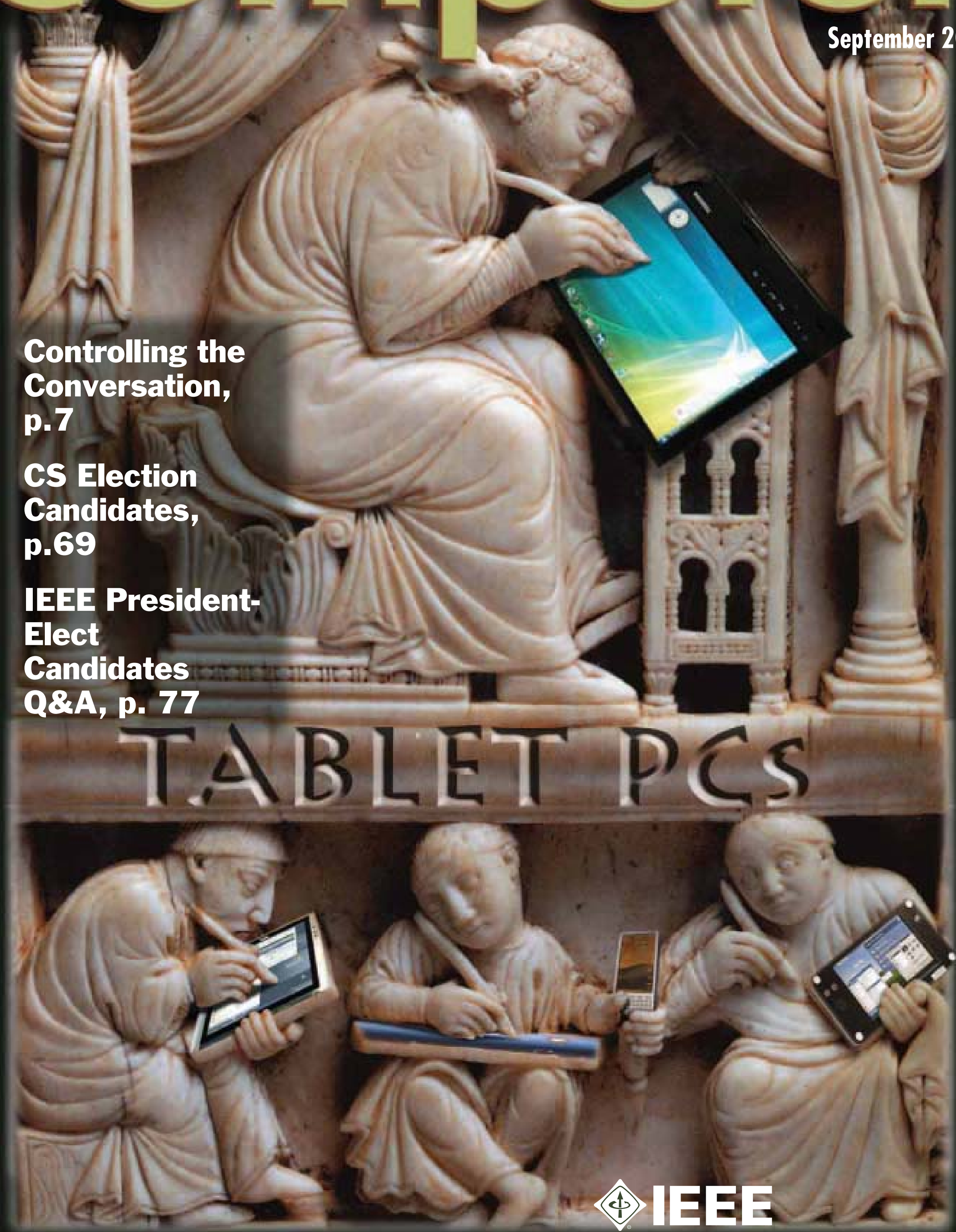
Innovative Technology for Computer Professionals

Computer

September 2007



http://www.computer.org

IEEE

IEEE computer society

Forschungszentrum Karlsruhe
in der Helmholtz-Gemeinschaft

Universität Karlsruhe (TH)
Research University • founded 1825

KIT
Karlsruhe Institute of Technology

Forschungszentrum Karlsruhe, member of the Helmholtz-Association of German Research Centres is one of the largest national laboratories in Germany with major activities in the research areas Energy and Atmosphere, Nano- and Microtechnology and Structure of Matter. Forschungszentrum Karlsruhe and University of Karlsruhe (TH) are currently merging their activities under the roof of the Karlsruhe Institute of Technology, KIT.

KIT invites applications for the joint positions of

Scientific Head of the Institute for Scientific Computing at the Forschungszentrum Karlsruhe

and

Professor (W3) for Parallel and Distributed High Performance Systems at the University of Karlsruhe (TH)

The Institute for Scientific Computing, IWR, of the Forschungszentrum Karlsruhe operates the central computing centre of the Forschungszentrum to serve the IT-demands of its various research programmes within the Helmholtz-Association of German research centres. IWR also runs the Grid Computing Centre Karlsruhe, GridKa, the central European node in the worldwide computing and data grid of the international elementary particle physics activities and is the driving force for the development of Grid computing in Germany.

In the framework of KIT the computing centres of Forschungszentrum and Universität Karlsruhe are merged to the Steinbuch Centre for Computing, SCC, to become one of the leading German scientific computing centres. The head of the IWR will be member of the directorate of the Steinbuch Centre for Computing. The SCC is linked to all research areas of KIT as a service and research partner and is involved in numerous national and international scientific computig projects. Applicants therefore apart from their scientific expertise should have appropriate application experience.

The scientific profile of the candidate in the field of "Parallel and Distributed High Performance Systems" should cover the following topics:

- grid computing
- development and evaluation of appropriate architectures, protocols and policies for grid computing
- quality management and performance management in distributed environments
- decentralized and self-organising solutions
- integrated service-oriented architectures
- virtualisation concepts

Candidates are expected to have internationally recognized scientific standing, the willingness for interdisciplinary collaboration and the ability to lead a large research and service institute. The holder of the chair will give lectures on the above mentioned topics at the KIT.

We are an equal opportunity employer, but as we wish to increase the proportion of females in higher management we especially encourage qualified women to apply for this position. If applicants are equivalently qualified, handicapped candidates applicants will be selected preferentially.

Applications with CV and publication list should be sent to **Prof. Dr. Reinhard Maschuw, Member of the Executive Board of Forschungszentrum Karlsruhe, P.O. Box 3640, 76021 Karlsruhe, Germany** by **30 September 2007**. Information on the candidate's previous experience in research and education as well as reprints of five major publications should be enclosed.

Internet: www.fzk.de

Innovative Technology for Computer Professionals
Computer

Editor in Chief
Carl K. Chang
Iowa State University
chang@cs.iastate.edu

Associate Editors in Chief
Bill N. Schilit
Kathleen Swigger
University of North Texas

Computing Practices
Rohit Kapur
rohit.kapur@synopsys.com

Perspectives
Bob Colwell
bob.colwell@comcast.net

Research Features
Kathleen Swigger
kathy@cs.unt.edu

Special Issues
Bill N. Schilit
schilit@computer.org

Web Editor
Ron Vetter
vetterr@uncw.edu

2007 IEEE Computer Society President
Michael R. Williams
president@computer.org

Area Editors

Computer Architectures
Steven K. Reinhardt
Reservoir Labs Inc.

Databases/Software
Michael R. Blaha
Modelsoft Consulting Corporation

Graphics and Multimedia
Oliver Bimber
Bauhaus University Weimar

Information and Data Management
Naren Ramakrishnan
Virginia Tech

Multimedia
Savitha Srinivasan
IBM Almaden Research Center

Networking
Jonathan Liu
University of Florida

Software
Dan Cooke
Texas Tech University
Robert B. France
Colorado State University
H. Dieter Rombach
AG Software Engineering

Column Editors

Broadening Participation in Computing
Juan E. Gilbert

Embedded Computing
Wayne Wolf
Georgia Institute of Technology

Entertainment Computing
Michael R. Macedonia
Michael C. van Lent

How Things Work
Alf Weaver
University of Virginia

In Our Time
David A. Grier
George Washington University

IT Systems Perspectives
Richard G. Mathieu
James Madison University

Invisible Computing
Bill N. Schilit

The Profession
Neville Holmes
University of Tasmania

Security
Jack Cole
US Army Research Laboratory

Software Technologies
Mike Hinchey
Loyola College Maryland

Standards
John Harauz
Jonic Systems Engineering Inc.

