some of the key features include:

- Inventorying embedded devices into unique embedded device collections.
- Deploying packages to write filter-protected embedded devices.
- Imaging embedded devices using OEM device imaging solutions.

Embedded Device Manager inventories embedded devices into device collections unique to each embedded device type. Thin clients appear in an **All Thin Client Devices** collection, POS devices appear in an **All Point of Service Devices** collection, and digital signs appear in an **All Digital Signage Devices** collection. Packages intended for embedded devices can be deployed specifically to these embedded device collections.



The inventorying process of adding embedded device collections to the Configuration Manager console is shown in the following figure.

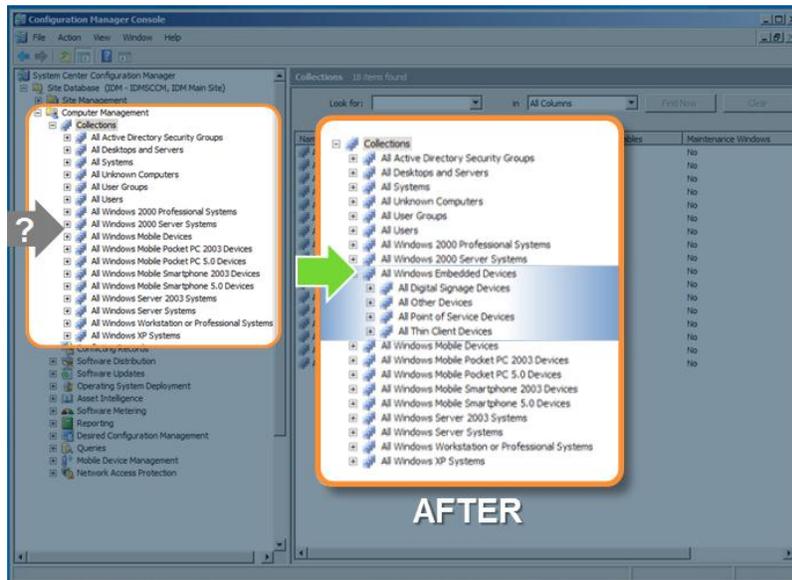


Figure 1. Adding embedded device collections to the Configuration Manager console

After the embedded device collections are added, an administrator can advertise and deploy software packages and configuration updates to embedded device collections with write filters programmatically managed as part of the process. The administrator needs to first confirm that write filters will be temporarily disabled and that the package can be processed in unattended mode. Then, write filters are turned off, allowing the package to be written to permanent storage media. The package is then processed on the embedded device. After the packages run, the write filters are then turned on, sending all writes intended to permanent media to volatile media instead, such as a RAM disk.

Simply put, the package deployment process occurs in three steps:

1. The write filters used on embedded devices are turned off.
2. The packages are deployed and run on the embedded devices.
3. The write filters are turned back on.

The package deployment process is shown in the following figure.

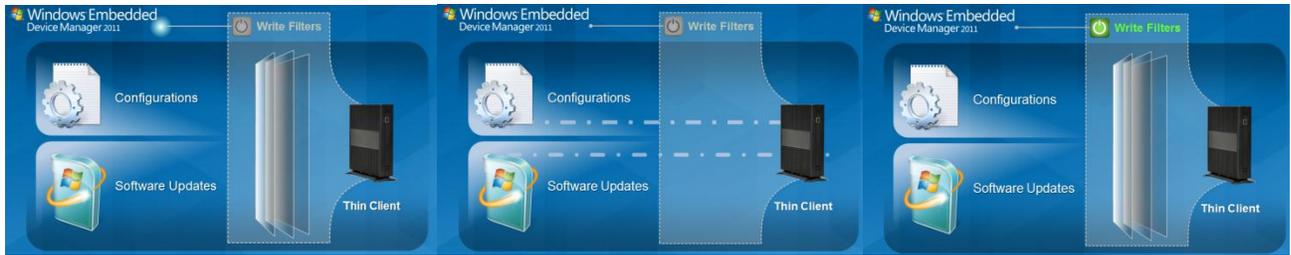


Figure 2. The package deployment process

OEM manufacturers can use their own image deployment technologies outside of Windows Deployment Services (WDS) and Preboot Execution Environments (PXE). Embedded Device Manager enables you to take advantage of an OEM device imaging solution to image the OEM’s embedded devices. The OEM device imaging solution process is shown in the following figure.



Figure 3. The OEM device imaging solution process

Embedded Device Manager device imaging requires:

- **An OEM device imaging solution.** A software solution created by the OEM device manufacturer that handles requests to deploy operating system images onto devices.
- **The device imaging component.** A software plug-in created by the OEM device manufacturer that implements a Component Object Model (COM) or Distributed Component Object Model (DCOM) interface to the Embedded Device Manager Device Imaging service. It also handles communication between the Device Imaging service and the device imaging solution. Embedded Device Manager provides a software development kit (SDK) to the OEM to create this COM/DCOM plug-in.

For more information about the Embedded Device Manager Device Imaging service, see the Embedded Device Manager SDK.

- **The Device Imaging service.** The Embedded Device Manager service manages communication between the device imaging component and Embedded Device Manager.



Why Use Embedded Device Manager?

