Embedded OSs Gain the Inside Track

Sixto Ortiz Jr.

The embedded operating system has become a key element that runs many systems essential to modern life. Embedded OSs are found in all kinds of devices and systems, from high-end routers and switches that keep networks running to medical devices that keep patients alive, as well as copiers, TV remote controls, factory-automation systems, and even "talking" dolls.

Just about any new system that must run automatically probably has an embedded OS orchestrating its components' performance.

The demand for embedded OSs is thus growing. As Figure 1 shows, Venture Development Corp., a market research firm, predicts that revenue from the sale of embedded OSs will more than double from $752 million this year to $1.59 billion in 2005.

The growing demand has made the market more attractive to new companies and established players. This, in turn, has made the market more wide open than the market for PC operating systems.

Microsoft and open-source vendors have also entered the market, joining established embedded-OS companies. And researchers are developing new architectures in an effort to improve OS performance.

NEW EMBEDDED-OS TRENDS

As embedded operating systems become a more important and lucrative product, more companies are entering the market and trying to gain a competitive advantage with their technology.

Marketplace battle

Unlike the Microsoft-dominated PC world, no single vendor dominates the

embedded-OS marketplace. The biggest player is WindRiver Systems, which has a 31.2 percent worldwide share of the market for embedded OSs and embedded-OS-development tools, such as compilers and debuggers, according to Daya Nadamuni, senior analyst for design and engineer software at market research firm Gartner Inc.

The marketplace is filled with various proprietary and open-source embedded operating systems. Vendors include Green Hills Software, Lineo, Microtronix, Monta Vista Software, OnCore Systems, QNX Software Systems, and Red Hat.

This competition will be healthier for the industry than having one dominant OS, said Jerry Krasner, chief analyst for American Technology International.

Some industry observers have expressed concern that the wide-open market will cause problems, such as product developers having to work with too many OSs. However, this shouldn't be a problem because eventually Linux and Windows will probably dominate the general market, said Barrie Sosinsky, chief analyst and founder of the Sosinsky Group, a market research firm.

Nadamuni said the large number of OSs is a sign of the market's immaturity. The situation will change, she added, as many smaller companies are either acquired or shut down.

Eventually, she predicted, "the industry will standardize around one or two OSs in each vertical market."

Open-source embedded OSs

A growing number of companies have released open-source embedded OSs, primarily those based on BSD Unix and Linux. These OSs have the same appeal that open-source software has in other markets: They are relatively inexpensive and users have access to and can customize the source code.

A recent survey of 500 embedded-systems developers by Evans Data, a software-development-market research firm, found that about 45 percent expect to release a Linux-based embedded system in the coming year and about 80 percent said Linux is important to their community.

Linux. Monta Vista Software President and CEO Jim Ready said Linux, which is the basis for his company's embedded OS, offers developers royalty-free licensing, more multivendor availability, and lower runtime costs than other embedded OSs.

Also, added Bill Weinberg, Monta Vista's director of strategy and evangelism, Linux "accelerates time-to-market by presenting standard, open, and ubiquitous APIs that streamline application porting and development."

Weinberg said key technical benefits include networking performance as well as robust, integrated memory management.

However, stated Dave Damast, operations manager for NetBSD vendor Wasabi Systems, Linux has real-time performance problems. Linux was designed to provide each task, regardless of priority, an equal share of processor time and other resources. Most embedded systems don't need hard real-time performance, though, and usually, Linux's capabilities are sufficient, Ready said.

Damast agreed that BSD and Linux are most often embedded in routers, switches, and other packet-based communications infrastructure devices, in which a few dropped packets won't cause problems.

And, Weinberg said, dedicated hardware or faster CPUs can often solve real-time requirements.

Damast also expressed reservations about Linux's GNU general public license. He said the GPL has not been tested in court, so some potential users fear its legal ramifications, as well as their ability to keep to themselves the additional code that they develop.

BSD Unix. Several types of BSD Unix, including FreeBSD and OpenBSD, have embedded versions. NetBSD is a particularly popular embedded OS.

According to Wasabi's Damast, NetBSD features a hardware abstraction layer that allows easy porting to new architectures. He said NetBSD also has effective networking capabilities because of its strong TCP/IP stack.

NetBSD requires from 400 Kbytes to 1.5 Mbytes of flash memory and from 2 to 16 Mbytes of RAM, depending on a version's functionality and whether it is compressed or includes network support.

