

# Windows Management Instrumentation - Win32 apps

By stevewhims

Archived: 2026-04-05 15:14:39 UTC

Windows Management Instrumentation (WMI) is the infrastructure for management data and operations on Windows-based operating systems. Although you can write WMI scripts or applications to automate administrative tasks on remote computers, WMI also supplies management data to other parts of the operating system and products. For example, System Center Operations Manager or [Windows Remote Management](#).

## Note

This documentation is for developers and IT administrators only. If you're an end-user who has experienced a WMI error message, search for the error code in [Microsoft Support](#). For more information about troubleshooting problems with WMI scripts and the WMI service, see [WMI isn't working](#).

WMI is fully supported by Microsoft. The latest version of administrative scripting and control is available through the Windows Management Infrastructure (MI). MI is fully compatible with previous versions of WMI, and provides a host of features and benefits that make designing and developing providers and clients easy. For more information, see [Windows Management Infrastructure](#).

## Where is WMI applicable?

Although system administrators can use WMI in all Windows-based applications, it's most useful in enterprise applications and administrative scripts. For more information about WMI, see [Further information for WMI](#).

## Developer audience

WMI is designed for programmers who create C, C++, and Visual Basic applications, or use a scripting language that has a Windows engine and handles Microsoft ActiveX objects. Although some familiarity with COM programming is helpful, it's not required. C++ developers can find examples for getting started at [Create a WMI application using C++](#).

To develop managed-code providers or applications in C# or Visual Basic using the .NET Framework, see [Use WMI with the .NET Framework](#).

Many administrators and IT professionals access WMI through Windows PowerShell. The `Get-WMI` cmdlet for PowerShell enables you to retrieve information for a local or remote WMI repository. As such, several WMI articles, especially [Create WMI clients](#), contain PowerShell examples. For more information about using PowerShell, see [PowerShell](#).

## Run-time requirements

For more information about which operating system is required to use a specific API element or WMI class, see the Requirements section of each article in the WMI documentation.

If an expected component appears to be missing, see [Operating system availability of WMI components](#).

To create scripts or applications for WMI, developers don't need to download or install a specific software development kit (SDK). However, certain WMI administrative tools might be useful to use. For more information, see [Downloads](#).

## In this section

Article	Description
<a href="#">About WMI</a>	General information about WMI.
<a href="#">Using WMI</a>	Information about how to develop applications to use WMI, which includes information about tools.
<a href="#">WMI reference</a>	Documentation about the WMI classes, WMI C++ classes, WMI COM API, scripting API, and other WMI reference material.
<a href="#">WMI glossary</a>	A list of WMI terms. Many of these terms are familiar to developers, but have new or altered definitions in the WMI environment.

---

Source: <https://msdn.microsoft.com/en-us/library/aa394582.aspx>