The advantages in using Embedded Device Manager 2011 can be listed in the form of challenges and opportunities:

- *Challenge:* Previously, embedded devices appeared as personal computers in System Center Configuration Manager, but you could not manage them like personal computers. Packages deployed to personal computers also deployed to embedded devices unintentionally. This is a problem as a package intended for a personal computer is often not applicable to an embedded device and may even cause the embedded device to malfunction or stop working altogether.
- ✓ *Opportunity:* Embedded Device Manager now inventories embedded devices into device collections that are unique to each embedded device type. Thin clients appear in an **All Thin Client Devices** collection, POS devices appear in an **All Point of Service Devices** collection, and digital signs appear in an **All Digital Signage Devices** collection. Packages intended for embedded devices can be deployed specifically to these embedded device collections.
- *Challenge:* Previously, it was challenging to deploy packages to a write filter-protected embedded device. Changes to a write filter-protected embedded device in operation were cleared entirely from the device after a restart, including packages deployed from System Center Configuration Manager.
- ✓ *Opportunity:* Embedded Device Manager can programmatically manage write filters as part of the package deployment process. When packages are deployed, write filters are turned off and then turned back on again after the package has been received and processed on the embedded device.
- *Challenge:* Some embedded devices do not support image deployment technologies such as WDS and PXE. Imaging these devices usually meant using physical media, such as a CD, a DVD, or a USB key.
- ✓ *Opportunity:* Embedded Device Manager takes advantage of OEM device imaging solutions to image embedded devices. An Embedded Device Manager SDK is available to OEMs to provide guidance toward achieving this result. The OEM uses the SDK to create a plug-in for the Embedded Device Manager server. The plug-in then facilitates communication from Embedded Device Manager server to the OEM device imaging solution to image the embedded device.



Common Usage Scenarios

Embedded Device Manager can be used in a variety of scenarios. Common scenarios include the following:

- A service business manages and monitors kiosks deployed to numerous locations.
 - Reporting templates installed aid in reducing downtime.
- Businesses, municipalities, and schools manage collections of thin clients assigned to varying user needs.
 - Subcollections of embedded devices can be created as needed.
- A retail business manages collections of POS terminals.
 - Network and password settings can be tightly controlled.
- A hospitality business that has deployed digital signs throughout its entire chain.
 - Power and energy management settings can be preconfigured.

What Comes with Embedded Device Manager?

Embedded Device Manager provides both server and client installers.

The server installer includes:

- A setup wizard.
- A command-line setup option (EDM_Setup.msi).
- The following Embedded Device Manager Core Components:
 - Inventory. A .mof file that supports asset inventory.
 - Embedded Device Manager Server Core functionality. Provides basic Embedded Device Manager functionality to System Center Configuration Manager.
 - Common Service Host. A Windows® Communication Foundation generic service host that includes provisioning.

Note: The Core Components cannot be installed individually. They are all installed together.

- Configuration Manager Console Add-in. This component allows administrators to access and use Embedded Device Manager from the Configuration Manager console.
- Device Imaging service. This component supports the deployment of images, including operating system images, to embedded devices and interfaces with software provided by the device manufacturer.

The client installer includes:

- A setup wizard.
- A command-line setup option (Edmclient.msi) that must be distributed to all embedded devices if used.



The following table lists which device categories can be managed by Embedded Device Manager, and which operating systems can be installed on those devices by the OEM.

Table 1: Device Categories and Required Operating Systems

Device category	Operating systems
Thin client	Windows® XP Embedded, Windows® Embedded Standard 2009, Windows® Embedded Standard 7
Point of service (POS)	Windows XP Embedded, Windows® Embedded for Point of Service, Windows® Embedded POSReady 2009
Digital signage	Windows Embedded Standard 2009, Windows Embedded Standard 7



Conclusion

Embedded Device Manager extends System Center Configuration Manager to manage embedded devices.

Embedded Device Manager provides an extensible management solution for embedded devices that enables embedded device collections, the deployment of packages to write filter-protected devices, and the imaging of embedded devices using OEM device imaging solutions.

For more information about Windows Embedded Device Manager 2011, see:

Windows Embedded web site: www.microsoft.com/windows/embedded/default.aspx



Additional Resources

Embedded Device Manager installation, deployment, and management is discussed in greater detail in the following white papers:

- "Manage Your Embedded Devices with Windows Embedded Device Manager 2011"
- "Deploy an Embedded Operating System Image to Your Device Using Windows Embedded Device Manager 2011"
- "Deploy Updates to Your Embedded Device Using Windows Embedded Device Manager 2011"
- "Configure Write Filters on Your Embedded Device Using Windows Embedded Device Manager 2011"
- "Integrate Your Embedded Device Imaging Solution Using Windows Embedded Device Manager 2011"

Other resources include:

- About Using the Configuration Manager 2007 Client with Write Filters for information about Enhanced Write Filter (volume-level) and File-Based Write Filter (file-level): <http://technet.microsoft.com/en-us/library/bb932155.aspx>
- Microsoft System Center Configuration Manager: www.microsoft.com/systemcenter/en/us/configuration-manager.aspx
- Using Managed Object Format (MOF) .mof files: <http://technet.microsoft.com/en-us/library/cc180827.aspx>
- Windows Embedded Products: <http://msdn.microsoft.com/en-us/windowseembedded/default.aspx>



Copyright:

This document is provided "as-is". Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it.

Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

This document is confidential and proprietary to Microsoft. It is disclosed and can be used only pursuant to a non-disclosure agreement.

© 2011 Microsoft Corporation. All rights reserved.

