INTERHACK

Interhack
5 E Long St 9th Fl
Columbus, OH 43215

VOX +1 614 545 HACK
FAX +1 614 545 0076
WEB http://web.interhack.com/

THE NEXT TWENTY-FIVE YEARS IN COMPUTER SCIENCE

C. MATTHEW CURTIN, CISSP

JANUARY 26, 2009

Part of the Franklin University Tech Connect Speakers Series.

The rapid pace of Information Technology has become an article of faith for many. While we are surrounded by new products and new versions of products, all of them rest upon common foundations established decades earlier. A firm grasp of computer science history and theory can make the difference between a frustrating career spent always attempting to catch up and one where "new" things can be quickly and properly classified and assimilated. In the past, a successful career as an information technologists required a solid background in computer science. In the next twenty-five years, this foundation will require the ability to make that science relevant—without watering it down.

Evolution of Technology

Information technology is pervasive, binding our entire world together. Communication is instant, as is access to information, which is being created at ever-increasing rates.

The advance of hardware is immediately evident when comparing present computing technology to that available just a few years ago. Here we will focus on other areas of information technology that have progressed over the past twenty-five years, namely:

- Operating systems—Controlling the hardware;
- Networking—Making the machines communicate; and
- Applications—Making the machines do things.

Developments in these areas have aggregated to revolutionize the way people think about communication and in how they manage their businesses and personal lives. How can professionals avoid becoming overwhelmed by changes and rendered irrelevant? An exploration of these critical areas might prove instructive.

Operating Systems

The slick "new" operating system that keeps my MacBook Pro humming along is MacOS X. That operating system was not developed from nothing, however. Its design and implementation can be traced back to the Mach micro kernel[1] dating from 1986, and FreeBSD dating as far back as 1993.[2] FreeBSD traces its roots back to 4.4BSD from 1986, and back to 4.2BSD from 1982,[3] and to Unix System IV from 1975, which can be followed all the way back to Unics (1969). Unics itself came from work at AT&T Bell Laboratories that grew out of earlier work on the Multics system.[4]

As for the point-and-click graphical user interface made famous by Apple, it didn't first appear on the Mac from 1984, but on the Apple Lisa in 1983,[5] the Three Rivers Computer Company's PERQ in 1982, and the Xerox Star in 1981.[6] Still other graphical environments were developed for research use in this period of time.[7]

Linux advocates will tell you that their favorite "distro"[8] is all the rage, and has all kinds of new stuff in it. Some investigation reveals that Linux is really just an operating system kernel.[9]

User utilities come largely from the GNU System,[10] which dates back to 1984.[11] Graphical utilities for Linux come largely from the X Window System.[12]

No discussion of computer operating systems is complete without mentioning the system everyone loves to hate—Microsoft Windows, which dates back to 1985.[13] Even so, it was a layer of software that ran atop the device driver and interrupt handler MS-DOS. (Calling it an "operating system" would probably be too generous.) When Windows '95 was released detractors referred to its history, defining it as a "32 bit extensions and a graphical shell for a 16 bit patch to an 8 bit operating system originally coded for a 4 bit microprocessor written by a 2 bit company that can't stand 1 bit of competition."[14]

[1] Mike Accetta, Robert Baron, William Bolosky, David Golub, Richard Rashid, Avadis Tevanian, and Michael Young. Mach: A new kernel foundation for unix development. pages 93–112, 1986

[2] Alexander Hars. Working for free? motivations for participating in open-source projects. *International Journal of Electronic Commerce*, 6: 25–39, 2002

[3] M.K. McKusick, K. Bostic, M.J. Karels, and J.S. Quarterman. *The design and implementation of the 4.4 BSD operating system*. Addison-Wesley, 1996

[4] É. Lévénez. Open Systems; and Dennis W. Ritchie. The evolution of the Unix time-sharing system. In Jeffrey M. Tobias, editor, *Language design and programming methodology: proceedings of a symposium held at Sydney, Australia, 10–11 September 1979*, volume 79 of *Lecture Notes in Computer Science*, pages 25–35, Berlin, Germany / Heidelberg, Germany / London, UK / etc., 1980. Springer-Verlag. ISBN 0-387-09745-7. URL http://cm.bell-labs.com/cm/cs/who/dmr/hist.html. Sponsored by the Australian Atomic Energy Commission and the University of New South Wales

