



Comparing Windows CE and Windows XP Embedded

by Mike Hall, technical product manager, Windows Embedded

Microsoft ships two embedded operating systems: Windows CE and Windows XP Embedded, both designed for use within embedded systems. So how do you determine which one may be appropriate for your device?

On initial inspection, Windows CE and Windows XP Embedded may seem to be similar, since both are componentized operating systems, both expose similar programming interfaces (Win32, MFC, ATL, and support for .NET applications), and both expose similar operating system technologies, which include support for networking, internet browsers, media players, and so on. The choice of operating system becomes easier, however, when you understand the design goals of each operating system.

Windows CE has been designed to be a small footprint, componentized, real-time operating system that runs on multiple processor architectures. Windows XP Embedded, on the other hand, can be considered to be a componentized version of Windows XP Professional Service Pack 2 with additional embedded-enabling features. Let's examine some of the features of both operating systems.

We already know that Windows XP Embedded is a componentized version of Windows XP Professional Service Pack 2; the operating system has been broken down into almost 12,000 individual components, approximately 9,000 device drivers, and 3,000 operating system technologies. You have the ability to pick and choose exactly which drivers, services, and applications are included in your final embedded operating system image. If, for example, you don't need a Media Player, Notepad, or Internet Explorer, then you don't need to include these features in your operating system image. Having the ability to pick individual components reduces the operating system size, and, of course, it also reduces the surface of attack from malicious code. While we're on the subject, Windows XP Embedded SP2 has all of the security updates found on Windows XP SP2. This includes having all ports on a system closed

by default (except HTTP). A system builder can, of course, decide to open the ports that are needed on his device. An average operating system on XP Embedded build is on the order of 40MB. This scales, depending on which operating system components/technologies you include in your final image.

One of the advantages of using Windows XP Embedded in an embedded system is the speed of development. Windows XP Embedded-based systems run on an x86 processor and PC Architecture hardware, both of which are very well understood. This means that any existing Windows 2000 or Windows XP device driver or application can run on Windows XP Embedded without modification. The Windows XP Embedded operating system ships as a set of pre-built desktop-compatible binary components, which ensures application and driver compatibility.

The typical development cycle for a Windows XP Embedded device is to first snapshot the underlying hardware using a tool called Target Analyzer, which generates an XML file that defines all of the hardware components found on your target device. The XML file can then be imported into the Component Designer tool and saved as a custom component that defines the hardware of your device. The Component Designer tool can also import .INF files, which makes it easy to generate components from existing Windows 2000 or Windows XP Device Drivers. Lastly, the Component Designer tool can also be used to create custom components, perhaps defining a component for a custom shell or application. The component defines the files, registry settings, and operating system dependencies for the component. Once components are defined, they can be checked into the Windows XP Embedded component database and then used in an embedded design.

The next step in developing your Windows XP Embedded operating system image is to use the Target Designer tool. This tool exposes all of the Windows XP operating system technologies and drivers and the Embedded Enabling Features (which we will discuss shortly). A system developer simply selects the components he needs and adds them to a project workspace. This can include the hardware definition component created by Target Analyzer, some design templates that define a starting point for a number of device categories, including Windows Based Terminal (Thin Client), Point of Sale device, Set Top Box, and so forth, and you also have the ability to pull individual components from the catalog to add to your project design. The Target Designer tool will run a dependency analysis on your project workspace at build time to ensure that all required operating system features are included in the final operating system image. An example of this would be a developer choosing the .NET Framework for his design, adding his .NET

application and then building the operating system image – the system designer doesn't need to know or understand the dependencies of the .NET Framework to build an operating system image.