Web Technologies
Simon S.Y. Shim
SAP Labs

Advisory Panel
James H. Aylor
University of Virginia
Thomas Cain
University of Pittsburgh
Doris L. Carver
Louisiana State University
Ralph Cavin
Semiconductor Research Corp.
Ron Hoelzeman
University of Pittsburgh
Mike Lutz
Rochester Institute of Technology
Edward A. Parrish
Worcester Polytechnic Institute
Ron Vetter
University of North Carolina at Wilmington
Alf Weaver
University of Virginia

CS Publications Board
Jon Rokne (chair), Mike Blaha, Doris Carver, Mark Christensen, David Ebert, Frank Ferrante, Phil Laplante, Dick Price, Don Shafer, Linda Shafer, Steve Tanimoto, Wenping Wang

CS Magazine Operations Committee
Robert E. Filman (chair), David Albonesi, Jean Bacon, Arnold (Jay) Bragg, Carl Chang, Kwang-Ting (Tim) Cheng, Norman Chonacky, Fred Douglis, Hakan Erdogmus, David A. Grier, James Hendler, Carl Landwehr, Sethuraman (Panch) Panchanathan, Maureen Stone, Roy Want

Editorial Staff
Scott Hamilton
Senior Acquisitions Editor
shamilton@computer.org
Judith Prow
Managing Editor
jprow@computer.org
Chris Nelson
Senior Editor
James Sanders
Senior Editor
Lee Garber
Senior News Editor
Margo McCall
Associate Editor
Bob Ward
Membership News Editor
Bryan Sallis
Publication Coordinator

Design and Production
Larry Bauer

Cover art
Dirk Hagner

Administrative Staff

Associate Publisher
Dick Price

Membership & Circulation Marketing Manager
Georgann Carter

Business Development Manager
Sandy Brown

Senior Advertising Coordinator
Marian Anderson

Circulation: *Computer* (ISSN 0018-9162) is published monthly by the IEEE Computer Society. **IEEE Headquarters**, Three Park Avenue, 17th Floor, New York, NY 10016-5997; **IEEE Computer Society Publications Office**, 10662 Los Vaqueros Circle, PO Box 3014, Los Alamitos, CA 90720-1314; voice +1 714 821 8380; fax +1 714 821 4010; **IEEE Computer Society Headquarters**,1730 Massachusetts Ave. NW, Washington, DC 20036-1903. IEEE Computer Society membership includes $19 for a subscription to *Computer* magazine. Nonmember subscription rate available upon request. Single-copy prices: members $20.00; nonmembers $99.00.

Postmaster: Send undelivered copies and address changes to *Computer*, IEEE Membership Processing Dept., 445 Hoes Lane, Piscataway, NJ 08855. Periodicals Postage Paid at New York, New York, and at additional mailing offices. Canadian GST #125634188. Canada Post Corporation (Canadian distribution) publications mail agreement number 40013885. Return undeliverable Canadian addresses to PO Box 122, Niagara Falls, ON L2E 6S8 Canada. Printed in USA.

Editorial: Unless otherwise stated, bylined articles, as well as product and service descriptions, reflect the author's or firm's opinion. Inclusion in *Computer* does not necessarily constitute endorsement by the IEEE or the Computer Society. All submissions are subject to editing for style, clarity, and space.

Computer | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | qMags

Computer
Innovative Technology for Computer Professionals

September 2007, Volume 40, Number 9
IEEE Computer Society: http://computer.org
Computer: http://computer.org/computer
computer@computer.org
IEEE Computer Society Publications Office: +1 714 821 8380

Cover design and artwork by Dirk Hagner

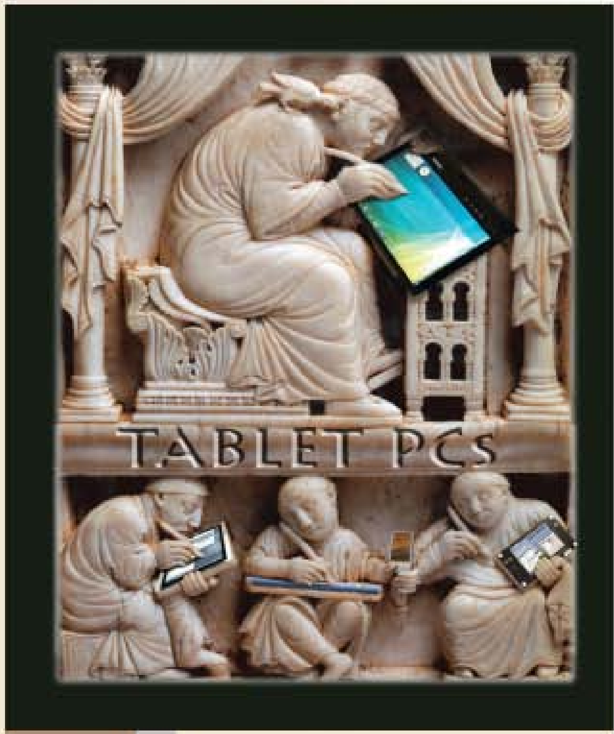
ABOUT THIS ISSUE

Whereas the insurance, healthcare, public safety, and real estate industries were the first users of tablet technology, the next big vertical enterprise to identify the usefulness of Tablet PCs is expected to be higher education. In this special issue, we present five articles that look at tablet technology from the perspectives of application development, research, and instruction.

COMPUTING PRACTICES

COVER FEATURES

Flagship Publication of the IEEE Computer Society



NEXT MONTH:
Embedded Computing

IEEE

IEEE computer society

BPA WORLDWIDE

Copyright © 2007 by the Institute of Electrical and Electronics Engineers Inc. All rights reserved. Abstracting is permitted with credit to the source. Libraries are permitted to photocopy beyond the limits of US copyright law for private use of patrons: (1) those post-1977 articles that carry a code at the bottom of the first page, provided the per-copy fee indicated in the code is paid through the Copyright Clearance Center, 222 Rosewood Dr., Danvers, MA 01923; (2) pre-1978 articles without fee. For other copying, reprint, or republication permission, write to Copyrights and Permissions Department, IEEE Publications Administration, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331.

ARTICLE SUMMARIES

A New Era of Performance Evaluation

pp. 23-30

Sean M. Pieper, JoAnn M. Paul, and Michael J. Schulte

Computers increasingly interact with humans, the physical world, and each other—often simultaneously. Overall performance in this context is a function not only of individual applications, but also of their interactions as they contend for both internal and external resources.

Researchers must describe modern computer usage in terms of scenarios consisting of numerous I/O streams, timing information, and parallel tasks that enter and leave the system, rather than in terms of programs executing in isolation from the physical world and each other. Such use represents a new style of computing, which the authors call *scenario-oriented* to contrast it with other well-established computing styles such as general-purpose and application-specific.

Magic Paper: Sketch-Understanding Research

pp. 34-41

Randall Davis

Sketching is ubiquitous: We draw as a way of thinking, solving problems, and communicating in a wide variety of fields, for both design and analysis.

Unfortunately in today's technology, sketches are dead—they're either graphite on slices of dead trees, or, if captured on a PDA or tablet computer, simply pixels of digitized ink. The Sketch Understanding Group at MIT has been working toward a kind of "magic paper"—that is, a surface that's as natural and easy to draw on as paper, yet that understands what you draw.

Ink, Improvisation, and Interactive Engagement: Learning with Tablets

pp. 42-48

Jeremy Roschelle, Deborah Tatar, S. Raj Chaudhury, Yannis Dimitriadis, Charles Patton, and Chris DiGiano

Socrates' pedagogical use of an early learning technology—improvising an informal sketch in the sand—demonstrates why today's teachers might prefer Tablet PCs over desktop or laptop computers to structure interactive, engaging learning experiences.

Given the central role of teaching practice in learning outcomes, advances in the use of tablets in education will be driven not primarily by technology features but rather by instructional models that reflective educators develop and share with their peers. Communities that form around platforms such as Classroom Presenter and the authors' own Group Scribbles should provide an excellent forum for such advances.

Handwriting Recognition: Tablet PC Text Input

pp. 50-55

James A. Pittman

In specific situations, machine-recognized handwriting serves as a useful alternative for entering small volumes of text. However, practical handwriting recognition isn't easy. Not only must handwriting-recognition systems handle many different shapes and styles for each letter, but humans also commonly produce sloppy script to increase throughput speed.

To support a wide range of writing styles and poorly formed cursive script, the Tablet PC input panel uses a time-delay neural network working with a lexicon. High-end versions of Microsoft's Vista now include Tablet PC software, with an improved handwriting recognizer that supports both personalization and error reporting.

Classroom Presenter: Enhancing Interactive Education with Digital Ink

pp. 56-61

Richard Anderson, Ruth Anderson, Peter Davis, Natalie Linnell, Craig Prince, Valentin Razmov, and Fred Videon

Networked Tablet PCs can be used for a wide range of classroom applications including note taking, interaction with presentation materials, simulations, in-class communication, and accessing outside resources.

The authors have developed Classroom Presenter, a distributed Table PC-based system that supports the sharing of digital ink on electronic slides among instructor, students, and public displays to explore a set of classroom interaction scenarios for enhancing student engagement in class. Initial deployments of the system show that instructors can exploit this technology not only to achieve a wide range of educational goals successfully but also to create a more participatory and collaborative environment.

Facilitating Pedagogical Practices through a Large-Scale Tablet PC Deployment

pp. 62-68

Joseph G. Tront

In the fall of 2006, the Virginia Tech College of Engineering became the first public college of engineering to require all incoming students to own a Tablet PC. The college has developed an implementation process for using the devices that includes computer acquisition, faculty training, infrastructure modifications, and multiple assessments for program evaluation. Initial results of this groundbreaking program are positive, showing measurable improvements in pedagogical practices that are ultimately expected to lead to learning improvements.

