# The Windows Embedded Academic Program – Retrospective & Directions, 2002-2006

Lindsay T. Kane
Product Manager
Mobile & Embedded Devices Group
One Microsoft Way, Redmond
WA 98052, USA
+1 (425) 707 7800
lindkane@microsoft.com

D. Stewart W. Tansley, PhD
Program Manager
Microsoft Research
One Microsoft Way, Redmond
WA 98052, USA
+1 (425) 707 5686
stansley@microsoft.com

## ABSTRACT

The Windows Embedded Academic Program (WEMAP) was founded in 2002 with the aim of bridging some of the gap between Microsoft's commercial training offerings for its embedded systems operating systems products, and the embedded systems teaching needs of higher education institutions. In addition, through partnering with Microsoft Research, it has explored how to make the Microsoft embedded systems products more accessible to the university research lab. This paper explores some of the history and lessons learned from the first four years of this program, summarizes the current and future directions of the program, and seeks to encourage a dialogue – through the vehicle of the WESE 2006 workshop and beyond – between professors active in teaching embedded systems and Microsoft embedded systems representatives, who are keen to shape the program to meet the needs of academia and industry alike. The paper presents the joint personal perspectives of the authors who have both been closely involved in the program, and does not necessarily represent an official position of Microsoft Corporation.

## 1. INTRODUCTION

Commercial embedded systems operating systems (OS) manufacturers such as Microsoft frequently make their products available for academic purposes, both in teaching and research contexts, often at little or no cost, for a variety of reasons. There is the obvious perceived commercial benefit of seeing students educated in the capabilities of their products, with the assumption that this will eventually transfer into the careers of graduates and perhaps their employers' potential purchasing decisions in due course. However, this is a long term benefit at best, and companies also value the many shorter term benefits, such as feedback on their products from a particularly eager and inquisitive community, experience with novel training techniques, and research opportunities. The latter deserves independent discussion and is largely deferred to be outside the scope of this paper. We instead focus on the pedagogical dimensions, and in particular, concentrate on the experiences of the Microsoft Windows Embedded Academic Program (WEMAP) [1] as an example of commercial embedded OS software that has been made especially accessible to university classrooms.

## 2. BACKGROUND

This section explains the scope of the WEMAP program for those with little or no prior knowledge of the Windows Embedded product portfolio.

### 2.1 What is Windows Embedded?

The Windows Embedded product group [2] has been responsible for two main embedded OS products, Windows CE [3] and Windows XP Embedded [4]. A number of more specialized systems for particular vertical markets are also covered by this group, but the Windows Embedded Academic Program discussed in this paper has focused solely on these two general purpose OSs.

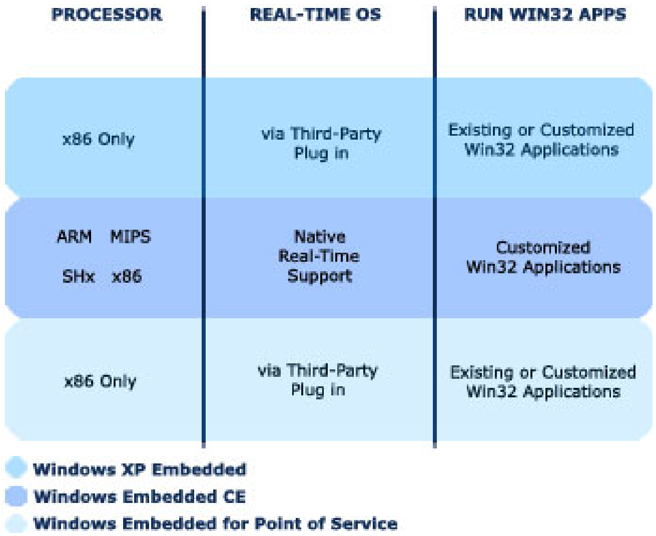
Windows CE (aka CE) is a 32-bit real-time OS for small footprint devices, such as battery-powered hand-held devices including PDAs (Personal Digital Assistant) [5] and Smartphones [6]. Windows CE is also used in industrial control devices such as robots [7]. The Windows CE kernel is unique to Windows CE, and contrary to widespread misconceptions, it is not based on either the MS-DOS (MicroSoft Disk Operating System) or Windows NT kernels used in Microsoft's desktop and server OSs. Windows CE is frequently seen embedded within the Windows Mobile platform (not currently covered by WEMAP), which is a special configuration of Windows CE purpose-built for PDAs and Smartphones. Windows CE is highly customizable by the embedded developer, allowing myriad special configurations of its hundreds of components appropriate to the specific target device and application. Windows CE runs in a number of CPU architectures beyond the familiar x86 desktop standard, including ARM, SuperH and MIPS. Windows CE is also notable for being supplied with many of its components visible in source code form.