[5] E.C. Ciccarelli IV. Presentation Based User Interface. 1984

[6] B.A. Myers. A brief history of human-computer interaction technology. *interactions*, 5(2):44–54, 1998

[7] Rob Pike, Bart Locanthi, and John Reiser. Hardware/software trade-offs for bitmap graphics on the Blit. *Software—Practice and Experience*, 15(2):131–151, February 1985. ISSN 0038-0644

[8] Common Linux parlance for a specific vendor's distribution.

[9] L. Torvalds. The Linux edge. *Communications of the ACM*, 42(4):38–39, 1999

[10] Richard Stallman. The GNU manifesto. *Dr. Dobb's Journal of Software Tools*, 10(3):30ff, March 1985. ISSN 1044-789X

[11] D. Bretthauer. Open Source Software: A History. *INFORMATION TECHNOLOGY AND LIBRARIES*, 21(1):3–11, 2002

[12] D. Pountain. The X Window System. *Byte Magazine*, 14(1):353–355, 336–360, January 1989. ISSN 0360-5280

[13] Myers.

[14] Original source unknown.

Networking

Getting computers to function is useful, but not as much as making them function with other computers. Hence, we have a pervasive network, the Internet, to tie it all together.[15] Surely this all must be new, right? Except that it is built atop version 4 of the Internet Protocol, defined in 1981.[16]

Of course, we use it differently today, with the Web able to do things that the lower level protocols were never designed to address. That's really the job of the Hypertext Transfer Protocol, first defined in 1991.[17] In turn, that protocol delivers content formatted in a markup language derived from the Standard Generalized Markup Language, which appeared in the literature as early as 1981.[18]

Today's Web is far more than text. Even so, multimedia content often appears in formats standardized in the late 1980s and early 1990s.[19] Increasingly, some of that content comes in the form of mobile code like JavaScript, which brings us to the topic of applications.

[15] K. Hafner. *Where Wizards Stay Up Late: The Origins of the Internet*. Simon and Schuster, 1999

[16] J. Postel. Internet Protocol. RFC 791 (Standard), September 1981. Updated by RFC 1349

[17] Tim Berners-Lee. The Original HTTP as defined in 1991. http://www.w3.org/Protocols/HTTP/AsImplemented.html

[18] C. F. Goldfarb. A generalized approach to document markup. *ACM SIGPLAN Notices*, 16(6):68–73, June 1981. ISSN 0362-1340

[19] Edward A. Fox. Guest editor's introduction: Standards and the emergence of digital multimedia systems. *Communications of the ACM*, 34(4):26–29, April 1991. ISSN 0001-0782

Applications

The Web was a huge step forward in the usability of the Internet. Hypertext would make it possible for people to use the infrastructure as a publishing mechanism. In many ways it worked like Apple's HyperCard, released in 1987.[20]

The idea of hypertext, as it turns out, goes back much further. Ted Nelson coined the term in 1963. And while the Web was widely used, it lacked functionality of Nelson's Xanadu.[21]

Additional functionality came from making executable code operate within the user's browser. Sun and Netscape pushed the Java Programming Language and JavaScript as a way to turn the Web into an application platform.[22] It worked using virtual machine technology that was explored since the mid-1960s.[23] A critical feature of the virtual machine would be to provide security, as discussed in the literature as early as 1973.[24]

[20] Computer History Museum. Timeline of computer history: 1987. http://www.computerhistory.org/timeline/?year=1987

[21] William Stewart. Ted Nelson Discovers Hypertext. Living Internet Web site. http://www.livinginternet.com/w/wi_nelson.htm

[22] Netscape Communications Corporation. Netscape and Sun announce Javascript, the open, cross-platform object scripting language for enterprise networks and the Internet. press release, December 1995. URL http://home.netscape.com/newsref/pr/newsrelease67.html

[23] R. J. Adair, R. U. Bayles, L. W. Comeau, and R. J. Creasy. A virtual machine system for the 360/40. Technical report, International Business Machines (IBM), Data Processing Division, 1966