Published by the IEEE Computer Society

Joint Call for Contributions

The 8th International Symposium on Applications and the Internet
The 32nd Annual International Computer Software and Applications Conference

COMPSAC & SAINT 2008

Turku, Finland, July 28 – August 1, 2008

Sponsored by

IEEE | IEEE computer society | IPSJ

IPSJ/IEEE Signature Conferences on Software and Internet Technology & *Applications*

Platinum Sponsor

NOKIA Connecting People

COMPSAC is a major international forum for researchers, practitioners, managers, and policy makers interested in computer software and applications. It was first held in Chicago in 1977, and since then it has been one of the major forums for academia, industry, and government to discuss the state of the art, new advances, and future trends in software technologies and practices. The technical program includes keynote addresses, research papers, industrial case studies, panel discussions and fast abstracts. It also includes a number of workshops on emerging important topics.

The creation of trustworthy and dependable software spans all aspects of software engineering and modern-day software technology and its applications. COMPSAC is a unique forum to bring together these facets and their major stakeholders. Building on the trustworthy, secure, and dependable software themes of highly successful recent COMPSAC conferences, the technical theme for the 32nd COMPSAC is **INTEGRATING THE DISTRIBUTED WORLDS.**

Multidisciplinary work, research and development software prototypes, industry-university collaborations, all based on new emerging and critical technologies will be of particular interest to this conference. Check http://www.compsac.org for detailed submission information.

SAINT is a major joint international conference co-sponsored by the Information Processing Society of Japan (IPSJ) and IEEE Computer Society. The Internet has been revolutionizing and changing the way we communicate, access information and conduct businesses. Today, the Internet is evolving into a pervasive and highly distributed ecosystem of connected computers, mobile devices, sensors, home appliances, and a variety of other Internet devices. This evolution is leading to deep intertwining of the physical and digital worlds, raising new research agendas and challenging our understanding of the requirements for networked computing and Internet based Service Oriented Architectures. SAINT focuses on emerging research challenges raised by the pervasive Internet, and its future applications and services, as well as their enabling technologies. The symposium provides a forum for researchers and practitioners from the academic, industrial, public and governmental sectors to share their latest innovations on Internet technologies and applications. The technical theme for the 8th SAINT is **THE PERVASIVE INTERNET.**

Areas of particular interest to SAINT include, but are not limited to: **(A) Internet Application Areas:** Content Management, Content Delivery, Web Services, E-Business, Collaboration, Internet communities, Wireless Access, Ubiquitous and Pervasive Computing; and **(B) Enabling technologies for the Internet:** Software Architectures, Standards, Internet Security, Network and Protocol Architectures, Information Appliances. Check http://www.saintconfrence.org for detailed submission information.

While COMPSAC and SAINT will be co-located, their accepted papers will be published separately in the electronic conference proceedings by the IEEE Computer Society, indexed through INSPEC and EI Index, and automatically included in the IEEE digital library. Plans had been made to forward select papers to archival journals for publications.

IMPORTANT DATES for COMPSAC

- **December 8, 2007:** Workshop/Panel Proposals Due
- **January 15, 2008:** Abstract due
- **January 31, 2008:** Full Paper Submission Due
- **March 1, 2008:** Workshop Paper Submission Due
- **March 15, 2008:** Author Notification
- **March 30, 2008:** Workshop Paper Notification
- **April 30, 2008:** All Final Manuscript and Author Pre-registration Due

IMPORTANT DATES for SAINT

November 15, 2007: Paper Submission Due
November 15, 2008: Panel Proposals Due
November 15, 2007: Workshop Proposals Due
February 15, 2008: Author Notification
March 1, 2008: Workshop Paper Submission Due
March 30, 2008: Workshop Paper Notification
May 1, 2008: All Final Manuscript and Author Pre-registration Due

PROGRAM DEVELOPMENT INQUIRIES

COMPSAC Program Co-Chairs:
Hannu Tenhunen, University of Turku, Finland, at hannu@ele.kth.se
T. Ming Jiang, National Chung Cheng University, Taiwan, at mjiang@ee.ccu.edu.tw
Frank Liu, Univ. of Missouri-Rolla, USA, at fliu@umr.edu

SAINT Program Co-Chairs:
Henry Tirri, Nokia Research and University of Helsinki, Finland, at Henry.Tirri@nokia.com
Yasuo Okabe, Kyoto Univ., Japan, at okabe@i.kyoto-u.ac.jp

GENERAL INQUIRIES

For further information about COMPSAC, please contact:
Carl Chang, Chair, Standing Committee, at c.chang@computer.org

For further information about SAINT, please contact:
Katsuyuki Yamazaki, General Chair, at yamazaki@ieee.org
Elisa Bertino, General Chair, at bertino@cs.purdue.edu

CONSCIOUSNESS AND COMPUTERS

In *Computer*'s July 2007 issue, Neville Holmes, who usually provides some references to back up his position, wrote what appears to be a secular antireligious article, without any justification of his claims (The Profession, "Consciousness and Computers," pp. 100, 98-99).

Historical documents indicate that animals have long been known to have self-awareness. Our cat was not only self-aware but showed emotions such as jealousy and love.

Holmes also erroneously claims that complex animals evolved from simple ones. Clearly, this could never happen, as the second law of thermodynamics, as well as all observational experiments, have shown.

Science requires the ability to confirm results with independent experiments. Evolution cannot be confirmed. It is merely the faith of the antireligion faction that believes in Darwin, instead of logic and facts.

Humans have been degrading over the centuries and are now more prone to disease and other problems. No new senses have evolved, and no new species have emerged. And mouthing a stallball line about just needing more time would be neither logical nor scientific.

A far better explanation is that all life is based on a preprogrammed DNA that has a calendar/clock capability and instantiates the existing potential in different forms at different times and circumstances. That is at least plausible.

Should Holmes or others then claim that that would invoke God, which they do not believe in, they would still have the problem of explaining where their first simple animals came from. Did they appear from nothingness following a random flash and boom because in the infinite duration of time events of probability zero could happen? They certainly did not "evolve" to become more complex.

Since none of this can be proven logically, the simplest solution—that is, programmed DNA—would be the better option based on the logic of Occam's razor.

William Adams
williamadams@ieee.org

Neville Holmes responds:

I'm not clear why I am taken to be an atheist. I am aware of myself as an agnostic with a tendency to numinism.

On evolution, I have found Howard Bloom's book *Global Brain* (howardbloom.net) to be fascinating reading, but it's somewhat overloaded with formal citations.

ERRATUM

In my article titled "Supporting Resource-Constrained Collaboration Environments" (The Profession, June 2007, ppp. 108, 106-107), there is an error in Figure 3. The date in the file name is shown as 060704 when it should be 070704.

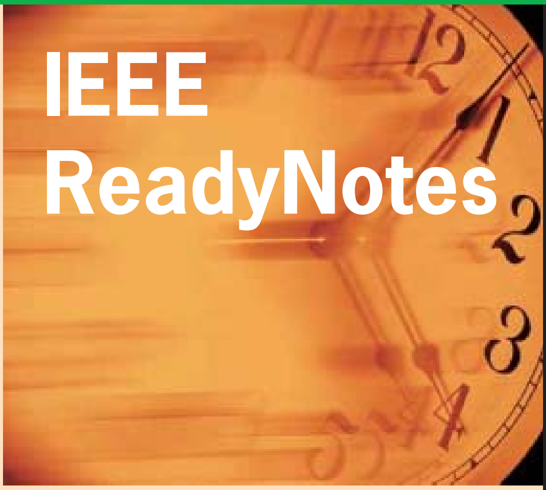
Sean Price
sean.price@sentinel-consulting.com

We welcome your letters. Send them to computer@computer.org.

IEEE Software Engineering Standards Support for the CMMI Project Planning Process Area

By Susan K. Land
Northrop Grumman

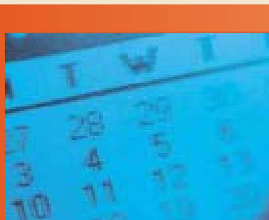
Software process definition, documentation, and improvement are integral parts of a software engineering organization. This ReadyNote gives engineers practical support for such work by analyzing the specific documentation requirements that support the CMMI Project Planning process area. $19
www.computer.org/ReadyNotes



IEEE | IEEE computer society

Controlling the Conversation

David Alan Grier
George Washington University



American automakers adopted computer controls to deal with complicated engineering problems only when pushed by forces beyond their control.

It's a truth, universally acknowledged, that when any group of men find themselves in want of a subject of conversation, they may find a rich topic for discussion in a short, simple phrase: "So what are you driving these days?"

Beginning with these seven words, generations of men have found an unending source of interesting topics. By talking about automobiles, men can express thoughts they otherwise would never put into words. They talk about love and hope, their fears for this world, and their embarrassment at weaknesses. Cars provide them a way to talk about fashion and to comment on the vast network of connections that holds the male world together.

During high school, when I briefly worked at an auto dealership, my boss took me aside and gave me some words of wisdom. "Women like to hear men talk about cars," he confided, "because women believe that the way a man treats his car shows how he will treat his wife."