Windows XP Embedded (aka XPE) is, simply put, a componentized version of Windows XP – Microsoft's flagship 32-bit x86 desktop OS. The componentization allows the embedded developer to leverage the vast PC ecosystem, together with its very attractive economics, while being able to tailor the

*WESE'06*, October 26, 2006, Seoul, South Korea.

OS platform to the specific needs of the target device, removing unwanted components. Special additional components provide extra functionality of benefit to the embedded developer, allowing booting from memory for example and avoiding the need for a hard disk. Because of the common codebase with the desktop OS, XPE runs on x86 CPUs only and is not real-time – though extensions are available from third parties. Typical applications of XPE include kiosks, ATMs and gaming machines [8].

Figure 1 summarizes some of the contrasting key differences between CE and XPE (citing also for contrast, one of the special vertical application embedded OSs, Windows Embedded for Point of Sale). For a more application-oriented comparison, see [9].



| PROCESSOR | REAL-TIME OS | RUN WIN32 APPS |
|---|---|---|
| x86 Only | via Third-Party Plug in | Existing or Customized Win32 Applications |
| ARM MIPS<br>SHx x86 | Native Real-Time Support | Customized Win32 Applications |
| x86 Only | via Third-Party Plug in | Existing or Customized Win32 Applications |

● Windows XP Embedded
● Windows Embedded CE
● Windows Embedded for Point of Service

**Figure 1. Windows Embedded OSs compared.**

## 3. WEMAP Milestones

This section explains the history of the WEMAP program in broad strokes, identifying the most significant activities and events in order to provide an overview and introduction to the program.

### 3.1 Origins

Microsoft has had a long history of relationships with academia and there have been a rich variety of means of engagement, ranging from individual product groups and simple sales relationships with specific schools, to deep, rich, long-term partnerships in a research context. Microsoft has a number of academic-focused groups [10] with specific responsibilities to work with schools in various ways. In general, a product group (whose primary role is to design, manufacture, ship and market their products) will defer engagements with academia to these academic-focused groups, except in rare situations. The Windows Embedded product group was therefore relatively bold in its conception of the WEMAP. It arose both from the particular efforts of individuals championing the importance of academia to the product group, and positive feedback from direct early engagements such as with Lancaster University [11].

### 3.2 Partnership with Microsoft Research

From an early stage, Microsoft Research's (MSR) University Relations (UR) group (now called various names, but External Research & Programs in the US, see [12]) proved to be a valuable partner to the early WEMAP concept. MSR is well known to university researchers as a leading center for computer science research, and its UR programs such as the RFPs (Requests for Proposals) are popular, especially in a research context. MSR teamed with the nascent WEMAP from an early stage, and soon a major joint project was conceived to help launch the program.

### 3.3 The 2003-2004 Request For Proposals

WEMAP and MSR UR together designed and implemented a major world-wide RFP program focused on research and teaching with Windows Embedded (both CE and XPE). Overall, $1.7M was distributed to 76 projects in 26 countries, following a review of over 150 proposals. Most of the projects were research-oriented, but 14 were focused on teaching. Projects were scheduled to last 1 year from the time of funding, and a wrap-up workshop was held in September 2004 in Cambridge, UK. Results were published on a per-project basis through regular academic channels (various conferences and journals), and projects were also encouraged to submit their materials to the MSDNAA Curriculum Repository (see section 4.5). At the time, this was the largest academic RFP ever run by Microsoft [13].

### 3.4 Windows Embedded Student Challenge

During 2003, WEMAP teamed-up with the IEEE (Institute of Electric and Electronic Engineers) CSIDC (Computer Society International Design Competition) to form a new student team-based embedded system design competition called the WESC (Windows Embedded Student Challenge) [14]. Run each year since Spring 2004, 30 finalist teams from around the world have convened at Microsoft's headquarters in Redmond, WA, to determine best embedded software design team with projects using Windows CE and a common reference CPU component. Hundreds of teams enter the competition during a document-based elimination round. From 2007, this competition is being integrated with the much larger general Microsoft student competition, the "Imagine Cup" – see section 6. We hope this will enable even more students to participate.

