

# Knowledgebase Article

---

## Introduction to Windows Management Instrumentation

© Copyright 2001-2012 EMCO Software

## Introduction to Windows Management Instrumentation (WMI)

Windows Management Instrumentation (WMI; formerly WBEM) and WMI extensions for Windows Driver Model (WDM) are the foundations for manageability in Windows XP /Windows 2000 and Windows Millennium Edition/Windows 98.

The purpose of WMI is to define a non-proprietary set of environment-independent specifications. These specifications allow management information to be shared between management applications that run in both similar and dissimilar operating system environments. WMI prescribes enterprise management standards and related technologies that work with existing management standards, such as Desktop Management Interface (DMI) and SNMP. WMI complements these other standards by providing a uniform model. This model represents the managed environment through which management data from any source can be accessed in a common way.

WMI uses the Common Information Model (CIM) and Internet protocols for providing management data. The DMTF will use current WMI technologies as it continues to develop its standards, including development of the CIM event model, query mechanisms to CIM, and XML encodings for CIM objects.

Even though interest in automating administrative tasks via scripting grows steadily, it will still be a while before Windows Scripting Host, Active Directory Services Interface, or Windows Management Instrumentation becomes common items in administrative toolkits on par with standard Administrative Tools included in Windows 2000 or XP.

Learning intricate object models used by ADSI or WMI is without doubt a challenging task, especially for those who lack prior programming or scripting experience. One of Microsoft's recent attempts in addressing this challenge is WMIC. This acronym stands for Windows Management Instrumentation Command Line Tool, which is a component available in Windows XP and the soon-to-be released Windows .NET server platform.

WMIC offers a friendlier, non-scripted way of using Windows Management Instrumentation for managing computing environment.

WMI not only allows you to remotely control services already installed locally or on remote servers and workstations, but it also provides capability to create or delete services. And unlike ADSI, WMI can be used to change passwords on service accounts (for services running under non-Local System accounts).

Microsoft is using the term Windows Management Instrumentation (WMI) to describe how Microsoft implements WBEM for Windows. It is fully compliant with the DMTF Common Information Model (CIM) and WBEM specifications. WMI is now used by Microsoft where the terms "WBEM" and "WBEM for Windows" were used previously, and WMI will remain conformant to future DMTF WBEM specifications.

The kernel side/device driver interface to WBEM, previously referred to as WMI, is now called the WMI extensions for WDM. Similarly, the provider that surfaces this device driver based information to the management infrastructure is now called the WDM provider instead of the WMI provider.

### **WMI and Microsoft Operating Systems:**

WMI provides fully integrated operating system support for uniform system and applications management, based on the Common Information Model (CIM) adopted by the DMTF. WMI provides a consistent and richly descriptive model of the configuration, status, and operational aspects of Windows operating systems, assisting management applications in creating solutions that reduce the maintenance and life cycle costs of managing Windows.

Used in conjunction with other management services provided in Windows 2000, such as the Microsoft Management Console (MMC), WMI helps simplify the task of developing well integrated management applications, allowing vendors to provide Windows customers with best-of-breed, enterprise-scalable management solutions. Local and remote events combined with a rich query language for the information model provide the means to create solutions to complex management problems. The ability to easily script these solutions in Visual Basic<sup>®</sup> or using Windows Scripting Host (WSH) adds an often-requested dimension to Windows management.

## **WMI and WMI Extensions for Windows Driver Model (WDM):**

WMI is the kernel-level instrumentation technology for Windows. Close coupling of WMI extensions for WDM with services developed to conform to the WMI initiative simplifies instrumentation and provides consistent, open access to management data.

### **WMI extensions for WDM facilitate these capabilities:**

- ▲ Publishing kernel instrumentation
- ▲ Configuring device settings
- ▲ Providing kernel-side event notification
- ▲ Publishing custom data
- ▲ Allowing administrators to set data security
- ▲ Accessing instrumentation by way of WMI

WMI and WMI extensions for WDM share a unified schema, with data made available to management applications by way of WMI. Instrumented hardware data is provided by way of drivers instrumented for WMI extensions for WDM. WMI extensions for WDM provide a set of Windows DDIs for instrumenting data within the driver models native to Windows, so OEMs and IHVs can easily extend the instrumented data set.

The Windows DDK includes WMI extensions for WDM documentation and sample code.

## Using the WMI Control

The WMI (Window Management Instrumentation) Control is a tool that lets you configure WMI settings on a remote computer or local computer

### You can use the WMI Control to:

- ✦ Set permissions for authorized users and groups
- ✦ Turn error logging on or off
- ✦ Back up the WMI repository
- ✦ Perform other configuration tasks.

WMI includes a CIM-compliant object repository, which is the database of object definitions, and the CIM Object Manager, which handles the collection and manipulation of objects in the repository and gathers information from the WMI providers. WMI providers act as intermediaries between components of the operating system and applications. For example, the registry provider draws information from the registry, while the SNMP provider provides data and events from SNMP devices.

In Windows 2000, several management tools have been WMI-enabled, including Logical Drives, System Properties, System Information, and the Dependencies component of Services. In addition, Windows 2000 provides a tool called the WMI Control that can be used to modify WMI configuration settings. These components are briefly described below:

- ✦ The WMI Control enables you to perform Windows Management configuration tasks, such as setting permissions for authorized users or groups, backing up the object repository, and turning error logging on / off.
- ✦ Logical Drives lets you manage mapped drives and local drives on a remote computer or a local computer. You can view drive properties, change drive labels, and configure security settings for drives.
- ✦ System Properties lets you view and change system properties on a local or remote computer. You can restart a remote computer to apply settings changes or to detect new hardware, view the computer name and domain information for other computers on your network, or change the settings for the virtual memory paging file on a computer that might run programs requiring a lot of memory. System Information collects and displays configuration information about your system. This is especially useful when troubleshooting your system with a support technician.
- ✦ Services help you manage the services on your computer. Services dependencies identify the services upon which the current service is dependent and the services that are dependent upon it.

## To open WMI Control

- ✦ To open the WMI Control console, click Start, click Run, type `wmingmt.msc`, and click OK.
- ✦ For information about using the WMI Control, click the Action menu in the console, and then click Help.