Though I have spent many an hour in the company of men discussing automobiles, I have never really understood why this subject has such a pull on the male mind. I have speculated that the Y chromosome might have some field devoted exclusively to the theory of the motor vehicle, that our education of boys instills a love of speed and motion, or that men have a fundamental character flaw that those advertising companies that market cars have ruthlessly exploited.

All or none of these possibilities might be true. All we know is that the automobile has, for more than a century, had a central place in modern, industrialized culture.

CAUTIOUS INNOVATION

The automobile is not only the most sophisticated mechanical device that most individuals will operate, it is also the most computerized. The modern automobile contains computers that control almost every aspect of operation: engines, transmissions, brakes, heating, ventilation and air conditioning, entertainment, navigation. This technology became a substantial part of the automobile at the start of the 1980 model year, when the Big Three American automakers had to confront new engineering problems and rethink how their firms interacted with the men and women who maintained their products.

They had to design cars for mass production on assembly lines, operation by people with little technical knowledge, and maintenance by a network of mechanics over whom they had little control. The American automakers were cautious innovators. They adopted computer controls only when pushed by a variety of forces, including inexpensive calculator chips, air-pollution and oil-conservation standards, a war in the Middle East, and a weak American president.

A BLEAK YEAR

In September 1973, shortly after release of their 1974 models, American automakers began to sense that something had changed in the auto market. Normally, the first week of the new model year brought a surge in sales. "Last year we sold a couple dozen cars on the first night," said one salesman. "This year we sold two."

No one expected 1974 to be a repeat of 1973, which had been the best year in history for Detroit's Big Three. However, the 1974 models came with features that worried the sales staffs. They included new safety features that did not sit well with some consumers. The bumpers stiffened to absorb shock. An electrical interlock prevented the engine from starting unless all the seatbelts were fastened. Antipollution systems scrubbed pollutants from the exhaust at the cost of decreased gas mileage.

In October 1973, the poor start to the sales year became worse. On 6 October, Egypt and Syria invaded Israel to start the Yom Kippur war; 11 days later, the major Arab oil-producing states embargoed oil destined for the US. The next six weeks, which proved to be the start of one of the harsher winters in recent history, became a time of long gas lines and short tempers, of cold houses and rising anxieties.

NEW STANDARDS

When spring returned and the embargo ended, the damage had been done. The US Congress was ready to take rather forceful steps to increase energy sources and reduce oil consumption. In particular, they were ready

IEEE Annals of the History of Computing

The *IEEE Annals of the History of Computing* publishes stories about the development of computing institutions and computing technologies. The current issue has a story on the Burroughs Corp. computers, which has nothing to do with automobiles, but the company was at least headquartered in Detroit.

to require fuel-efficiency standards for all automobiles sold in the US.

The automobile industry disliked the proposed standards and tried to thwart them. After the bill passed Congress, automakers urged President Gerald Ford, formerly a congressman from Michigan, to veto the legislation.

"Why don't we outlaw cancer?" complained a Chrysler executive. "Then any doctor who doesn't cure cancer in five years should have his license picked up."

In other times, President Ford might have been willing to veto the bill, but he held a weak hand: He was an unelected president without national support. Thus, he signed the bill. The best that Ford could do was to relax a few of the safety standards. The new law gave the automakers five years to improve the gas mileage of their cars by 40 percent.

MEETING THE CHALLENGE

In that bleak winter, the automakers knew that computers would likely provide the technology to meet the efficiency standards. For three or four years, their engineers had been discussing the processor chips that were being used in pocket calculators. They speculated that these chips could be used in future automobiles as precision engine controllers that would improve fuel economy and simultaneously reduce pollution, but they also acknowledged that several decades might pass before such controllers became common.

In 1974, the automakers used engine controls that differed little from the systems they had deployed in the 1940s. These systems were built from a combination of mechanical and electrical parts: the carburetor, distributor, and spark advance. Because these controllers were relatively crude, they rarely guided the engines into an optimal state. They would often provide slightly too much fuel to the cylinders or ignite the spark a little too late to make the best use of the gasoline. Still, they were simple to make and well understood by thousands of mechanics and car owners.

New systems

The first computer-controlled ignition system to reach the market was MISAR, a General Motors product that was installed on the 1977 Oldsmobile Toronado. The Toronado was a luxury vehicle, and the customers who purchased it were able to afford the costs of a new technology. Furthermore, the car needed to be redeemed from a spate of bad publicity. At the start of the decade, the US government had identified it as a highly wasteful vehicle that could travel only 8.6 miles on a gallon of gas.

The "MISAR system provides almost infinite [control] flexibility to meet the requirements for fuel economy, engine performance, and exhaust emissions," reported a description of the system's hardware components. Using a microprocessor and electronic sensors, the system monitored factors that influenced engine operations: air/fuel ratio, intake air temperature, intake manifold air pressure, engine temperature, engine speed, throttle position, unburned fuel in the exhaust. The system converted this information into commands that controlled the engine's timing and the amount of fuel injected into the cylinders.

General Motors followed the MISAR project with a second, more sophisticated computer system and declared, with perhaps too much optimism, that computer-controlled automobiles were "on the threshold of ending all auto pollution."

Ford Motor Co. followed with a digital system of its own. The company bragged that Ford "control-system engineers are bridging the gap between engine system engineers and semiconductor designers to create microcomputers, sensors, and actuators with the latest technology."

That new technology required the automakers to spend the winter of 1978-1979 retooling their manufacturing facilities. These efforts acquired additional urgency when a second oil crisis, this one sparked by political unrest in Iran, brought a return of long lines at gas stations. The winter also uncovered supply and maintenance problems that the automakers would need to solve before they could deliver cars with computer controls.

Processors and procedures

The American semiconductor industry could not provide a sufficient number of processors to the automakers. "It's a worldwide supply situation," reported a General Motors engineer, "because the industry is really going to be saturated with orders."

General Motors turned to European suppliers for their chips, giving these firms a big boost in business. Ford purchased integrated processors from Japan's Toshiba, which expanded its production to meet the demand.

The automakers also realized that they would need an entirely new set of maintenance procedures. The cars "are getting more complicated," reported a maintenance manager at General Motors. There "isn't any way you're going to go out and find 30,000 PhD electronics people to service cars at the dealerships." Ford's management concurred with this assessment. The "do-it-yourselver is being closed out on such things as routine engine tune-ups and carburetor adjustments," said a company representative. He observed that "changing and adjusting ignition points, for example, used to be a ritual of auto maintenance. But this chore has vanished due to electronic ignition systems that are now used industry-wide."

While the assembly lines began to build the first of the new models, the maintenance departments wrote new manuals, retrained the company staff of mechanics, and deployed a new diagnostic system. These systems were also small computers that read data from sensors or directly from the memory of the car's processors.

The new equipment was in place by September 1979, the month that the new 1980 cars, offering more fuel efficiency and cleaner air emissions, reached the showroom floors. According to a General Motors engineer, the new automotive computers represented the "biggest technological advance since the 1950s" for the auto industry. "We would not have installed them without the government's mandate for better fuel economy and cleaner emissions," he declared.

Most of the 1980 models had only a few chips that controlled the engine's operations. More expensive cars had chips that controlled the brakes, door locks, and air-conditioning system. As the models progressed through the 1980s and 1990s, the cars acquired more and more processors until digital electronics controlled nearly every aspect of the machine.

I was a little surprised when Chip asked, "Do you want to see my hundred-dollar switchblade?" It seemed like a strange question because I thought the two of us had been talking about cars.

Chip, a truck driver out of Sioux Falls, S.D., had just purchased a new van and was showing me every piece of digital technology it contained. It had a computer for its engine control, another for nonskid brakes, a third for headlight control, another for the cab heater, and the list continued with a digital radio and a navigation system. We had been happily discussing all these little computers when Chip unexpectedly asked me about seeing his switchblade.

Before I could answer, Chip reached into his pocket, extracted a black plastic rectangle and slapped it on the table. I picked it up, turned it over, and was a little surprised when something sprung from one side.

"It's a computer," he said. Chip then described how he had needed a spare key for the van and had ordered one from the factory for a cost of $100. When it arrived, he had asked the van dealer if the processor needed to be programmed. After assuring Chip that the key should be ready to go, the dealer followed him to his van, where Chip put the key in the ignition and tried to start the vehicle.

Nothing happened. No clicks. No buzzes. Nothing.

Chip then took his old key from his pocket and tried to start the van. Again, nothing happened. At first, the two of them thought something might be wrong with the van, but they quickly realized that the security system had taken control of the vehicle. The system didn't find the right security code on the key, so it refused to start the engine.

After identifying one bad key, it would allow no one to start the engine, not even someone with the correct key.

The slightly embarrassed dealer quickly had the security system reset, but the incident gave Chip a story to tell and issues to ponder. He truly enjoyed the van and enjoyed discussing how much it improved his daily work. At the same time, he also recognized that the vehicle's processors collected a lot of operational data. "It tracks the oil in the crankcase," he said. "If you use the wrong brand, it will know."

Chip and I spent much of that day together, running errands and talking about his new van. For once, I had the feeling that we were something more than two guys talking about cars. I was suspicious that the van might be monitoring our words and, at some point, might control the conversation.