We've already mentioned that Windows XP Embedded ships with some Embedded-specific features, including the ability to boot and run Windows XP Embedded from a CD-ROM, or to boot from Flash media. In both cases, a developer would want to treat the underlying media as Read-Only, since flash only supports a limited number of writes, and a CD-ROM would, in this boot environment, also be read-only. Windows XP Embedded ships with an Enhanced Write Filter component that intercepts operating system and application write instructions, and, instead of writing to the boot media, writes to an in-memory cache instead. This means that on power down, the underlying media cannot be corrupted, and the device always boots in a known good state – add to this support for resuming from a hibernation file multiple times, and you have a robust boot environment that provides consumer electronics level at boot time.

So how does Windows XP Embedded compare with Windows CE? Windows CE has been designed as a small footprint, hard real-time embedded operating system that runs on multiple processor architectures, which includes x86, MIPS, ARM, and SH4. A kernel-only build is approximately 200kb, and of course the operating system scales depending on the components you include in your design. A typical residential gateway configuration that supports 802.11a/b/g, remote administration through a web-based interface, and security would be less than 4MB in size. A fully configured Web Pad image that includes Web Browser, Media Player, Office File Viewers, support for the .NET Compact Framework, and Windows Explorer Shell would be closer to 18MB in size.

There are a number of core differences between Windows CE and Windows XP Embedded. First, Windows CE is a full 32-bit, Unicode operating system that doesn't support MS-DOS or Windows 3.x applications. Also, the Windows CE architecture is different from its desktop cousin. On the desktop you would expect to find three core operating system components: Kernel32, GDI32, and User32. None of these components exist on Windows CE. Their counterparts can be considered to be Coredll (kernel32), and GWES, the Graphics Windowing and Event Subsystem (you can think of this as being a combination of GDI32 and User32), which means that even simple desktop applications will not run on Windows CE. At a minimum, desktop applications would need to be recompiled to run on a Windows CE device, not only because of the architectural differences, but also because the Windows CE

device is more than likely running on a non-x86 processor.

You may think that Windows CE is limited to running only on handheld, battery powered devices such as Windows Mobile devices. While it's true that this is perhaps the most visible device shipping based on Windows CE, and that Windows CE does have a great power management story, that's certainly not the case. You can find Windows CE running on industrial control systems, medical monitoring devices, robots, retail point-of-sale devices, and many other categories of embedded systems. Windows CE has been designed to be a hard-real-time embedded operating system supporting interrupt latencies in the sub-ten-microsecond range. This is more than adequate for many real-time embedded designs.

Windows CE ships with a suite of tools that can be used to configure, build, and debug an embedded operating system image. The Platform Builder tool is in many ways similar to the Windows XP Embedded Target Designer tool. A developer has the ability to use a Platform Wizard (similar to the Windows XP Embedded design templates) to configure the initial operating system configuration, and he can then add or remove features from the operating system design. Platform Builder also contains support for operating system feature dependencies. For example, adding the .NET Compact Framework to a design will also add support for the required operating system technologies.

You typically find Windows CE running on custom hardware where the designer is looking to reduce the bill of materials by reducing the size of the operating system image, the power of the processor, and the supported peripherals, but at the same time include support for the latest operating system technologies. This process often requires that Windows CE is "ported" to new, custom hardware. To assist with this porting process, Windows CE ships with millions of lines of source code, which gives developers the ability to examine the way in which Windows CE works, and to perhaps modify the behavior of the source. Providing access to source also assists with the porting process.

Once a Windows CE operating system image is built, there are a number of tools that can be used to debug or test the Windows CE operating system image, including source code debuggers, performance monitors, and profiling tools.

We've briefly discussed some of the technical aspects of Windows CE and Windows XP Embedded, but it certainly doesn't end there. Microsoft provides evaluation versions of Windows CE and Windows XP Embedded as free

downloads at <http://www.windowembeddedkit.com> Licensing of the operating system images doesn't cut in until you actually start shipping devices, and Microsoft also supports the embedded operating systems for ten years! Perhaps it's time to start evaluating Windows CE or Windows XP Embedded for your next embedded design.

by Mike Hall, technical product manager, Windows Embedded

October 4, 2005