Selecting an Embedded Operating System:

Key Decision Factors for Device OEMs

When Considering Embedded Linux and Windows Embedded

ABSTRACT

This paper describes:

- Cost factors to consider when choosing among embedded OS options
- Comparative data on Windows Embedded and embedded Linux costs
- Other factors to consider regarding Windows Embedded and embedded Linux

By:

John Coyne
Mike Starnes
Solutions Architects
BSQUARE Corporation

Table of Contents

Introduction

An OEM's choice of embedded operating system when designing a new device will have strategic implications that affect the time to market, total development and support costs, and ultimately the profitability of the product. As the earlier widespread use of proprietary, in-house embedded operating systems has fallen off dramatically, many OEMs find themselves choosing between embedded Linux and the Microsoft® Windows® Embedded family of operating systems.

Linux has attracted attention recently as an option for embedded devices. Platform and application developers are attracted most of all to the open availability of community-maintained OS source code, while business decision makers find the absence of license royalties on the Linux kernel appealing.

The Windows Embedded platform, which includes Windows XP Embedded and Windows CE, is appreciated for robust and proven technologies and its Microsoft support. Software developers like the familiar Win32 API and the availability of mature, well integrated development tools, while business decision makers appreciate the wide availability of Windows programming skills and the speed with which a product can be brought to market.

This paper is intended to help an OEM ask the right questions in order to make an informed choice between embedded Linux and Windows Embedded.

Determining the true OS cost

When calculating the true cost of the OS and related components for a new device, it's important to consider several elements.

- **Developer tools.** Any development effort requires developer tools. Both the direct and indirect costs for these tools should be compared. Direct costs are those that must be paid to buy or license the tools up front, plus any annual fees required for ongoing maintenance and support. Indirect costs include the learning curve for developers who may be unfamiliar with the tools, and inefficiencies and delays that may result if the tools being used are not mature, robust and well integrated.

- **OS adaptation and integration.** Any OS requires extensive development work to adapt it for use in a particular device. Alternative OS development and Quality Assurance costs should be compared in terms of low-level hardware adaptation (drivers, firmware, etc.), addition or removal of OS components, and the porting and integration of custom and third-party applications.

- **Component royalties.** Many embedded devices require either real-time (bounded, deterministic) performance or advanced user features such as browsing and media streaming. Cost comparisons between OS choices should include the costs of licensing any such advanced features required by the project.

Comparing Costs: embedded Linux vs. Windows Embedded

Developer Tools

Microsoft's Windows Embedded products are built around a comprehensive set of integrated development tools. Depending on the chosen OS, the central platform development tool will be Platform Builder for Windows CE or Target Designer for Windows XP Embedded. In either case the cost is $995 per developer, with no subscription or mandatory maintenance or support fees.

Different sets of embedded Linux tools are offered by different companies with widely varying cost structures. A survey of available options found the one-time cost for embedded Linux tools to average just under $4000 per developer per year.[1]

A more significant factor in the long run can be subscription and maintenance fees. Microsoft Windows Embedded tools require no subscription or maintenance fees, and the platforms are supported by Microsoft for a total of ten years from release.[2] By contrast, most embedded Linux tools require either an annual subscription fee or separate annual support and maintenance fees. Maintenance fees can range from $750 up to $10,000 per developer per year, while costs for those requiring subscriptions can run even higher. The longer the support lifecycle of the embedded device being built, the more important these recurring annual subscriptions become.

OS Adaptation and Integration Effort

Windows Embedded developers take advantage of an integrated development environment and a mature, common API. OS images can be built using either a GUI-based or command-line interface to suit developer preference and the various sub-tools are built in so that tasks can be performed in a single environment.

Linux did not originate as an embedded operating system, and there is no single embedded development environment for Linux. Developers must self-assemble a working embedded development tool chain from among the various tools available on the Web or from specific embedded Linux tool chain distributors. Few of these mostly command-line based development tools were created with the needs of embedded developers in mind.

Embedded developers using Linux face a number of other hurdles compared to using Windows Embedded:

---

[1] "Total Cost of Development: A comprehensive cost estimation framework for evaluating embedded development platforms," Jerry Krasner, Ph.D., Embedded Market Forecasters (EMF), July 2003. Read the whitepaper at http://www.embeddedforecast.com/EMFTCD2003v3.pdf . Linux tools and support costs in this section are derived from the table in Appendix C.