David Alan Grier *is the editor in chief,* IEEE Annals of the History of Computing, *and the author of* When Computers Were Human *(Princeton University Press, 2005). Grier is associate dean of International Affairs at George Washington University. Contact him at grier@gwu.edu.*

Computer Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

32 & 16 YEARS AGO

1975•1991•1975•1991•1975•1991•1975•1991

SEPTEMBER 1975

SYSTEM PRINTERS (p. 15). "Most DP managers know that systems come with printers. They also know that when ordering a system, one must analyze the printed output requirements and order a sufficient number of printers to handle the load on schedule. The units are assumed to be reliable and, in view of the cost of the large volume of paper processed each month, not overly expensive. In most medium- and large-size installations, the cost of the paper exceeds the cost of the printers.

"The system designer's view of printers is not very exciting either. He views them much as a logic designer views power supplies. Most logic designers know power supplies are expensive, come with several different ratings, and are heavy. The system designer knows these same things about his printer. He also knows he must make provision to attach one or more of them because his system is not complete without printing capability. (Hard copy seems to have a soothing effect on the end user.)

"The printer designer, of course, believes he is about to revolutionize the industry. …"

IMPACT PRINTERS (p. 29). "The modern impact printer tends to be noisy, expensive, mechanically complex, and requires frequent maintenance and service. In spite of these drawbacks, however, it has thrived as a computer peripheral mainly due to the absence of a suitable alternative. It will continue to thrive until a non-impact printer is available which will, as a minimum, offer the equivalent of the accepted features such as plain paper, multiple copies, and equivalent print speed and ease of use."

"The ideal or at least the acceptable printer is still some time away. Meanwhile, high speed impact printers everywhere are cranking out data in tons of unwieldy, oversized, and expensive paper. Data processing installations are increasing in number, output is increasing with increasing printing speeds, and above all the cost of paper is increasing."

NONIMPACT PRINTERS (p. 41). "Although some market success has been attained in certain specialized application areas, nonimpact printing systems have not yet revolutionized computer printing. One reason is that the many conventions and practices which have been developed for computer printing have been adapted to the strengths and limitations of impact printers. In addition, there is a huge investment in host computer software which supports impact printers. The cost of special papers which are required by many nonimpact printers is another drawback, especially if the print volume is high.

"On the other hand, xerographic nonimpact printing permits new capabilities, including better print quality, as well as those that improve total system effectiveness by improved throughput, reduced paper cost, reduced need for preprinted forms, and reduced or improved post-print operations."

ELECTROSTATIC PRINTER (p. 57). "The Honeywell Page Printing System has as its basis a new printing technology which uses a highly reliable, high-speed printing technique that has been adapted to the environment of computer output printing. The system design has been conceived to offer the user a broad capability to significantly improve the management of his printing facility. The design integrated a broad range of diverse technologies—electrostatic charge imaging, liquid toner hydraulic flow control, heat transfer, vapor recovery and air flow, mechanics of paper web and sheet transport, digital electronics for control and data processing, system interface, software design, human engineering, machine aesthetics, system integration, component reliability, and failure compensation and diagnosis. …"

OPTICAL MARK READER (p. 58). "A new optical mark reader system that checks and grades test papers at the rate of 7000 answer sheets an hour has been introduced by National Computer Systems of Minneapolis.

"The scanner system, called the Sentry 7020, reads and scores both sides of each sheet simultaneously, using 16 levels of discrimination to tell the difference between a valid mark and a smudge or erasure. It can handle true/false or multiple choice questions and can allow multiple correct answers to a question if desired.

"Besides checking and grading test papers, the system can handle a wide range of other tasks—such as class and school attendance records, registration, schedules, and surveys—where a mass of data must be compiled rapidly and accurately. Response sheets are custom-designed to fit the school's needs."

COMPUTERIZED PEN (p. 66). "A computerized ballpoint pen that 'knows' if a forger is using it has been developed by engineers at Stanford Research Institute (SRI), Menlo Park, Calif.

"Resembling an ordinary ballpoint pen, the pen is wired to a computer system that recognizes the patterns of pressure and motion that the signer makes as a signature is written. The system requires a would-be forger to duplicate the signer's dynamic motions as well as the physical appearance of the signature.

"'It clearly separated the "good guys" from the "forgers" among a dozen of us here in the lab,' says the pen's inventor, Dr. Hewitt Crane."

PDFs of the articles and departments of the September 1991 issue of Computer *are available through the Computer Society's Web site, www.computer.org/computer.*

Editor: Neville Holmes; neville.holmes@utas.edu.au

SEPTEMBER 1991

THE NEXT DECADE (p. 15). "… The challenges ahead are enormous. Can we continue to afford major advances in microelectronic technology such as 0.1-micron feature sizes and 100 million devices on a chip? Are truly distributed systems consisting of heterogeneous nodes interconnected through gigabit-per-second networks possible? Can operating system technology be advanced to the point where we have true automatic system management?"

COMPUTER ARCHITECTURE (pp. 30-31). "… the 1990s could end with the entire line of computer offerings affected by fundamental limitations on clock speed. If the clock-speed barrier cannot be overcome, then this decade will mark the end of the sustained increases in uniprocessor performance of the past 40 years.

"How can the designer overcome this obstacle? Everything has to shrink in dimension. The ideal is to put everything on one chip or one wafer and avoid long metal interconnections completely. Where it is impractical to use wafer-scale technology, a compromise is to use very dense packaging such as multichip modules to minimize the length of metal interconnections between VLSI chips. …"

"One possible alternative is optical interconnections. … Optical technology is still young and prohibitively expensive today for parallel interconnections inside a computer. This is unlikely to change in the near term, but we may see parallel optical buses by the end of this decade or early in the next, if progress goes well."

USER INTERFACES (p. 56). "Future developments emphasize higher level interaction with three-dimensional display; multimedia and hypermedia presentations with color, animation, and video; support for group interaction; and intelligent agents. … Instead of multiple-keystroke characters or mouse-button selections to signify particular nouns and verbs, users will employ combinations of explicit commands and more implicit, more natural gestures. For example, it will become possible to express succinctly the following directive to an intelligent design assistant: 'Increase the rounding of the corners on the dinner plates Sandy designed last week so that they fit our production methods for Rosenthal china and meet hotel restaurant durability requirements.'"

COMPUTER GRAPHICS (p. 65-66). "… As far as the workstations of the future go, Jim Clark of Silicon Graphics may have the best crystal ball around. Today, SGI's high-end VGX system can display a million polygons per second. The single-processor version has 30 MIPS/5 Mflops computing power, which extends to 200-plus MIPS/30-plus Mflops for the eight-processor version. At a recent CEO panel, 'Visualization—The Next Decade,' Clark predicted that through the use of multi/micro RISC processors, high-end workstations ($150,000) will approach Gflop … performance in the next two to three years. He also made some interesting predictions for the low end. He described the multimedia machine of the future as a device integrating the capabilities of (1) full 3D, interactive, real-time computer graphics, (2) full motion video, (3) image processing capabilities, (4) fully integrated geometric text, and (5) full compact-disc digital sound. He predicts that this will be available in a portable notebook-sized device with motion sensors in the next five to ten years for under $5,000."

DISTRIBUTED COMPUTING (p. 75). "With the advent of NREN [the National Research and Education Network], this decade will see the start of a planetwide network of parallel and distributed computing systems linked by ultrahigh-bandwidth optical communications. By the end of the 1990s, ultrafast, highly parallel optical computers will start to replace electronic supercomputers and will speed switching of optical data communications. The result will be a sea of general-purpose and personalized computer systems with islands of very fast specialized computers and massive databases. Mankind will start to paint its dreams on this global computing canvas."

SOFTWARE PATENTS (p. 113). "… Similarly, proponents claim that software patenting furthers software technology by making it more profitable to develop new software. Actually, it creates a minefield in which every design decision runs the risk of violating some obscure or ill-advised patent. In fact, several polls have indicated that the majority of software professionals oppose software patenting, on the grounds that the purported profit incentives are neither necessary nor beneficial in furthering the development of new software technology.

"IEEE, as an organization dedicated to the support of professionals in the service of society, should lend its weight to this position as well."

KANJI (p. 126). "Motorola has announced that its Delta Series 3000 computers now support *kanji* through their System V/68J operating system, which is the company's implementation of Unix System V Release 3. The operating system was developed by Nippon Motorola, the Motorola Computer Group, and AT&T's Unix System Laboratories in Japan. The Delta 3000 series uses 32-bit 68030 complex instruction-set computers.

"The computers write text by interpreting a combination of *kana* (syllabic characters that represent the sound of the Japanese language) that represent the same syllables as a *kanji* pictograph. All *kana* characters can be placed on a standard computer keyboard. Since a combination of *kana* characters can represent several different *kanji* characters, the operating system selects the proper *kanji* character for the context of the sentence."

