Longhorn Ties Platform Apps to Core Operating System

Alexander Wolfe, Science Writer

Call it all-tools-for-all-things-Microsoft. That's the way I'm beginning to think about Longhorn, Microsoft's next-generation operating system, expected in 2006. The operating system will herald more than just the transition of mainstream computing to 64 bits. It will mark the emergence of a new, multifaceted programming model.

Breaking away from traditional computer science taxonomy, where operating systems stood barely removed from the hardware abstraction layer, Longhorn will be more like a multi-tentacled container for a wide range of administration and application tools.

Longhorn's basic operating system will concern itself with the control of interrupts and memory, and the compartmentalization of systems resources so that apps don't crash. Here, Longhorn will carry forward the concept of CLR (common-language runtime), which Microsoft has introduced as part of its .NET initiative.

CLR, which enforces the partitioning of processes into zones of managed code, encapsulates applications. It handles low-level details such as memory management and security functions in order to avoid crashes and to prevent the execution of malicious code.

Support from hyperthreading and the soon-to-be-released multicore processors from Intel and AMD will also be key elements of Longhorn.

It is Longhorn's companion packages, however, that will constitute a remarkably broad applications grab—from a company that's being pressured (by the European Commission, for one) to unbundle.

Please note that I'm not taking a position on what Microsoft is doing here; I'm just reporting it. In Redmond's defense, I would point out that the x86 platform has become so complex that Microsoft's move to address all elements of the platform via a coherent, overarching software architecture may in fact be the only sensible thing to do. That complexity is becoming amply evident with the imminent expansion of the x86 microprocessor to 64 bits through the use of instruction set extensions in AMD's Opteron and Athlon 64, and in Intel's new 32/64 Xeon. (Fully 64-bit architectures such as Intel's Itanium 2 are seeing service in high-end servers, not mainstream PCs). At the same time, the surrounding PC platform is

Will Microsoft's

NEW OS BE A DEVELOPER'S

DREAM-COME-TRUE?

becoming host to next-generation I/O channels such as PCI Express, faster memories in the form of DDR2, and increasingly

sophisticated graphics architectures.

But back to Longhorn. This operating system will implement a new file system (WinFS) and a new programming model (WinFX). Its companion Indigo package will support Web services for the Longhorn platform. Yukon will deliver Microsoft's next-generation implementation of SQL Server. Avalon will constitute the media-presentation technology component.

The emergence of advanced Web services as a key part of the Longhorn architecture has been spurred by the maturity of the Internet and by Microsoft's intention to become a mainstream player in enterprise computing.

This strategy, in turn, will yield a major role for XAML in both Indigo and Avalon. XAML is a declarative markup language aimed at supporting Web services. Longhorn programmers will be able to use XAML to design the user interfaces for their applications. XAML will provide a high-level syntax, which Microsoft wants software designers to use to create the front ends of their applications. Microsoft is providing hooks so that XAML descriptors can invoke functionality from the .NET Framework. The latter is Microsoft's massive, interactive development environment, complete with programming language and compilation support, and a runtime repository.

To fill out top-level XAML with lower-level business logic, developers will be encouraged to use a "procedural .NET Framework language," a.k.a. a homegrown Microsoft programming language—such as C# or Visual Basic. (They'll be able to use C++, but that language's tie-in to Microsoft's software libraries will be more tenuous.)

Avalon, the media-presentation technology component of Longhorn, is being positioned as a new graphics subsystem that serves as a foundation for Longhorn's shell. Avalon will come with a full set of the user interface components for Longhorn, and here, too, XAML will serve as a front-line tool.

Because XAML is a generic mechanism for describing

managed objects, Microsoft is drafting it into service to build both the Windows Forms UI (user interface) and Avalon UIs. (Programmers will also be able to intermix Avalon and Windows Forms controls using XAML.)

Two features much touted by Microsoft for Avalon are composability and navigation. Using the former, new Windows Forms can be constructed by combining old ones—that is, building new controls out of other controls.

Avalon will remove previous limits on the depth of the control tree that can be used for a form. Microsoft says this will enable the creation of applications requiring a high degree of customized controls. While it's true that the easing of such size restrictions should make software reuse easier, some wags may say that it will also just encourage ever-more-bloated kluges.

Regardless, Avalon will also implement an application model fitted with new navigational capabilities, which will make it easier for developers to move between pages within an application. This, again, is of a piece with the Web-browsing paradigm that has so influenced the thinking of Longhorn's architects.

COMPATIBILITY

Perhaps more than anything else, the salient feature of Longhorn is that the industry's move to 64 bits allows Microsoft to overhaul its systems software sans the historical restriction of full legacy support.

"There are certain features that we're able to do in Longhorn that are difficult to do in backward-compatible operating systems," John Shewchuk, Microsoft architect for Longhorn's Indigo Web services component, told me in a telephone conversation. These include features such as support for next-generation messaging protocols. "One of the killer features is the ability to do direct memory access from network cards," Shewchuk said.

Microsoft's strategic thinking is evident. While XAML

is taking a leading role in terms of what developers see, Shewchuk and his team are architecting an underlying foundation that will speed the messaging traffic required to conduct e-commerce-based Web services.

"We've written very hand-crafted code to provide a full Web services communications model," Shewchuk explained. "This is a high-performance, OS-level messaging-processing engine."