[2] In June of 2004 Microsoft announced their product support lifecycle would grow from eight years to 10 years. Five years mainstream support includes warranty support, plus five more years extended support includes security update support at no additional cost. For details see: http://msdn.microsoft.com/embedded/support/lifecycle/newlifecyc

- Linux lacks a common API, despite the Embedded Linux Consortium's attempts to define one, while Windows Embedded uses the Win32 API already familiar to millions of developers. (Windows XP Embedded exposes virtually the same 10,000 APIs as XP Professional, while Windows CE uses over 1,800 of the most important APIs with minor variations.)

- Device driver development is perhaps the single largest and most difficult set of tasks involved in using Linux for embedded applications, due to inconsistent driver coverage and less mature driver development tools. By contrast, Windows XP Embedded includes about 9,000 drivers and can support any WHQL-certified driver with minimal effort, while Windows CE ships with hundreds of the most popular device drivers in fully tested source code form and offers powerful driver development and porting tools. There are also hundreds of other Windows CE device drivers available from third parties.

- Linux distributions typically lack some key applications and technologies required by embedded devices. As a result, embedded Linux projects must include the effort of developing -- or the cost of licensing – and integrating such features as web browsers, multimedia codecs, Bluetooth communications or real-time kernel support. Windows Embedded projects begin with technologies such as these already integrated into the tool kit and covered in the licensing structure for reduced complexity and shorter development time. (Licensing costs are discussed in more detail below under Component Royalties.)

The different level of effort between OSs is reflected in team size and time-to-market differences. Windows Embedded projects tend to be completed faster using a smaller development team.

Two comparative studies of real-world embedded development projects, conducted in 2003, found that Windows Embedded projects were completed in less time than comparable projects using embedded Linux. Venture Data Corporation (VDC) found that a substantially higher percentage of Windows Embedded projects were completed in less than six months compared to embedded Linux projects (see Figure 1).

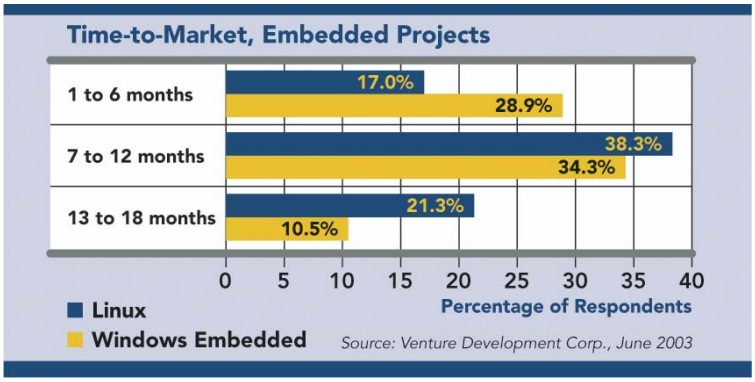

Figure 1.
Devices developed with a Windows Embedded OS are completed more quickly than those developed with embedded Linux.

Embedded Market Forecasters similarly found the average embedded Linux development project required 14.3 months to complete, compared to an average of only 8.1 months for Windows Embedded projects.[3]

Meanwhile, embedded Linux development teams have been found to be significantly larger than teams working with a Windows Embedded OS, according to three studies by two different organizations (see Figure 2).

Study	Windows Embedded (N)	Embedded Linux (N)	Other Commercial OS (N)
EMF 2003	7.9 (50)	14.2 (50)	N/A
VDC 2003	3.7 (36)	4.8 (44)	6.6
VDC 2002	4.2 (27)	6.3 (75)	5.4

***Note:** *N=Number of Respondents. The relatively larger team sizes reported by EMF may reflect in part the fact that EMF specifically included test and QA and project management in the count, while VDC was more narrowly focused on actual software developers.*

Figure 2.
Three studies show average Windows Embedded development teams are smaller than average embedded Linux teams.[4]

It stands to reason that larger teams working longer on projects will incur a larger total cost of development. In an effort to quantify the comparative development costs, EMF created a model to analyze the data collected on 100 embedded development projects. The results, shown in Figure 3, indicate total development costs for Windows Embedded at less than half those of embedded Linux.