### 3.5 A New Proposal for Academic Training

In the summer of 2006, a new approach to Windows CE training materials specifically designed for academics was decided to be based around the forthcoming next release of the product. Rather than periodically inviting professors to attend "Train The Trainer" (TTT) events (see section 4.6) which are based on commercial professional developer training materials, this approach would call for curriculum specifically designed by professors for the University classroom.

A group of established Windows Embedded professional trainers, embedded MVPs ("Most Valuable Professional"), and professors were invited to respond to an RFP in May 2006. Georgia Institute of Technology (principal investigator Professor James Hamblen) was selected to develop this curriculum based on its direct experience using Windows CE in the University classroom. The resulting curriculum materials arising from this contract will provide professors with the resources for learning about embedded development using Windows CE and will be made widely available to all academics in early 2007. [15]

# 4. KEY COMPONENTS OF WEMAP

We now turn to discussing the component parts of WEMAP that have not already been covered by the events described above.

## 4.1 Accessing the Products

As mentioned earlier, the WEMAP scope is Windows CE and Windows XP Embedded. The standard route for academics to actually get these products, both professors and students, is through Microsoft's regular academic software access channel, MSDNAA (Microsoft Developer Network Academic Alliance) [16]. This is a minimal cost program and provides very wide access to a large array of software (not just embedded), but it relies on the school's department to make annual subscription payments and there is a certain administrative overhead involved in downloading the software. Occasionally therefore, we have had to find workarounds to MSDNAA in certain situations, and this can be challenging as the Windows Embedded products are not sold through retail channels – i.e. they are not standard "shrink wrapped products". A temporary workaround is that full-featured evaluation editions of both CE and XPE are available as downloads on the public Internet [17].

## 4.2 Shared Source[1] – Curriculum License

Professors are asked to sign a special but simple licensing agreement if they wish to teach using the built-in source code in CE, called the Windows CE Academic Curriculum License [19]. To date, over 400 licensees have signed around the world. While this agreement is not onerous, and it does provide a means to start a conversation between WEMAP and professors who are interested in using CE in their classes, the requirement is a (small) barrier to overcome. We have periodically reviewed the necessity for this license and confirmed it must be continued. However, the current fax-back mechanism we use could be improved and we are seeking a more convenient process as soon as possible.

This attention reflects a trend towards increased accessibility of CE source code that reflects the very different requirements that embedded systems developers have for source code compared to other Microsoft products. Note that we have never required a curriculum license for teaching with XPE – though its source code is not available through standard channels for XPE, so such a license is moot.

## 4.3 Shared Source – Research License

Historically, a "research source code license" was available to support researchers using CE who needed access to certain software modules that were not available in source form in the standard distribution of CE. This license was based on the original source code license for the regular desktop Windows, which qualified researchers have been able to access for some years. Recently, the research license for CE has not been active but we are reviewing this situation based on demand from academia. One challenge with this license has been its restriction to certain countries [20].

## 4.4 The Hardware Empowerment Program

Originally conceived as a way to help researchers and educators get up to speed quickly in their RFP projects, the Hardware Empowerment Program (HEP) [21] is a way for embedded development board manufacturers to provide incentive discounts to academics, as well as the BSPs (Board Support Packages) to allow marrying of the Windows Embedded OSs to their hardware. At the time of writing, 10 such manufacturers are part of the program. The discounts are solely the responsibility of the manufacturer and vary between product and manufacturer. Microsoft's role is to provide a readily-accessible portal for its academic partners to see the various hardware and discounts on offer. Microsoft welcomes new vendors applying to join this program.

## 4.5 The Curriculum Repository

The Curriculum Repository, recently renamed "The Academic Alliance Repository" [22] has been a key feature of the MSDNAA software distribution portal for a number of years. Its purpose is to provide a freely-available large database of proven useful downloadable content materials produced primarily by other academics for academics. (There is some Microsoft-authored and third-party-company authored material too, but academic-authored material dominates). The Repository is fully searchable by keyword and browsable by topic area. At the time of writing, there are over 50 embedded systems data items across 10 types present, most of which are Windows Embedded related. We continue to encourage all of our sponsored or otherwise supported academic projects (e.g. RFPs) to submit their embedded systems and other contents to the Repository, in order to provide useful items for the academic community at large and further the reputation of the authors within their peer groups.

