From heavyweight applications to Embedded Linux

OS TO THE HEART OF GADGETS

DR. INDER SINGH

Dr. Inder Singh CEO and chairman LynuxWorks

Embedded Linux is here to stay. The only question is: how big a player can it really become in the marketplace?

The early focus for embedded Linux was on thin clients and network appliances – extending the operating system's pivotal part in Web and mail servers, Internet gateways and firewalls. Also, with heavyweight support from software and hardware developers in the real-time arena, Linux is being adopted as the platform for critical sub-systems in communications, retail technology, industrial control, transportation systems and aerospace. Pretty soon it will be leading the way towards a post-PC generation of gadgets in the home, as well as PDAs, mobile phones and mobile entertainment systems.

For those who are used to Linux as the OS of choice for their servers – and increasingly their desktop workstations – the idea of Linux inside their PDA or games console, let alone their mobile or MP3 player, seems surprising. Yet Linux enthusiasts have been recompiling Linux kernels to run on their Palm devices for years, and building hobby projects such as MPEG decoders in their basements.

For real commercial applications, however, it has required robust, productive development environments, specialised distributions and broader CPU support to get the embedded Linux bandwagon rolling.

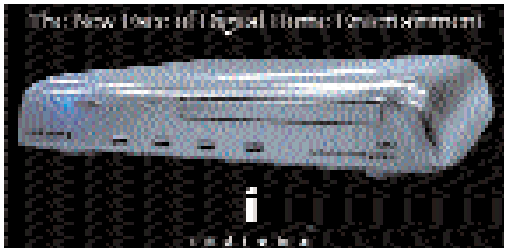
Yet rolling it surely is. Over the last few months the computer press has reported a wave of new embedded Linux designs that will appear in the shops very soon. This includes a mobile phone with a 10cm square screen; a screen phone, a high-end games console, a home networking gateway and a palm-sized PDA.

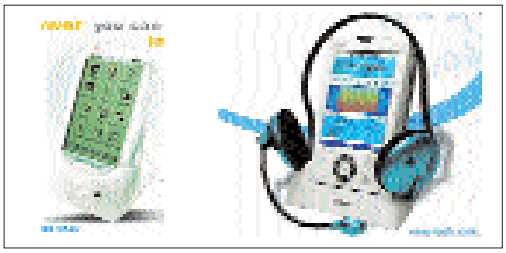
The appeal of Linux to embedded designs is clear. In contrast with proprietary operating systems such as Windows CE, Embedded NT, or Wind River Systems' VxWorks, it is both Open Source and an open system. It's royalty-free and, like the best of real time OS', Linux supports POSIX and other standard, open interfaces for networking and graphics.

Developers are protected from single-vendor dependency. They can readily move applications to Linux systems from multiple suppliers and other flavours of Unix, and benefit from a massive body of existing software and expertise. Wide availability of drivers, open source code and licensed software, all work to reduce development effort and cost.

Linux is well supported by the thousands of developers working to refine and improve the OS and has a reputation for being robust, reliable and secure. These are major advantages for embedded systems.

Most importantly for the gadget industry, the configurable, modular kernel of Linux makes it easy to meet the resource constraints of embedded applications. It is relatively simple to compile a compact OS that supports just the functions required, and in the tiny footprint of a portable, battery-operated device.





[links]
The game console Indrema L600 with Linux.

[mitte]
Agenda (Agendacomputing) and Yopy (G-Mate) are available, soon.

[rechts]
Looks like a PDA, but is a cell phone: the IMT-2000 from PalmPalm

Comprehensive hardware support

For embedded system developers, comprehensive hardware support is a must: the operating system has to perform with specific processors, I/O devices, buses, networking and graphics standards. Linux does not disappoint – it has already been ported to more CPUs than any other OS. In addition to Intel x386 to Pentium-class processors, the latest versions of Linux have been ported to Power PC, Hitachi SuperH, ARM, StrongARM and MIPS embedded CPUs.

Hitachi processors are widely used in a variety of embedded systems such as multimedia and consumer products – including hand-held personal digital assistants, digital still cameras, and game machines. Meanwhile, ARM versions are set to accelerate development of new applications, including audio and digital imaging, video appliances and kiosks, Web-enabled cell phones, highly intelligent point-of-sale (POS) terminals, residential gateways, navigation systems and others.

And Linux' hardware capabilities go a lot further. It can handle a rich mixture of target environments, from single boards to VME- or PCI-based multiprocessor systems, and a wide variety of devices and system buses.

Drivers are quickly made available to the developer community – for free – as new I/O devices are introduced with increasing frequency. For bus-based systems there are Linux ports to VME and PCIbus single-board computers and drivers for a variety of other standards, including CAMAC, CAN, GPIB and BitBus.

Despite these attractions, adopting Linux for embedded systems has only recently become a simple task. Why? For a start there have been few distributions specifically for the embedded developer. By and large developers have had to work with distributions geared mainly to desktop computers and servers. Worse, these constitute a fast-moving target as new kernel versions are posted frequently. Embedded developers have to balance cutting edge innovation against stability.

Fortunately, specialised versions of Linux – such as BlueCat Linux from Lynuxworks – are now being introduced to address these issues. Typically, these bundle a version of Linux with a range of specialist tools to aid embedded system development. In common with several other alternatives, the LynuxWorks option features a highly configurable kernel, making it simple to create variations according to user requirements.

Development environments

Designing an embedded system is a very different process from developing software to run on a workstation or server. Traditionally, developers use their familiar workstation or PC environments to develop code, then use specialised tools for subsequent cross deployment in the more resource-constrained target hardware. The development environment needs to support this cross development paradigm with powerful cross compilers, debuggers, image binding tools, and networking resources on the host. Equally critical, developers using Linux need various add-on tools that enhance the ability to build, debug, test and deploy embedded applications.

BlueCat Linux, for example, includes industry-standard ANSI C and C++ compilers, structured macro assemblers for supported target processors, the ld Linux linker for ELF relocatable file binding, and mkimage, a powerful tool for building kernel downloadable images containing complete bootable and ROMable user application and Linux OS binaries.

To provide cross-execution of system-level code in a controlled setting, Linux distributions can carry a variety of debugging tools aimed at embedded system developers. For example, the Total/db source level debugger is based on the popular GNU gdb, but enhances its core features to target the Linux OS kernel, specialised device drivers and embedded applications.

Embedded Linux is also benefiting from add-on tools, such as new performance analysers, like the LynuxWorks SpyKer event tracing tool, to find elusive errors resulting in resource contention, throughput bottlenecks, deadlocks and race conditions, which source-level debuggers alone do not easily tackle.

The fact that these facilities are now starting to appear is a clear indication that Linux is set to grow substantially in the embedded applications market. Once developers of embedded systems find that they can count on support by vendors who are focussed on their needs, and can provide Linux solutions that meet their most demanding requirements, industry observers expect the use of embedded Linux to far outweigh the operating system's use in general computing applications. ■

Read further:

LynuxWorks: http://www.lynuxworks.com
Lineo: http://www.lineo.com
Montavista: http://www.mvista.com
Embedded Linux News Channel: http://www.linuxdevices.com

■