EMF Study Data Results	Windows CE .NET	Windows XP Embedded	All Windows Embedded	Embedded Linux
Total Time to Market, months (mean)	8.2	8.0	8.1	14.3
Software Engineers/ Project, #	8.3	7.3	7.9	14.2
Development Man Months (mm)	68.1	58.4	64.0	203.1
Cost/mm of developer's time	$7,500	$7,500	$7,500	$7,500
Total Cost of Development	$510,450	$438,000	$479,925	$1,522,950
Comparative Savings (relative to embedded Linux)	66.5%	71.2%	68.5%	

Figure 3.
A comparison of total development cost for projects using a Windows Embedded OS or embedded Linux

[3] EMF 2003, p. 10.

[4] Sources: EMF 2003, p. 10; and "Comparing Linux and Windows embedded development," Chris Lanfear and Steve Balacco, Venture Development Corporation,, August 11, 2003. This white paper can be read at http://www.windowsfordevices.com/articles/AT7804488382.html.

Component Royalties

Software licensing costs are a factor in Linux as well as Windows Embedded projects. How those costs compare will depend on the production quantity and feature requirements of the devices. More sophisticated devices will incur higher licensing costs. On the Linux side these costs will generally come from licensing individual third-party solutions such as a web browser, a real-time kernel, or thin client terminal software. On the Windows Embedded side the increased cost will come in the form of licensing a more expensive OS or version that includes the needed features.

Windows CE is a componentized OS with different licensing levels, ranging from under $3 to over $18, depending on volumes and licensing levels. Licensing levels correspond to feature components included in the build (see Figure 4). Windows XP Embedded has only one licensing level (about half the price of XP Pro), though it is also componentized so that unneeded features can be removed from the footprint. Depending on device requirements, then, there are four levels of Windows Embedded licensing, as shown in Figure 4. Exact costs depend on license quantity and distributor pricing.

Windows Component	**Windows CE Core ($3-$4)**	**Windows CE Professional ($14-$15)**	**Windows CE Pro Plus ($18-$20)**	**Windows XP Embedded ($65-$75)**
Real-time OS kernel, support for ARM, MIPS, SHx, X86 processors	✓	✓	✓	
File system, networking, hardware support, ActiveSync	✓	✓	✓	✓
Browser, help system, inbox, Pocket Outlook, messenger		✓	✓	✓
Media player, WMV, MP3, MPEG-4 streaming		✓	✓	✓
Digital Rights Management		✓	✓	✓
Thin client / terminal		✓	✓	✓
Desktop synchronization (Inbox Sync, Pocket Outlook Sync)		✓	✓	✓
Office/Acrobat file viewing			✓	✓
Ability to run full XP applications unmodified				✓

Figure 4. Windows Embedded licensing options: key components and approximate costs based on 10,000 units

Although open-source Linux kernel code is available without royalty, many of the features needed for embedded projects are not included. Some features may be developed in-house by those OEMs with substantial software development capabilities, but in many cases the solution will be to license individual components from third-party vendors. Figure 5 shows average estimated costs, in the form of run-time royalties, for licensing some widely used application components.

Linux Component	Fee
Real-Time OS Kernel	$ 9.00
Web Browser	$ 8.12
Media Player	$ 2.00
MP3 Decoder	$ 0.75
MPEG-4 Decoder	$ 0.25
Digital Rights Management	$10.00
Thin Client / Terminal	$14.00
Desktop Synchronization	$10.00
Residential Gateway Stack	$ 3.00

Figure 5 Estimated embedded Linux component run-time royalties, based on a volume of 10,000 units.[5]

Clearly the bottom line calculation of royalty costs will be different for every project and will depend on choice of vendors as well as device features and production quantities. Because there are so many variables, it is well worth the effort to calculate estimated royalty costs for different scenarios during the decision-making process.

Non-cost factors in choosing the right embedded OS

Cost is important but should not be the only factor determining which embedded OS to choose. OEMs also need to evaluate the risks involved in the choice.

Financial Risk

It is important to ask, at what stage in the project will each of these costs be incurred? Of course, most of the actual engineering costs will occur during the design and development stage, but other payments may be due either before or after the device goes to market. It is clearly less risky to choose a model in which as many costs as possible come due only after the device goes to market and can begin generating revenue.