Another interesting "big-think" element of Longhorn is that Microsoft apparently has an inkling of turning Web services into a new paradigm for distributed computing. That's because the message traffic that's the defining element of such services also constitutes a natural way to control geographically dispersed networks of computers.

"We think this service-oriented architecture [SOA, Microsoft jargon for Web services] can serve as the basis of distributed computing," Shewchuk says. "If you were to look inside a computer today, those [Web] services aren't based on a bus architecture, but on a Web-based messaging model. So you could think of your computer as a collection of autonomous components that communicate via SOA."

Shewchuk hopes Indigo will enable a new paradigm, under which programmers will use the new Longhorn programming model to access SOA, or Web services.

That Microsoft programming model, which it calls WinFX, will use CLR to bring together languages and APIs that heretofore had to be handled separately.

CLR is the nuts-and-bolts mechanism enabling the operating system, database, and Web-services elements of Longhorn to talk to each other. Perhaps more important in this hacker-conscious age, CLR will place each separate application in its own "sandbox." This marks a more conscious attempt at software security—an area in which Microsoft has been criticized for lacking in the past.

Longhorn's sandboxing will apply Microsoft's SEE (Secure Execution Environment). SEE implements a hierarchy of restricted permissions to limit where applications can run on the local machine, as well as controlling what local resources those apps have access to.

TAKING A TEST DRIVE

As is to be expected with large projects, Microsoft has been somewhat evasive about a release date for Longhorn and its companion components. Regardless of whether the currently expected beta next year and release in 2006 come to pass, software developers can begin kicking Longhorn's tires today. That can be done by experimenting with the latest version of its Visual Studio 2005 (formerly code-named Whidbey) development environment.

Whidbey implements CLR and Microsoft's new paradigm of managed code. Supported programming languages include Visual Basic and C++, as well as Microsoft's home-grown C# and Java-like J#.

Indigo support is not yet in place. The new Visual Studio still works with Microsoft's existing ASP.NET Web services model. On the plus side, the model included introduces a more secure approach for handling client files in Web applications. Microsoft says that the move to Indigo won't break backward compatibility with ASP.NET.

On the tutorial side, Microsoft has released a Longhorn software development kit, which is a collection of code samples and documentation intended to help developers dip their toes into the upcoming operating system and related technologies. ☺

LOVE IT, HATE IT? LET US KNOW

feedback@acmqueue.com or www.acmqueue.com/forums

ALEXANDER WOLFE received his electrical engineering degree from Cooper Union in New York City. A science writer based in Forest Hills, New York, he has contributed to *IEEE Spectrum*, *EE Times*, *Embedded Systems Programming*, and *Byte.com*.

© 2004 ACM 1542-7730/04/0900 \$5.00

RESOURCES

Visual Studio

A beta version of Visual Studio 2005 is available for free download from Microsoft at <http://lab.msdn.microsoft.com/vs2005/>.

Microsoft Longhorn Developer

Preview PowerPoint presentations on Longhorn, Avalon, WinFS, and Whidbey at <http://www.codezone.info/en/tours.aspx>.

WinHEC 2004

Technical presentations, including PowerPoint presentations on Longhorn, Avalon, and Indigo, can be found at <http://www.microsoft.com/whdc/winhec/pres04-tech.mspx>.

Microsoft's Longhorn Software Development Kit (SDK)

Access this SDK at <http://msdn.microsoft.com/Longhorn/understanding/documentation/releases/default.aspx>.

ACM Members experience: Full access to the ACM Guide to Computing Literature is a basic membership benefit!

To subscribe to the ACM Portal,
contact ACM Member Services:

Phone: 1.800.342.6626 (U.S. and Canada)
+1.212.626.0500 (Global)
Fax: +1.212.944.1318
Hours: 8:30 a.m.-4:30 p.m.,
U.S. Eastern Time
Email: acmhelp@acm.org
Mail: ACM Member Services
PO Box 11414
New York, NY 10286-1414 USA

Professional, Student, and SIG members enjoy full and unlimited access to the Guide as a part of their basic membership package!

The Guide offers an enormous bank of over 750,000 bibliographic citations extending far beyond ACM's propriety literature, covering all types of works in computing such as journals, proceedings, books, technical reports and theses. To access the Guide, members must first establish a free ACM web account. www.acm.org/guide

EXPERIENCE THE FULL POWER OF THE ACM PORTAL!

- The **ACM Portal** fully integrates the ACM Digital Library, the Guide, and the Online Computing Reviews Service. www.acm.org/portal
- The **ACM Digital Library** offers over 30 publications including all ACM journals, magazines, and conference proceedings, plus vast archives, representing over 1.2 million pages of text. The ACM DL includes full-text articles from all ACM publications dating back to the 1950s, as well as third-party content with selected archives. www.acm.org/dl
- The **Online Computing Reviews Service** includes reviews by computing experts, plus an interactive discussion forum, and more.

ACM Professional Members can add a full ACM Portal subscription for the low price of just \$99 (USD) per year.*The Portal is included in ACM's Student Portal Packages.

Association for Computing Machinery
The First Society in Computing
www.acm.org

*Special Guide and Digital Library packages are available for Institutions, Corporations and Consortia (Guide access is not included with Institutional Membership). Please contact ACM for more information.

AD24