[24] Stuart E. Madnick and John J. Donovan. Application and analysis of the virtual machine approach to information system security and isolation. In *Proceedings of the ACM Workshop on Virtual Computer Systems*, pages 210–224, New York, NY 10036, USA, 1973. ACM Press. URL http://portal.acm.org/citation.cfm?id=803961&coll=portal&dl=ACM

Another important feature of this client-executable code was that it would be platform independent. Java in particular would be compiled to a machine-independent form that would be interpreted at runtime.[25] That basic technology was discussed in the research literature by 1980,[26] and was highly reminiscent of the P-code system.[27]

Uses

How people use all of this technology becomes more important than the technology itself. Surely this is new, isn't it? As it turns out Using the computer for teaching dates back to the early 1970s.[28] The same PLATO system that was used for teaching also proved foundational for computer-based collaboration.[29]

Using the computer for publishing is a common activity now. It began nearly twenty-five years ago. Hewlett-Packard released the first laser printer in 1984, the same year the PostScript page definition language was introduced.[30] The first what-you-see-is-what-you-get program for desktop publishing was released in 1985, as was the PageMaker software from Aldus.[31]

Business

The world of business undoubtedly is different now from what it was. Everyone has Internet email, the foundations for which date at least as far back as 1982.[32]

Electronic Data Interchange (EDI) is saving businesses tremendous amounts of money over using paper. EDI is not new either, dating back to Western Union transmissions in the 1850s. One hundred twenty years later, standards were created to ease EDI even further.[33]

Information technology had been developed well enough in the laboratory that by the mid-1980s, business was able to see the technology as a way to change the way that organizations compete.[34]

[25] *The Java Virtual Machine Specification*. Sun Microsystems, 1.0 beta edition, August 1995. URL http://java.sun.com/doc/vmspec/VMSpec.ps

[26] S. A. Kent. A programmable network virtual machine. *Computer Networks: The International Journal of Distributed Informatique*, 4(3):125–137, June 1980. ISSN 0376-5075

[27] S. Bodilsen O. Dommergaard. A formal definition of P-code. Technical report, Department of Comp. Sci., Techn. Univ. of Denmark, 1980

[28] C. Illinois. Report X-5, January 1972. j 8. Stifle, J. *The PLATO IV Architecture*

[29] IBM. The History of Notes and Domino, November 2007. http://www.ibm.com/developerworks/lotus/library/ls-NDHistory/

[30] POSTSCRIPT *language manual*. Adobe Systems Incorporated, 1585 Charleston Road, P. O. Box 7900, Mountain View, CA 94039-7900, USA, Tel: (415) 961-4400, first edition, revised, September 1984. edition, 1984

[31] Jacci Howard Bear. When was desktop publishing invented? About.com. http://desktoppub.about.com/cs/beginners/f/when_dtp.htm

[32] D. Crocker. STANDARD FOR THE FORMAT OF ARPA INTERNET TEXT MESSAGES. RFC 822 (Standard), August 1982. Obsoleted by RFC 2822, updated by RFCs 1123, 2156, 1327, 1138, 1148

[33] American National Standards Institute Accredited Standards Committee. The Creation of ASC X12. http://www.x12.org/x12org/about/X12History.cfm

[34] F. Warren McFarlan. Information Technology Changes the Way You Compete. *Harvard Business Review*, May 1984. http://harvardbusinessonline.hbsp.harvard.edu/b01/en/common/item_detail.jhtml?id=84308

Information Technology Is Business

I submit to you that Information Technology today is business. We look at information technology and apply it to issues that arise. In our own enterprises, we seek to reduce costs or to open new opportunities: we're supporting the business. In media and consumer space, we are enabling new ways for people to interact with one another and to do whatever it is that they want to do.

> *The best way to predict the future is to invent it. This is the century in which you can be proäctive about the future; you don't have to be reäctive. The whole idea of having scientists and technology is that those things you can envision and describe can actually be built.*[35]

[35] Alan C. Kay. Predicting The Future. *Stanford Engineering*, 1(1):1–6, Autumn 1989. http://www.ecotopia.com/webpress/futures.htm

Technology can be wonderful. It can be exciting. It can open up entirely new worlds of possibilities. What has made people succeed in information technology for the past twenty-five years is not just an understanding of mechanism: it has been about application.