Embedded Linux requires higher upfront costs of developer tools, OS support and maintenance costs—regardless of whether the project is ever completed and shipped. This cash outlay represents a financial risk if the project is canceled or the design abandoned because of changing market directions or business demands.

---

[5] EMF 2003, p. 16.

The Windows Embedded business model is principally based on run-time royalties paid at the time the device ships. This model offers low upfront costs including low-cost developer tools, free OS maintenance for ten years, and a "pay as you go" plan for OS support.

With this model, revenue on royalties for Microsoft and Windows Embedded licensing Value-Added Partners such as BSQUARE will incur at the time the product is shipped. This incentive gives Microsoft a vested interest in the success of its customers' projects and in continually improving the Windows Embedded OS functionality, developer tools, and related products.

Technology Life Cycle Risks

The longer the life cycle of the device under development, the more important it is to think about whom will maintain and support of the code down the road.

OEMs who choose to begin with open-source code and modify it themselves need to be ready to maintain and support not only their own applications but the quality of the entire OS code base for the life of the product, whether that is three, five, or ten years into the future.

OEMs who choose to license embedded Linux components from vendors and pay annual support and maintenance fees need to ask themselves if there is a risk that those vendors will no longer be in the support and maintenance business five or ten years later. If components are licensed from multiple vendors, the risk some of them may not be around in the future is heightened.

OEMs who choose to license Windows Embedded products gain the benefit of Microsoft employing thousands of dedicated software test engineers and leveraging the efforts of thousands more through beta programs. The Microsoft policy of maintaining and supporting Windows Embedded products for ten years from release is a major risk reduction.

Intellectual Property Risks

When comparing business models, smart device makers should also bear in mind the issue of who owns the source code running their embedded devices.

Open source code access has certain risks and ambiguities. It is beyond the scope of this paper to untangle the intricacies of General Public Licensing (GPL), but only to advise OEMs considering embedded Linux to make sure they are comfortable with the legal ramifications regarding any intellectual property that will be included on the device.

Windows Embedded OS source code is protected by Microsoft's commercially-proven licensing models that preserve an OEM's intellectual property rights. The details are spelled out in a Customer License Agreement (CLA) and an Additional License Provisions (ALP) agreement.

OEMs who are most protective of their intellectual property also appreciate the way BSQUARE can act as a trusted intermediary between themselves and Microsoft, further maintaining the confidentiality of all proprietary source code.

Device Requirements

What device requirements must be served by the OS?

- Interoperability with other devices
- Customization
- Interface with Windows-based applications
- Support for required hardware and software components
- Special application features

Costs

What are the development costs for this device?

- Developer tools licensing and subscription
- Adaptation and integration expense
- OS licensing
- OS maintenance and support fees
- Required application components licensing
- Run-time royalties

Time to Market

What is the projected development schedule?

- OS installation, configuration, adaptation
- Application development
- Testing and quality assurance

Development Team

What are the OS requirements?

- Developer skills and availability
- Estimated team size

Risks

What are the financial risks?

- Costs due on project success, or up front
- Partners vested in OEM success, or not

Who will maintain and support the code in the future?

- OEMs themselves
- Embedded Linux vendors
- Microsoft

How can the OEM protect valuable IP?

- IP ownership issues or risks
- Licensing issues for OS and/or internal IP
- Trusted intermediary

Figure 6. Checklist of issues to consider when choosing an embedded OS for new device project.

Sources:

Krasner, Jerry: *Total Cost of Development*. A report from Embedded Market Forecasters, July 2003.

Balacco, Stephen and Christopher Lanfear: *Linux in the Embedded Systems Market. The Embedded Software Strategy Market Intelligence Program, Vol. III*. Venture Development Corporation, June 2003.

Balacco, Stephen and Christopher Lanfear: *Linux's Future in the Embedded Systems Market. The Embedded Software Strategic Market Intelligence Program 2001/2002 Volume VI.* Venture Development Corporation, April 2002.

About BSQUARE

BSQUARE is a leading global provider of software, engineering services and consulting for the smart device market. Offering more than 10 years of experience with embedded technology, BSQUARE guides OEMs in selecting the best operating system for a device. For more information, visit **www.BSQUARE.com**, or call **888-820-4500** or **425-519-5900**.

######