Because of its connectivity-related capabilities, embedded NetBSD currently appears primarily in routers, telecommunications switches, and some Internet appliances but will appear in additional devices as they link to networks or the Internet, Damast predicted.

eCOS. Red Hat, a Linux vendor, built eCOS from the ground up as an open-source, real-time, non-Linux-based embedded OS, according to Chief Technology Officer Michael Tiemann. To address real-time needs, eCOS includes features such as interrupt and exception handling, schedulers, and timers.

eCOS can work with memory ranging from dozens to hundreds of Kbytes, which is less than Linux, whose memory footprint begins at 2 Mbytes. eCOS is used in such products as laser printers and audio players.

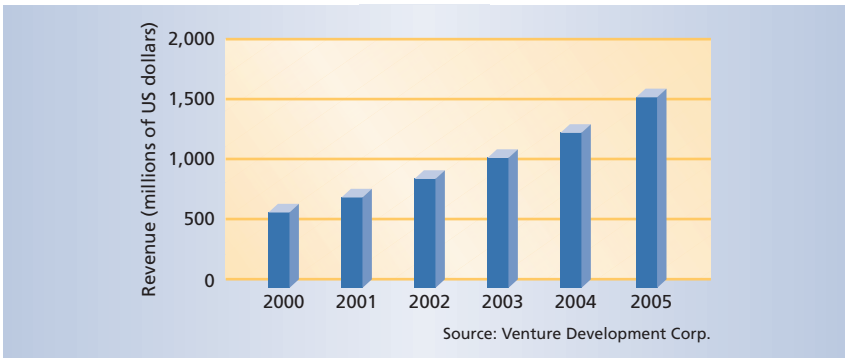

Figure 1. Venture Development Corp., a market research firm, predicts that sales of embedded operating systems will more than double between 2001 and 2005.

Inside the Embedded OS

Embedded operating systems consist of a kernel and ancillary modules, such as a real-time executive (which oversees OS process management and resource allocation), a scheduler, a file-management system, networking and communications protocols, and device drivers. The number of modules depends on the feature set the OS must support.

Embedded OSs that work autonomously, such as those in antilock braking systems, or those that monitor continuously running systems must operate without supplied system commands. These OSs have sensors that capture environmental stimuli that then generate interrupts. The interrupts cause the OS to react in a prescribed manner.

Embedded OS developers have traditionally faced several key challenges. For example, because the OSs operate in resource-constrained environments, they must work with limited memory, storage, and processing power.

Because they often perform specialized functions deep within systems, many embedded OSs must have small footprints and memory capacities. For example, some systems operate with as little as 50 to 100 Kbytes of memory, noted Daya Nadamuni, senior analyst for design and engineering software at Gartner Inc., a market research firm.

Embedded OSs must support multiple network and multiprocessor types, as well as a growing variety of increasingly complex applications with expanding feature sets. They also must work with growing system loads. For example, all embedded OSs running a high-end router must continue functioning as network traffic loads increase. Therefore, the operating systems must be scalable, flexible, and adaptable.

Many embedded OSs also must function under hard-real-time constraints, in which a system fails if it can't execute a critical task on time. In these cases, the OS must automatically prioritize critical tasks and determine when to execute them.

Embedded Windows

The embedded OS marketplace, with its lack of a dominant player, has caught Microsoft's eye. The company ships two embedded OSs, Windows CE Embedded and Windows NT Embedded, and plans to launch Windows XP Embedded in the near future.

Microsoft built Windows CE Embedded with real-time features, such as nested interrupt support, but didn't include such elements in Windows NT Embedded.

Windows CE Embedded works with x86 and non-x86 processors, including Hitachi Super-H, Motorola PowerPC, and Mips Technologies' PMC-Sierra chips, as well as those based on Arm Ltd.'s technology. Windows NT Embedded works with only x86 processors.

Windows CE has a small, 400-Kbyte footprint, and can be stored on a ROM chip. Windows NT Embedded, on the other hand, sports a large footprint for