I believe there are two things that will help those of use who pursue work in technology to continue to advance throughout our careers. The first is an understanding of history. By knowing the literature of our field, we will have a far greater vision than that afforded by our own individual memories. The second is an understanding of theory. Computer science rests upon foundations like data structures and algorithms. These are not things that need to be understood "just by programmers." Understanding the underlying theory allows us to see something as the product of its parts, how they relate to one another, and how they work together.

The combined understanding of history and theory allows us to look not just at the issue before us, but to put it in context. It means asking the right question before charging ahead to find an answer.

Bill Gates saw a world of independent software producers and built Microsoft to be a company in that world. It wasn't the quality of the software that put them on top: it was the vision of the future. The Xerox Star, by comparison, was a fabulously forward-looking machine, but it was closed and no one but Xerox could write software for it. Context made the difference. One envisioned a relatively open market for software companies while the other saw things as they were at the time: hardware companies using software to sell their computers.

"Those who cannot remember the past are condemned to repeat it."[36] If history teaches us anything, it is the need to adapt. The history of our field is full of approaches to improve our own productivity and capability. The theory of our field provides us the tools we need to think meaningfully about our opportunities and challenges, and to go about building what we need to continue to drive us forward. Knowing history and theory will equip us to remain eternally relevant no matter how fast our progress.

[36] G. Santayana. *The Life of Reason; Or, The Phases of Human Progress: Reason in Science.* C. Scribner's Sons, 1905

About the Author

C. Matthew Curtin, CISSP, is the founder of Interhack Corporation, a Columbus-based information security and forensic computing firm, aiding executives and attorneys facing challenges and opportunities involving the management of information. He and his team provide security assessments and incident response drills, as well as services to work with data in legal proceedings. Their work is used to find the right questions to ask and the best answers science can provide. Since 1998, Mr. Curtin has maintained a regular academic appointment as a Lecturer at The Ohio State University's Department of Computer Science and Engineering, teaching courses in the Common Lisp programming language and operating systems. He is the author of Developing Trust: Online Privacy and Security (Apress, 2001) and Brute Force: Cracking the Data Encryption Standard (Copernicus Books, 2005).

Bibliography

Mike Accetta, Robert Baron, William Bolosky, David Golub, Richard Rashid, Avadis Tevanian, and Michael Young. Mach: A new kernel foundation for unix development. pages 93–112, 1986.

R. J. Adair, R. U. Bayles, L. W. Comeau, and R. J. Creasy. A virtual machine system for the 360/40. Technical report, International Business Machines (IBM), Data Processing Division, 1966.

POSTSCRIPT *language manual*. Adobe Systems Incorporated, 1585 Charleston Road, P. O. Box 7900, Mountain View, CA 94039-7900, USA, Tel: (415) 961-4400, first edition, revised, September 1984. edition, 1984.

Jacci Howard Bear. When was desktop publishing invented? About.com. http://desktoppub.about.com/cs/beginners/f/when_dtp.htm.

Tim Berners-Lee. The Original HTTP as defined in 1991. http://www.w3.org/Protocols/HTTP/AsImplemented.html.

D. Bretthauer. Open Source Software: A History. *INFORMATION TECHNOLOGY AND LIBRARIES*, 21(1):3–11, 2002.

E.C. Ciccarelli IV. Presentation Based User Interface. 1984.

American National Standards Institute Accredited Standards Committee. The Creation of ASC X12. http://www.x12.org/x12org/about/X12History.cfm.

D. Crocker. STANDARD FOR THE FORMAT OF ARPA INTERNET TEXT MESSAGES. RFC 822 (Standard), August 1982. Obsoleted by RFC 2822, updated by RFCs 1123, 2156, 1327, 1138, 1148.

S. Bodilsen O. Dommergaard. A formal definition of P-code. Technical report, Department of Comp. Sci., Techn. Univ. of Denmark, 1980.