EARLY REGISTRATION ENDS SEPTEMBER 21, 2007

IEEE VISUALIZATION 2007

October 28-November 1
Sacramento, CA, USA

Visit the web site for conference registration and hotel reservations

VIS 07

CALL FOR PARTICIPATION

Vis 2007 is the premier forum for visualization advances in science and engineering for academia, government, and industry. This event brings together researchers and practitioners with a shared interest in tools, techniques, and technology. The conference will include an exciting and informative collection of workshops, tutorials, papers, panels, demonstrations, posters, and exhibitions. We invite you to participate by sharing your research, insights, experience, and enthusiasm in Sacramento, California.

vis.computer.org/vis2007

CO-LOCATED WITH VIS 2007 ARE THE FOLLOWING SYMPOSIA:

InfoVis 2007: IEEE Symposium on Information Visualization
Chair: John Stasko, *Georgia Institute of Technology*
http://conferences.computer.org/infovis/infovis2007

VAST 2007: IEEE Symposium on Visual Analytics Science and Technology
Chairs: John Dill, *Simon Fraser University* & Bill Ribarsky, *University of North Carolina, Charlotte*
http://conferences.computer.org/vast/vast2007

For questions, email:
VIS: info@vis.computer.org
InfoVis: infovis@vis.computer.org
VAST: vast@vis.computer.org

Conference Chairs:
Kenneth Joy, *University of California, Davis*
Amitabh Varshney, *University of Maryland*

Sponsored by the IEEE Computer Society Visualization and Graphics Technical Committee.

IEEE vgtc · IEEE · IEEE Computer Society

Motion Capture Moves into New Realms

Brett Allan King and Linda Dailey Paulson

When many people think of motion-capture technology, they think of people wearing reflective markers and having their movements captured on camera for use in modeling figures created by animation or computer graphics.

Motion capture is frequently associated with movies or videogames. For example, most of the passengers seen strolling the ship's deck in the movie *Titanic* were created with computer graphics, their movements made lifelike via motion capture. And games such as Electronic Arts' popular *Madden NFL* series use the technology to make characters' movements more realistic.

Now, motion technology is moving in some new directions. Researchers are working on techniques that don't, for example, require placing hundreds of markers on models.

Also, motion-capture applications are becoming less expensive and more readily available. Fields as diverse as medicine, manufacturing, and advertising are thus beginning to utilize the technology more often.

Vendors are also working on motion-based approaches to enable people to use gestures to navigate computerized visual presentations or issue commands to devices.

Globally, although not in the US, nonentertainment motion-capture applications now account for more revenue than entertainment-related ones, noted Julian Morris, deputy chair and chief technology officer of the Oxford Metrics Group, a leading motion-capture and movement-analysis vendor.

Nonetheless, motion capture still faces challenges, such as output precision and ease of use.

INSIDE MOTION-CAPTURE TECHNOLOGY

Motion capture has been the subject of research for several decades. It has been pursued for its potential advantages in areas besides moviemaking. For example, in manufacturing, motion-capture technology places simulated users in virtual products to test them, which is less expensive than placing real people in real products and having to build and rebuild physical prototypes.

Motion-capture displays are also an engaging form of advertising, said John Payne, president of Monster Media, a digital-advertising-technology vendor.

History

Motion capture's first uses, beginning in the mid-1970s, included Polhemus Navigation Services' tool for magnetically tracking military pilots' helmet movements to help them lock in on targets.

The first 3D motion-capture product, delivered in 1980 to the University of West Virginia and Children's Hospital Boston, was the Oxford Instruments Group's Vicon System, which captured images of children's gaits to enable physicians to detect disabilities.

Motion capture wasn't tried in movies until 1990, with the film *Total Recall*. The Motion Capture Society, a professional organization that promotes the technology, considers 1992's *Lawnmower Man* the first live-action movie to feature body-motion capture.

Advances in computer graphics, software, processing power, and algorithms made possible the real-time conversion of raw motion-capture data into either an animated figure or a command, noted Oxford's Morris.

How it works

There are several ways to capture body motion. The *passive optical* approach is the dominant technology, according to Morris. This technique places Ping-Pong balls or other types of markers coated with a reflective material on one or more moving subjects' bodies at joints and other points that are key to identifying motion.

Multiple video cameras with strobes around the lenses provide the light that falls on the markers and then reflects back into the lenses. The cameras capture movements at from 240 to 2,000 frames per second. They then send the information to a computer running software that records various aspects of the motion, including position, angle, velocity, and acceleration. The motion-capture system can import this information as a 3D model for use with animation software.

Other approaches capture body motion by, for example, using LEDs as markers to provide the necessary light to cameras. This represents an *active optical* technique.

Systems also utilize magnetic, mechanical, and acoustic technologies.

Acoustic systems track movement by receiving signals from transmitters strapped to a performer's body. Mechanical systems measure the physical motion of a subject's body. And electromagnetic motion capture uses a central transmitter and sensors to relay the position and orientation of an object's parts as they move, based on electromagnetic field changes.

Motion-capture systems then use software and, occasionally, human intervention to conduct the calibration necessary to make an animated character's movements match the model's captured motions.

Calibration is needed to make motions captured from, for example, a performer's elbow match those of an animated character's elbow. Moreover, multiple cameras capture motion from each point on a performer to create a 3D image. Calibration is necessary to combine the information from different angles into the one image.

A NEW DAY FOR MOTION TECHNOLOGY

Processing power is much less expensive than in the past. Also, cameras, other equipment, and animation software are less costly because of technology advances and increasing demand from movie studios and other users.

Because of improved performance and lower cost, motion-capture technology is finding its way into applications other than movies and videogames.

Sports experts use motion capture to, for example, track golf swings or swimming strokes to improve participants' performance. The technology is also used in motion-analysis research and video surveillance.

Proponents tout technology that captures user movements, identifies the motions, and then utilizes them as a potential navigation tool for computerized visual presentations or for issuing commands to TVs and other consumer electronics.

The US Defense Department has requested research into developing such a technology, said Oxford's Morris.

Gesture Studios, a motion-capture technology provider, is working on the GoodPoint gesture-based device-control system.

Companies are using motion-capture technology in more than just movies and videogames.

Advertising

Advertisers are beginning to use motion technology in innovative ways.

MonsterMedia has utilized the technology in campaigns for clients like Adidas, Lexus, and Target in outdoor advertising in high-traffic areas such as Las Vegas' McCarran International Airport and New York's Herald Square subway station, noted Monster Media's Payne. This is important, he said, because outdoor formats such as electronic billboards and display walls are becoming more significant forms of advertising.

MonsterMedia's MonsterVision ground-, wall-, and screen-based systems help keep consumers focused on the advertising by enabling them to interact with the content being displayed. The systems track passersby's body positions to let them manipulate the items being shown by, for example, brushing away virtual snowflakes to reveal a scene behind them, shooting virtual basketballs, or kicking objects projected onto the floor.

The system operates via a PC or laptop linked wirelessly to MonsterMedia's control center, which monitors the content of screens around the country, gauges performance-related issues such as CPU usage and the strength of projector light bulbs, and updates advertising content.

The system also includes a projector, infrared diodes, and a video camera. Pedestrians move between the projector and display surface, onto which the system emits both an infrared light grid and images.

When the pedestrian moves, the camera detects motion via changes in the infrared grid. The PC- or laptop-based software converts the camera input into structured data for analysis. The software then calculates the user interaction with the virtual objects and displays the appropriate result.

Medicine

Doctors frequently use motion-capture technology with patients to conduct gait analyses that identify walking abnormalities that might indicate the presence of a disease. For example, a child's limp could reflect a disorder such as cerebral palsy or muscular dystrophy.

Related technology is also helping with medical education, in elaborate computer-based surgical-simulation systems. These systems frequently use haptics, the science of applying tactile sensations and controls to interactions with computer applications.

The applications re-create medical situations in 3D virtual-hospital settings, complete with avatars and virtual tools, explained Parvati Dev, director of the Stanford University Medical Media & Information Technologies (SUMMIT) Lab.

Dev and other SUMMIT researchers are improving their Spring real-time, soft-tissue simulation platform for building and running surgical simulators. As Figure 1 shows, Spring provides the virtual world in which the simulations occur.

The system uses haptics technology that creates physical resistance, which lets users working with virtual scalpels or other surgical tools feel like they are cutting or manipulating bone or tissue.

Computer Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

Spring captures user motions mechanically via its haptic tool or another external input device such as a mouse. The input device sends information to a haptic system that processes and sends 3D movement data to the virtual world for viewing, usually via IP connections. The haptic system also operates motors that control the input device's feel.

Enhancing engineering and manufacturing

Some large companies—such as BMW, Caterpillar, John Deere, and Lockheed Martin—that make complex items like jets or cars are using motion-tracking systems to design virtual products without having to build large numbers of expensive prototypes. The technology lets engineers determine how well people would work with proposed product designs.

Ford Motor Co. uses motion capture in both product development and vehicle design, noted Elizabeth Baron, one of the company's virtual-reality and advanced-visualization technical specialists. In addition, Ford utilizes the technology for manufacturing simulations to, for example, cost-effectively design safe and efficient assembly lines, said Allison Stephens, the company's technical leader for assembly ergonomics.

Manufacturers also use the technique to analyze existing assembly lines for inefficiencies.

Face tracking

Used primarily to make animated faces move realistically, face tracking works differently than body-capture technology.

Systems capture body movements via strategically placed markers attached to a subject. This works because the human body moves primarily via its large skeletal structure. Markers don't work as well for tracking faces, where subtle expressions are created primarily via small muscle movements, Oxford's Morris explained.