## 4.6 Train the Trainers

An important recognition in the commercial side of Windows Embedded is that Microsoft is better placed to be a "trainer of trainers" rather than of individual developers. This reflects the business model of Windows Embedded whereby Microsoft sells its embedded products through embedded systems software distributors rather than directly to developers or device manufacturers. Microsoft therefore works with training vendors to significantly multiply the reach of its training efforts. To support this approach, Microsoft develops in-house materials to train such trainers, and then the trained trainers will augment these materials to deliver individual developer training.

Microsoft has periodically invited professors interested in teaching using Windows Embedded products to join in these TTT ("Train The Trainer") events. We have also invited groups of professors, often also with their research students, to attend quasi-training events including Microsoft's professional developer conferences. An example is the annual MEDC (Mobile and Embedded Developer Conference), which is packed densely with detailed technical briefing sessions, hands-on labs, and demonstrations. We also try to make these materials available more widely on disk after the event itself.

However, Microsoft is not in a position of expertise when it comes to the pedagogical requirements of a teaching professor and their students in a university classroom, so it is not likely optimal for Microsoft to train such professors in the way that it trains its commercial training and developer partners. It is this recognition that has led to the development of authoritative curriculum materials through contracting with a respected professor and their team, as described in section 3.5, and an

[1] "Shared Source" is the collective program name for Microsoft's source code availability to customers, partners, developers, academics, and governments worldwide. See [18].

evolved outreach approach in the form of “Academic Days” described in section 5.2.

## 4.7 Other Components

Over the years, there have been a number of other aspects to WEMAP, but the ones above are dominant in terms of the program’s structure. The other aspects include an academic association with the MVP (Most Valuable Professional) [23] program that is a common part of Microsoft’s extensive developer community, sponsorship of conferences or academic attendance at conference (including Microsoft “developer conferences”, aka DevCons, such as MEDC mentioned above [24]) and various awards to specific projects under special circumstances. We have also provided a default point of contact for academics seeking information or other questions about Windows Embedded for academic use. These items will be deferred from this paper for the sake of focus.

# 5. NEW DIRECTIONS

## 5.1 Software Licensing

We expect continual evolution in the software licensing aspects of WEMAP as Microsoft continues to evolve its Shared Source initiative. Today, it should be noted, there are minimal or no special software licensing requirements that a professor needs to consider when using Windows Embedded software, beyond the click-through EULAs (End User License Agreement) in the product themselves, and the fax-back one-page Shared Source license for teaching using the source code. Microsoft is particularly keen to hear from the academic community what source code is required for teaching and research purposes if it is not already available – the trend to date has been to make steadily increasing amounts available according to customer demand (including academic customers).

## 5.2 Academic Days

As described in sections 3.5 and 4.6, we recognize room for improvement in the ways we can help professors maintain their knowledge of Windows Embedded products and technologies. We do not feel this is a “solved problem” in any sense – the rapid pace of evolution of the embedded systems field, and the Windows Embedded portfolio in particular, makes this especially challenging.

A new approach we are evolving to is the use of dedicated intense training events for professors, based on a model that Microsoft’s general developer education organization has found effective, but specifically tailored for academics and with academic-only attendance – we call these “Academic Days”. This is a “crash course” style event held over 2-3 days for large numbers of professors (~50-100 is typical) who are invited to make their own way to a venue, usually a university campus. Microsoft provides a series of academic-oriented training and seminar material presented by a mix of professors, embedded MVPs and Microsoft staff, usually from the core product development team. We will continue to refine this approach over time based on feedback from attendees and welcome comments.

## 5.3 Application Development for Embedded

Expanding on the theme of application development further, it is notable that the scope of “embedded system” is broadening rapidly – today’s everyday pocket devices like our smartphones are powerful computers in their own right, and often as powerful as the desktop computers of just a few years ago. As we discuss embedded systems development education therefore, the educator must consider the stretching of what it means to teach embedded systems to higher orders of software development.

For example, Microsoft’s Compact Framework (“NETCF”) [25] is a special version of the .NET Framework [26] managed code development environment, where the framework provides basic services such as memory management and security to the programmer, enabling greater programmer productivity. NETCF runs on Windows CE, enabling the embedded systems application developer to use managed code such as C# on their device rather than native languages without built-in memory management and security such as C or C++. The implications of such powerful applications development environments for the embedded systems programmer are still being realized. WEMAP today does not directly address such a scope – but perhaps it will need to consider this in the years to come. As with the other topics in this paper, we are keen to hear professors’ views on this.

