Nederland (Nederlands)   Aanmelden

Zoeken in Windows met Bing

| Home | Windows 10 | Windows 8.1 | Windows 7 | Previous Versions | Windows Phone | MDOP | Surface | Library | Forums |

OSVERSIONINFO structure

OSVERSIONINFO structure

Contains operating system version information. The information includes major and minor version numbers, a build number, a platform identifier, and descriptive text about the operating system. This structure is used with the **GetVersionEx** function.

To obtain additional version information, use the **OSVERSIONINFOEX** structure with **GetVersionEx** instead.

Syntax

```
typedef struct _OSVERSIONINFO {
  DWORD dwOSVersionInfoSize;
  DWORD dwMajorVersion;
  DWORD dwMinorVersion;
  DWORD dwBuildNumber;
  DWORD dwPlatformId;
  TCHAR szCSDVersion[128];
} OSVERSIONINFO;
```

Members

dwOSVersionInfoSize
The size of this data structure, in bytes. Set this member to `sizeof(OSVERSIONINFO)`.

dwMajorVersion
The major version number of the operating system. For more information, see Remarks.

dwMinorVersion
The minor version number of the operating system. For more information, see Remarks.

dwBuildNumber
The build number of the operating system.

dwPlatformId
The operating system platform. This member can be the following value.

Value	Meaning
VER_PLATFORM_WIN32_NT 2	The operating system is Windows 7, Windows Server 2008, Windows Vista, Windows Server 2003, Windows XP, or Windows 2000.

szCSDVersion
A null-terminated string, such as "Service Pack 3", that indicates the latest Service Pack installed on the system. If no Service Pack has been installed, the string is empty.

Remarks

Relying on version information is not the best way to test for a feature. Instead, refer to the documentation for the feature of interest. For more information on common techniques for feature detection, see Operating System Version.

If you must require a particular operating system, be sure to use it as a minimum supported version, rather than design the test for the one operating system. This way, your detection code will continue to work on future versions of Windows.

The following table summarizes the values returned by supported versions of Windows. Use the information in the column labeled "Other" to distinguish between operating systems with identical version numbers.

Operating system	Version number	dwMajorVersion	dwMinorVersion	Other
Windows 10	10.0*	10	0	OSVERSIONINFOEX.wProductType == VER_NT_WORKSTATION
Windows Server 2016 Technical Preview	10.0*	10	0	OSVERSIONINFOEX.wProductType != VER_NT_WORKSTATION
Windows 8.1	6.3*	6	3	OSVERSIONINFOEX.wProductType == VER_NT_WORKSTATION
Windows Server 2012 R2	6.3*	6	3	OSVERSIONINFOEX.wProductType != VER_NT_WORKSTATION
Windows 8	6.2	6	2	OSVERSIONINFOEX.wProductType == VER_NT_WORKSTATION
Windows Server 2012	6.2	6	2	OSVERSIONINFOEX.wProductType != VER_NT_WORKSTATION
Windows 7	6.1	6	1	OSVERSIONINFOEX.wProductType == VER_NT_WORKSTATION
Windows Server 2008 R2	6.1	6	1	OSVERSIONINFOEX.wProductType != VER_NT_WORKSTATION
Windows Server 2008	6.0	6	0	OSVERSIONINFOEX.wProductType != VER_NT_WORKSTATION
Windows Vista	6.0	6	0	OSVERSIONINFOEX.wProductType == VER_NT_WORKSTATION
Windows Server 2003 R2	5.2	5	2	GetSystemMetrics(SM_SERVERR2) != 0
Windows Server 2003	5.2	5	2	GetSystemMetrics(SM_SERVERR2) == 0
Windows XP	5.1	5	1	Not applicable
Windows 2000	5.0	5	0	Not applicable

* For applications that have been manifested for Windows 8.1 or Windows 10. Applications not manifested for Windows 8.1 or Windows 10 will return the Windows 8 OS version value (6.2). To manifest your applications for Windows 8.1 or Windows 10, refer to Targeting your application for Windows.

Examples

For an example, see Getting the System Version.

Requirements

| Minimum supported client | Windows 2000 Professional [desktop apps only] |

Minimum supported server	Windows 2000 Server [desktop apps only]
Header	Winnt.h (include Windows.h)
Unicode and ANSI names	**OSVERSIONINFOW** (Unicode) and **OSVERSIONINFOA** (ANSI)

See also

Version Helper APIs
OSVERSIONINFOEX