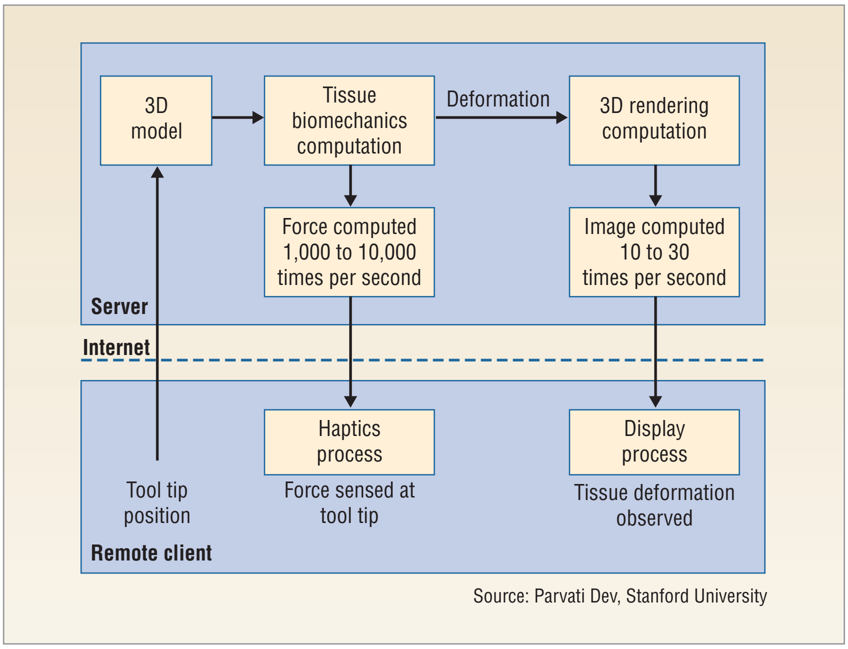


Figure 1. Stanford University's Spring real-time, soft-tissue simulation platform for building and running surgical simulators uses motion capture and haptics. Spring captures user motions mechanically via a haptic or other input tool. The tool sends information to the haptic system, which processes and sends 3D movement data to the virtual world for viewing.

Vicon and vendor Mova are working together on markerless motion tracking. With the Mova Contour System, models apply phosphorescent makeup, which is subject to fluorescent strobe lighting, to their entire face. This enables the capture of a face's motions, yielding a natural surface texture. It also provides a high contrast between the facial surface and the background, Morris noted.

OBSTACLES IN MOTION

Despite improvements, processing speed and price still pose problems for some larger and more specialized motion-capture applications, particularly those that include computationally intensive technologies such as haptics. Smaller, less costly systems will help the market grow because a greater number of small design houses could afford to use them.

Output accuracy is also a challenge, according to motion-capture expert David J. Sturman, who has conducted motion-capture research and worked as chief technology officer for a game developer that used the approach. He is now vice president of technology at Massive Inc., a Microsoft subsidiary that runs a videogame-advertising network.

It is difficult to precisely map and match large numbers of captured motion points to a virtual representation, Sturman explained.

Working with captured data can also be a problem, said Kathleen Maher, analyst and vice president of Jon Peddie Research, a market-analysis firm. The process of modeling and animating characters based on captured motions is complex, although the software for doing this is becoming more effective and less expensive, she explained.

Moreover, because motion-capture systems can be complicated, implementing, configuring, or using them can be a chore. This makes working with them difficult for some smaller organizations with personnel who might not have sufficient expertise or experience with the technology.

In the future, said Stanford's Dev, many more types of medical procedures should be available via motion-capture-based simulations. Currently, she noted, most procedures have not been virtually animated this way.

The long-term goal is to develop systems that can include a specific patient's information, rather than just generic data, to help surgeons rehearse particularly complex operations, she explained.

The long-term goal for motion-capture technology in general is to develop approaches in which users can capture motion by simply pointing a camera at a subject, without using markers, said Sturman. "A lot of the pieces are already in place," he said.

"I think most of the demand will remain in the [advertising, manufacturing], and entertainment industries, with some growth coming from other fields such as sports and medicine," said Maher. "There are no explosions on the horizon, though. We are now following a period of consolidation, and companies are exploring new areas for growth." ■

***Brett Allan King** is a freelance technology writer based in Madrid. Contact him at prensa@europe.com. **Linda Dailey Paulson** is a freelance writer based in Ventura, California. Contact her at ldpaulson@yahoo.com.*

Editor: Lee Garber, *Computer*, l.garber@computer.org

Windows

Windows Kernel Source and Curriculum Materials for Academic Teaching and Research.

The Windows® Academic Program from Microsoft® provides the materials you need to integrate Windows kernel technology into the teaching and research of operating systems.

The program includes:

- **Windows Research Kernel (WRK):** Sources to build and experiment with a fully-functional version of the Windows kernel for x86 and x64 platforms, as well as the original design documents for Windows NT.
- **Curriculum Resource Kit (CRK):** PowerPoint® slides presenting the details of the design and implementation of the Windows kernel, following the ACM/IEEE-CS OS Body of Knowledge, and including labs, exercises, quiz questions, and links to the relevant sources.
- **ProjectOZ:** An OS project environment based on the SPACE kernel-less OS project at UC Santa Barbara, allowing students to develop OS kernel projects in user-mode.

These materials are available at no cost, but only for non-commercial use by universities.

For more information, visit www.microsoft.com/WindowsAcademic or e-mail compsci@microsoft.com.

BECOMING A CERTIFIED SOFTWARE ENGINEER

The IEEE Computer Society (the world's largest professional computer organization) has launched a exam-based process for certifying software engineers as software engineering professionals.

This certificate establishes that the certificate holder is capable of using software engineering methods, tools, and techniques to develop and build software systems and, in addition, can fulfill the roles of:

- Software architect
- Software project manager
- Software configuration manager
- Software quality-assurance expert
- Software test lead, and so forth

IEEE Computer Society csdp Certified Software Development Professional
Registered Preparation Materials

Dr. Richard Hall Thayer has developed a *self-teaching, multi-media, CD training course to aid the CSDP certificate candidate in passing the certification exam.*

For more information go to **www.CSDP-Training.com**

For Programmers, Multicore Chips Mean Multiple Challenges

David Geer

Manufacturers have found themselves unable to effectively continue improving microprocessor performance the old-fashioned way—by shrinking transistors and packing more of them onto single-core chips.

This approach generates too much heat and uses too much power, noted Louisiana State University (LSU) professor Thomas Sterling.

In response, vendors are increasing performance by building chips with multiple cores. Principal microprocessor manufacturers Advanced Micro Devices (AMD) and Intel began releasing multicore chips for PCs and laptops two years ago. Last November, Intel released quad-core chips for PCs and servers.

Multicore chips improve performance by handling various parts of an application in parallel. Single-core chips, on the other hand, undertake tasks serially, said University of Southern California (USC) associate professor Mary Hall.

In some cases, programmers have used workarounds to enable PC games designed to run on single-core chips to run on multicore processors. For example, one type of workaround tries to assign each of multiple threads to a separate core on a chip, explained Martin Walker, a veteran multimedia software developer and chief technology officer of game developer Artificial Mind & Movement.

However, Hall said, this doesn't work as well—and won't enable vendors to add new capabilities to their applications as easily—as designing programs from the beginning to run on multicore chips.

So far, though, vendors haven't moved fast enough to keep pace with the introduction of multicore chips, said Chad Marshall, an IT expert, author, and Bank of America's operational risk manager for technology and end-to-end processes.

One reason they haven't done so is that writing and rewriting programs is time-consuming and expensive, according to Dio De Niz, a member of the technical staff at the Software Engineering Institute, a federally funded R&D center based at Carnegie Mellon University. This is a particular issue for large software vendors with many legacy applications written to run on single-core chips.

Nonetheless, many vendors are already undertaking this task, including game developers, who are always interested in performance.

MULTICORE PROGRAMMING CHALLENGES

Sun Microsystems shipped the first multicore processor in 2000, said Marc Tremblay, the company's chief technical officer for microelectronics. Sun designed the two-core MAJC (Microprocessor Architecture for Java Computing) chip to run 3D-graphics processing, videoconferencing, and general-purpose Java computing on PCs, but it never became commercially viable.

Before that, a number of high-performance computers and servers used multiple processors, which required applications designed to run in parallel. These programs have been on the market since the early 1990s, Tremblay noted.

Programming for single-core and multicore chips

For single-core chips, programmers write applications so that they prioritize tasks in the order they must be performed to do the designated activity most efficiently, explained De Niz.

The operating system then assigns the tasks to be run serially on the processor, said LSU's Sterling.

Developers write applications for multicore chips so that they will divide into tasks that the operating system can assign to run in parallel on multiple cores, he noted. And each core has its own multithreading capabilities and thus can divide its own tasks into parts that can run in parallel.

A key issue for programmers is how to divide up an activity into tasks. One common approach separates an activity into its various discrete subfunctions, said De Niz.