## 5.4 Hardware Availability

A perennial challenge in embedded software development in the university classroom is the availability of inexpensive, robust, but capable, hardware platforms. Unfortunately the economics of embedded systems hardware means that these boards have tended to be relatively expensive (at least hundreds of US dollars per board is typical).

Alternatives such as using a PC to emulate an embedded system are pragmatic and somewhat useful, but many professors (and students) feel this ultimately frustrates the learning experience for the student – “it’s not truly embedded” is a common sentiment.

Much cheaper embedded systems boards are available (typically as evaluation kits for low-end processors), but they are generally far less capable, running 8-bit microcontrollers generally, and may not even run an OS – including not being able to run Windows Embedded.

Major systems trends such as Moore’s Law [27] have had an important impact however, and we are starting to see 32-bit boards become available at prices much closer to US $100 – an attractive price point in the student context, since it is around the price of a good text book. So while we do not see this as a solved problem today, the direction looks promising. Examples (will) appear at [21].

## 5.5 Can I Re-flash My Smartphone for Class?

Related to the hardware question but in terms of more complete devices, a common question from academics who want to teach or research using Windows Embedded has long been, “can I change the OS on a Windows Mobile device?” – such as a retail PDA or Smartphone that they or a student might already have.

The answer in general is no – at least, not unless this scenario is specifically supported by the manufacturer of the device in question, which is rare. Figure 2 illustrates the general case, with the software stack shown as an application running on the OS running on the BSP running on the hardware. Note it is possible to patch parts of the OS and achieve some customization in the Windows Mobile case. It is also possible of course to fully customize your application in both cases.

The common misunderstanding here is of the precise role of Microsoft in the ecosystem that produces such devices. Microsoft is the OS developer, and does not develop or manufacture complete Windows Mobile devices today. Often the crucial IP (Intellectual Property) that lies in the BSP for such devices is simply not made available by the device manufacturer. We continue to explore ways to improve this situation for the academic developer.

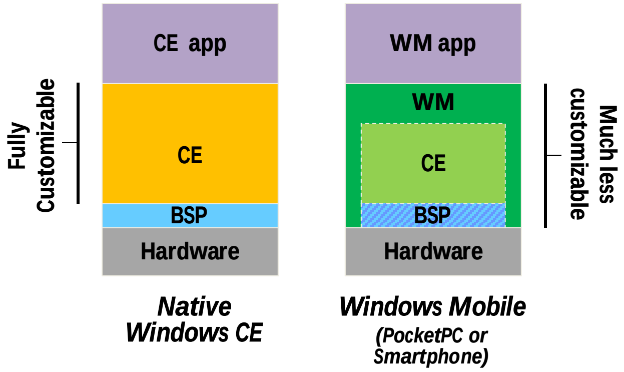


**Figure 2. Windows CE & Windows Mobile.**

## 6. WEMAP TODAY

What is the status of WEMAP today?

From the website [1], "The Windows Embedded Academic Program (WEMAP) helps provide a better understanding of the Windows CE and Windows XP Embedded operating systems to faculty and students. As a participant in this program, you can learn how to be part of the next generation of embedded developers or educators on Windows Embedded."

The website includes the following topics:

**Getting Started with Windows Embedded in the Classroom**

*Learn more about how to get started using Windows Embedded as an academic. Learn how to access Windows Embedded software and find out about the best resources to help you get up and running.*

**Faculty**

*Academic Curriculum: Learn more about curriculum to help you implement Windows Embedded in the classroom.*

*Faculty Training: Learn more about opportunities to get training on Windows Embedded.*

*Research Projects: Find out about Microsoft Research projects.*

*Discounted Hardware: Learn how to get discounted hardware from leading hardware vendors through the Hardware Empowerment Program (HEP).*

**Students**

*The Imagine Cup 2007 Embedded Development Invitational: Are you ready to change the world? Formerly the Windows Embedded Student Challenge, this competition challenges you to build your own embedded device that will have an impact on solving some of our world's toughest problems.*

*Community Development Projects: Find out about ways you can get involved in Windows Embedded Community projects.*

*Windows Embedded Career Opportunities: Are you interested in exploring opportunities to work at Microsoft? Find out about ways you can apply for Microsoft internships of full-time opportunities.*

Following these key topic areas, there is a news section, and some remarks about the difference between Windows Mobile and Windows Embedded (a common source of confusion and frequently asked question, per section 5.5). There is also a feedback section – again, we are keen to hear your comments and suggestions!