Edward A. Fox. Guest editor's introduction: Standards and the emergence of digital multimedia systems. *Communications of the ACM*, 34(4):26–29, April 1991. ISSN 0001-0782.

C. F. Goldfarb. A generalized approach to document markup. *ACM SIGPLAN Notices*, 16(6):68–73, June 1981. ISSN 0362-1340.

K. Hafner. *Where Wizards Stay Up Late: The Origins of the Internet*. Simon and Schuster, 1999.

Alexander Hars. Working for free? motivations for participating in open-source projects. *International Journal of Electronic Commerce*, 6:25–39, 2002.

IBM. The History of Notes and Domino, November 2007. http://www.ibm.com/developerworks/lotus/library/ls-NDHistory/.

C. Illinois. Report X-5, January 1972. j 8. Stifle, J. *The PLATO IV Architecture*.

Alan C. Kay. Predicting The Future. *Stanford Engineering*, 1(1):1–6, Autumn 1989. http://www.ecotopia.com/webpress/futures.htm.

S. A. Kent. A programmable network virtual machine. *Computer Networks: The International Journal of Distributed Informatique*, 4(3):125–137, June 1980. ISSN 0376-5075.

É. Lévénez. Open Systems.

Stuart E. Madnick and John J. Donovan. Application and analysis of the virtual machine approach to information system security and isolation. In *Proceedings of the ACM Workshop on Virtual Computer Systems*, pages 210–224, New York, NY 10036, USA, 1973. ACM Press. URL http://portal.acm.org/citation.cfm?id=803961&coll=portal&dl=ACM.

F. Warren McFarlan. Information Technology Changes the Way You Compete. *Harvard Business Review*, May 1984. http://harvardbusinessonline.hbsp.harvard.edu/b01/en/common/item_detail.jhtml?id=84308.

M.K. McKusick, K. Bostic, M.J. Karels, and J.S. Quarterman. *The design and implementation of the 4.4 BSD operating system*. Addison-Wesley, 1996.

Computer History Museum. Timeline of computer history: 1987. http://www.computerhistory.org/timeline/?year=1987.

B.A. Myers. A brief history of human-computer interaction technology. *interactions*, 5(2):44–54, 1998.

Netscape Communications Corporation. Netscape and Sun announce Javascript, the open, cross-platform object scripting language for enterprise networks and the Internet. press release, December 1995. URL http://home.netscape.com/newsref/pr/newsrelease67.html.

Rob Pike, Bart Locanthi, and John Reiser. Hardware/software trade-offs for bitmap graphics on the Blit. *Software—Practice and Experience*, 15(2):131–151, February 1985. ISSN 0038-0644.

J. Postel. Internet Protocol. RFC 791 (Standard), September 1981. Updated by RFC 1349.

D. Pountain. The X Window System. *Byte Magazine*, 14(1):353–355, 336–360, January 1989. ISSN 0360-5280.

Dennis W. Ritchie. The evolution of the Unix time-sharing system. In Jeffrey M. Tobias, editor, *Language design and programming methodology: proceedings of a symposium held at Sydney, Australia, 10–11 September 1979*, volume 79 of *Lecture Notes in Computer Science*, pages 25–35, Berlin, Germany / Heidelberg, Germany / London, UK / etc., 1980. Springer-Verlag. ISBN 0-387-09745-7. URL http://cm.bell-labs.com/cm/cs/who/dmr/hist.html. Sponsored by the Australian Atomic Energy Commission and the University of New South Wales.

G. Santayana. *The Life of Reason; Or, The Phases of Human Progress: Reason in Science*. C. Scribner's Sons, 1905.

Richard Stallman. The GNU manifesto. *Dr. Dobb's Journal of Software Tools*, 10(3):30ff, March 1985. ISSN 1044-789X.

William Stewart. Ted Nelson Discovers Hypertext. Living Internet Web site. http://www.livinginternet.com/w/wi_nelson.htm.

The Java Virtual Machine Specification. Sun Microsystems, 1.0 beta edition, August 1995. URL http://java.sun.com/doc/vmspec/VMSpec.ps.

L. Torvalds. The Linux edge. *Communications of the ACM*, 42(4): 38–39, 1999.