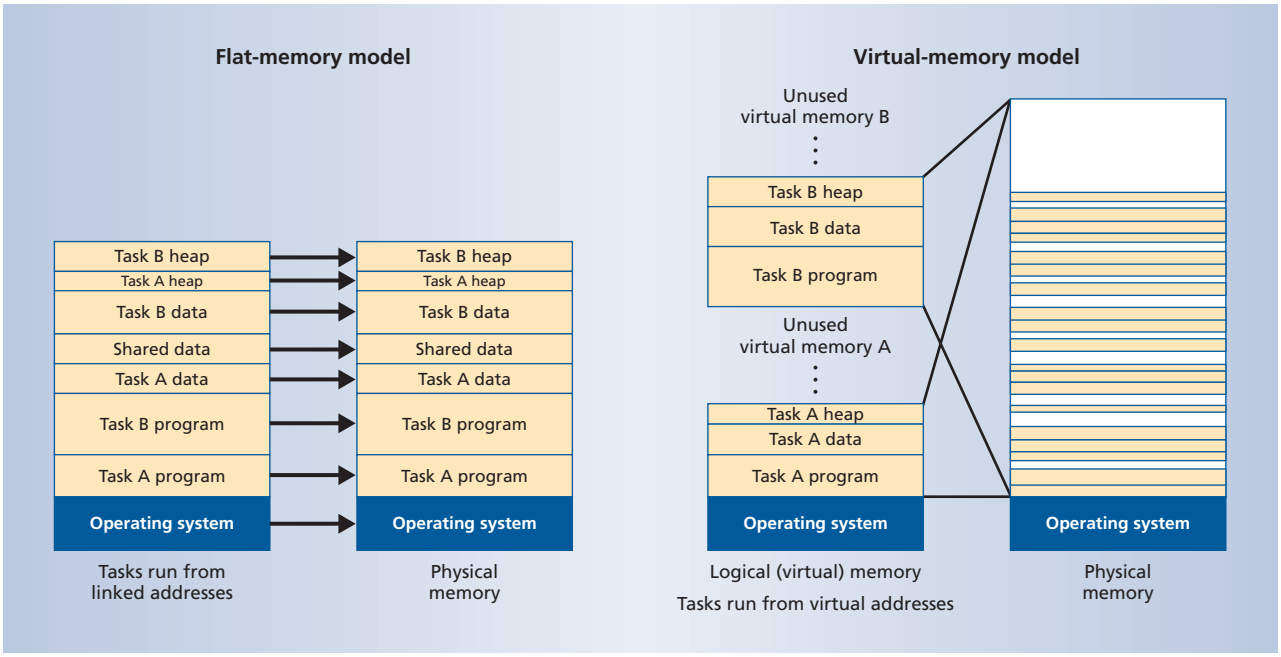

Figure 2. In the flat-memory model, all program memory is statically mapped one to one with physical memory. If two tasks accidentally write to the same memory address, the system could crash. In the virtual-memory model used by OnCore Systems' embedded OS, every task is in its own protected virtual-memory area, which can be mapped to locations within the full range of physical memory. A task cannot access areas outside its virtual-memory area and thus cannot crash the system.

an embedded OS. A minimal system with no network support requires 9 Mbytes of memory and 8 Mbytes of persistent storage. The addition of networking components and device drivers can push these requirements to 13 Mbytes of memory and 16 Mbytes of persistent storage. Therefore, NT Embedded is not ideal for very small embedded systems.

Microsoft promotes its embedded Windows products as something that will appeal to the many developers already familiar with the Windows API.

New architectures

While some vendors are using existing open-source and proprietary architectures for their embedded OSs, others are working on new architectures.

For example, OnCore Systems has developed an embedded real-time OS that uses a virtual-memory architecture.

For different threads, explained OnCore President and CEO Chip Downing, the virtual-memory model sets up independent, protected virtual-memory areas, each with access to the full range of physical memory addresses, as Figure 2 shows.

Traditionally, embedded OSs use a flat-memory model that statically maps logical memory addresses to specific physical memory locations, said Donn Rochette, OnCore's chief technology officer and vice president of engineering.

With a flat-memory model, Rochette said, if an application accidentally overwrites some critical data or code in the OS or elsewhere, the entire system can be corrupted. With virtual-memory architecture, he said, a misbehaving application damages only the section of memory with which it is working. Thus, an errant task cannot crash the entire system.

In addition, the virtual-memory approach eliminates the time-consuming need in the flat-memory model for programmers to assign each task a specific memory address and to keep track of these allocations, explained Phil Parker, OnCore's vice president of marketing.

Meanwhile, Downing said, the OnCore OS can scale from a microkernel to an enterprise environment and uses as little as 512 Kbytes of memory. The OS supports x86 and PowerPC processors, as well as an array of APIs.

Venture Development analyst Chris Lanfear said increases in product complexity and time-to-market pressures are driving commercialization of embedded-OS technology.

Thus, predicted American Technology International's Krasner, as the demand for new proprietary OSs decreases, vendors will shift to developing APIs and applications for their OSs. Specifically, Krasner said, they will begin focusing on connectivity to optical networks, which are becoming increasingly popular.

Commercialization of embedded-OS technology will also present challenges to vendors, as they have used customization in the past to cope with many design difficulties. ✷

Sixto Ortiz Jr. is a freelance technology writer based in Spring, Texas. Contact him at sortiz1965@yahoo.com.

Editor: Lee Garber, *Computer*, 10662 Los Vaqueros Circle, PO Box 3014, Los Alamitos, CA 90720-1314; l.garber@computer.org