## 7. CONCLUSIONS

We have described the Windows Embedded Academic Program (WEMAP) from its founding in 2002 through its formative years, up to the present day in 2006. We have done this through illustrating some of the key interesting lessons learned, and provided pointers to some important future directions. We seek to encourage a dialogue through the paper's presentation at the WESE 2006 workshop, between professors active in teaching embedded systems and Microsoft embedded systems representatives, who are keen to shape the program to meet the needs of academia and industry alike.

## 8. ACKNOWLEDGMENTS

We wish to thank the numerous professors and students who have used the services of the Windows Embedded Academic Program since its inception and provided extremely valuable feedback, and also to our hardware partners in the Hardware Empowerment Program.

## 9. MORE INFORMATION

For more information on the current WEMAP program, please visit reference [1].

## 10. REFERENCES

[1] WEMAP home page,
http://msdn.microsoft.com/embedded/community/wemap/

[2] Windows Embedded home page,
http://www.microsoft.com/embedded/
See also the developer version of this page,
http://msdn.microsoft.com/embedded/

[3] Windows CE home page,
http://www.microsoft.com/windows/embedded/eval/wince/

[4] Windows XP Embedded home page,
http://www.microsoft.com/windows/embedded/eval/xpe/

[5] E.g., Windows Mobile Pocket PC Phone Edition home page,
http://www.microsoft.com/windowsmobile/pocketpc/phone/
(PDA is a device category available from other manufacturers).

[6] E.g., Windows Mobile Smartphone home page,
http://www.microsoft.com/windowsmobile/smartphone/

(Smartphone is a device category available from other manufacturers).

[7] E.g., KUKA Controls GmbH product, http://www.kuka-controls.com/product/cewin

[8] Case studies of devices using XPE (and CE), http://www.microsoft.com/windows/embedded/testimonials/

[9] Choosing an Operating System by Device Category, http://www.microsoft.com/windows/embedded/eval/choosing.mspx (contains a useful table of device types).

[10] Microsoft home page for Education, http://www.microsoft.com/education/

[11] *Academic Collaboration With Lancaster University Leads to Implementation of IPv6*, Redmond, WA, July 30, 2002. http://www.microsoft.com/presspass/press/2002/jul02/07-30NextWindowsCEPR.mspx

[12] Microsoft Research University Relations world-wide groups, http://research.microsoft.com/ur/

[13] *Microsoft Awards Academic Grants for Cutting-Edge Device Research and Curricula*, Redmond, WA, June 25, 2003. http://www.microsoft.com/presspass/press/2003/Jun03/06-25AcademicGrantPR.mspx

[14] Windows Embedded Student Challenge home page, http://www.windowschallenge.com/

[15] Professor James Hamblen home page, http://users.ece.gatech.edu/~hamblen/
(See also: http://msdn.microsoft.com/embedded/community/wemap/Faculty/default.aspx#WEAC)

[16] MSDNAA home page, http://msdn.microsoft.com/academic/

[17] Windows Embedded evaluation editions, http://www.microsoft.com/windows/embedded/eval/trial.mspx

[18] Shared Source home page, http://www.microsoft.com/resources/sharedsource/default.mspx

[19] Windows CE Academic Curriculum License, http://msdn.microsoft.com/embedded/community/wemap/Faculty/default.aspx#WEAC

[20] Shared Source Licensing Programs: Availability by Geographic Market, http://www.microsoft.com/resources/sharedsource/Licensing/Availability.mspx

[21] Hardware Empowerment Program, http://msdn.microsoft.com/embedded/community/wemap/Faculty/default.aspx#HEP

[22] Academic Alliance Repository, http://www.msdnaacr.net/curriculum/facetmain.aspx

[23] Windows Embedded MVPs, http://msdn.microsoft.com/embedded/community/community/mvps/
(See also the general parent program: http://mvp.support.microsoft.com/mvpexecsum)

[24] E.g., Mobile & Embedded Developer Conference 2006, http://www.medc2006.com/

[25] Compact Framework home page, http://msdn.microsoft.com/netframework/programming/netcf

[26] .NET Framework home page, http://msdn.microsoft.com/netframework

[27] Moore, G. E, *Cramming more components onto integrated circuits*, Electronics Magazine, 19 April 1965.