Developers also must figure out how to rewrite legacy programs built for single-core chips, a process that can be costly and time-consuming. In this case, developers must first determine which routines and tasks take up most of the compute

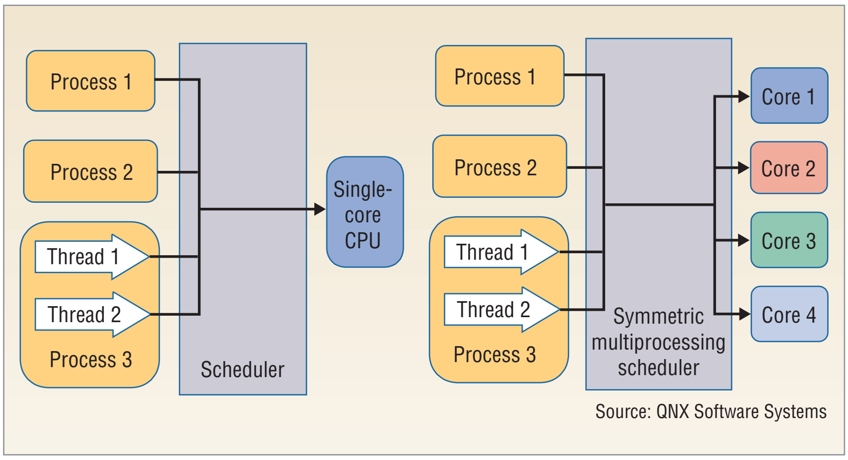


Figure 1. QNX Software Systems has a real-time operating system that offers the scheduling of tasks for single-core and multicore-chip applications. For single-core processors, the scheduler prioritizes tasks to be run serially on the chip. The symmetric multiprocessing scheduler prioritizes tasks too, but it also must determine how best to run them in parallel on a multicore chip's processing units.

time, explained Sterling. Then, he noted, they rewrite those tasks that can be easily parallelized. However, they often must leave other tasks as they were originally written.

Multicore challenges

While the goal of programming for multicore chips is clear, doing so isn't necessarily easy.

Dividing activities into smaller parts. A key concern for programmers is that some activities—such as those related to graphics—don't divide easily into subfunctions with naturally occurring beginning and ending points, De Niz said.

In these cases, the programmer must perform complex transformations that yield points at which an application can be broken into tasks that separate software into parts that are smaller and more granular than hard-to-divide subfunctions, or that change entire data structures.

Data dependency. Programmers must synchronize the chip's handling of an activity's various parts so that each calculation will have the access to the processed data from other parts that it needs to perform its work, an issue known as *data dependency*.

If synchronization requires more processing than is saved by parallelizing the data, parallelism may not be useful.

Developers sometimes write *safety properties* into programs to ensure that applications don't try to run operations until they have the fully processed data they need from other operations and that they pass on only properly handled information, according to De Niz.

Data splitting. Data splitting occurs when, in the course of dividing a program for handling by multiple cores, two or more parts of the same data set are divided and sent to separate cores, according to De Niz. If, in the middle of a task, a computation needs part of a data set that is on another core, the results may turn out to be incorrect.

Balance. One of multicore developers' main goals is to maximize efficiency by splitting up tasks among different cores so that each is doing work of equal value to the overall process.

The programmers' effectiveness depends in part on their assumptions as to the relative value of subfunctions that run on separate cores. If one task designated to run as a separate subfunction turns out to be not all that important to the overall application, it might waste the resources of the core on which it is run.

This issue will become a bigger problem and thus create scalability issues as processors get more cores, according to Ray DePaul, CEO of RapidMind, a vendor of development tools for applications that run on multicore chips.

Testing. Applications that have multiple pieces running on different cores can have many execution paths. This makes testing a sufficient number of the paths difficult.

Programs that run serially, on the other hand, have fewer potential execution paths and thus are easier to test.

ANSWERING THE CALL

Various vendors are addressing the need to develop programs that run optimally on multicore chips.

For example, Microsoft designed Windows Vista to work efficiently with chips that have up to four cores. Apple's upcoming MacOS X v10.5 will also accommodate multicore processors.

QNX Software Systems—a provider of real-time operating system software, development tools, and services for embedded designs—has an RTOS that can effectively schedule the parts of an application that will run on multicore processors, as Figure 1 shows.

Intel has released tools—Intel Compiler 10.0 Professional Edition; Intel Thread Checker; Intel Thread Profiler; and Intel Thread Building Blocks for Windows, Linux, and the MacOS—to help developers write programs for the company's multicore chips.

AMD works with other companies and organizations—including Microsoft, the Portland Group, Sun, and the GNU open source community—to ensure their tools will be able to develop applications that run on AMD multicore chips.

Moreover, the vendor has designed its multicore chips to work with multiple brands of compilers, noted Margaret Lewis, AMD's director of commercial solutions.

The chipmaker has also provided

libraries—including the AMD Core Mathematical Library and AMD Performance Library—with highly optimized routines for developing, debugging, and optimizing applications that run on its multicore processors, she added.

RapidMind Development Platform parallelizes code that developers are working on, said DePaul. The platform comes with a runtime environment, APIs, and software libraries.

The developer can continue to implement in a familiar environment that looks single threaded. The code is then passed on via the APIs to the platform, which maps the work to all of the available cores, explained DePaul.

Some schools are preparing future developers to write applications for multicore processors. For example, Purdue University offers classes on this topic, noted associate professor Suresh Jagannathan.

NEW LANGUAGES FOR MULTICORE APPLICATIONS

Approaches that make multicore programming as simple as single-core, single-threaded programming will enable vendors to quickly get products to market, according to DePaul.

To accomplish this, some proponents say new programming languages may be necessary.

The new languages would let developers continue using the serial-programming approaches they are familiar with. However, the languages would automatically handle some of the manual choices developers currently must make to accommodate multicore programming, according to Jim Sexton, a staff member at IBM's T.J. Watson Research Center.

The Institute for Defense Analysis, a nonprofit R&D center that works for US government organizations, is developing such a language, Universal Parallel C, for the National Security Agency.

Sexton said IBM is working on the X10 language for writing applications that will run on multicore chips in PCs, workstations, laptops, and supercomputers.

Many industry observers say multicore processors' future is bright. According to DePaul, 16-core chips will be available by the end of this decade. Intel has already developed a research chip with 80 cores.

Vendors won't have to rewrite applications that normally handle work serially, such as e-mail, word processing, and most traditional PC-based programs, because they won't benefit from parallelization.

For the same reason, they also won't have to rewrite software that doesn't use many resources, such as calculators, calendars, and other simple interactive utilities, noted USC's Hall.

According to Artificial Mind & Movement's Walker, most game developers have avoided the need to rewrite applications so far by building patches that let existing games run on multicore processors.

However, he explained, simply assigning each of the multiple threads to a separate core on a chip won't necessarily maximize the use of each core's capabilities and will require vendors to regularly rewrite patches to account for the increasing number of cores that chips will have.

Vendors will have an incentive now to rewrite CAD and other programs that are computation-intensive or those that can be easily parallelized such as database applications, for multicore chips.

And numerous developers of games and other types of programs are already writing their applications from scratch to work with multicore chips, a trend expected to continue in the future. ■

David Geer is a freelance technology writer based in Ashtabula, Ohio. Contact him at david@geercom.com.

Editor: Lee Garber, *Computer*, l.garber@computer.org

Silver Bullet
Security Podcast series

Sponsored by cigital and IEEE SECURITY & PRIVACY

Check out the Silver Bullet Security Podcast with host Gary McGraw, author of *Software Security*, *Exploiting Software*, and *Building Secure Software*! This free series features in-depth interviews with security gurus, including

- Avi Rubin of Johns Hopkins
- Marcus Ranum of Tenable Security
- Mike Howard of Microsoft, and
- Bruce Schneier of Counterpane Internet Security

Stream it online or download to your iPod...

www.computer.org/security/podcasts

Computer | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | qMags

Company Says Diagonal Wiring Makes Chips Faster

A leading semiconductor-design software-tool vendor has developed an approach that would increase chip performance by running wiring diagonally—as well as horizontally and vertically, as is the case now—between transistors, on-chip memory, and other processor elements.

Cadence Design Systems says this technique would also reduce manufacturing complexity and improve energy efficiency.

Typically, the copper or aluminum wires in a chip run only along horizontal and vertical axes, an approach known as Manhattan routing. The interconnections deliver power and carry data, explained Chi-Ping Hsu, Cadence's corporate vice president for integrated-circuit digital and power forward.

In today's complex chips, which can contain "tens of miles" of wiring, there can be as many as 15 layers of circuitry, Hsu noted. Each layer generally has wiring running in just one direction. This avoids having wires in a single layer cross one another, which could cause short circuits. Additional interconnects run between the layers.

Cadence's X Architecture approach uses wiring that runs diagonally, along with the connections that run horizontally or vertically, in some layers.

Diagonal wiring enables shorter, more direct connections between processor elements in many cases and thus reduces the manufacturing complexity that Manhattan routing can require over longer distances.

The X Architecture saves energy by having signals travel between points over direct diagonal paths, which are shorter than the more indirect paths that use both the horizontal and vertical wiring in multiple layers, as Manhattan routing can necessitate.

Adding diagonal wiring also gives designers more routing options and thus greater flexibility, Hsu said. The architecture lets chip designers make more design trade-offs to get the chip characteristics needed for specific applications, such as lower dynamic operating power in mobile devices.

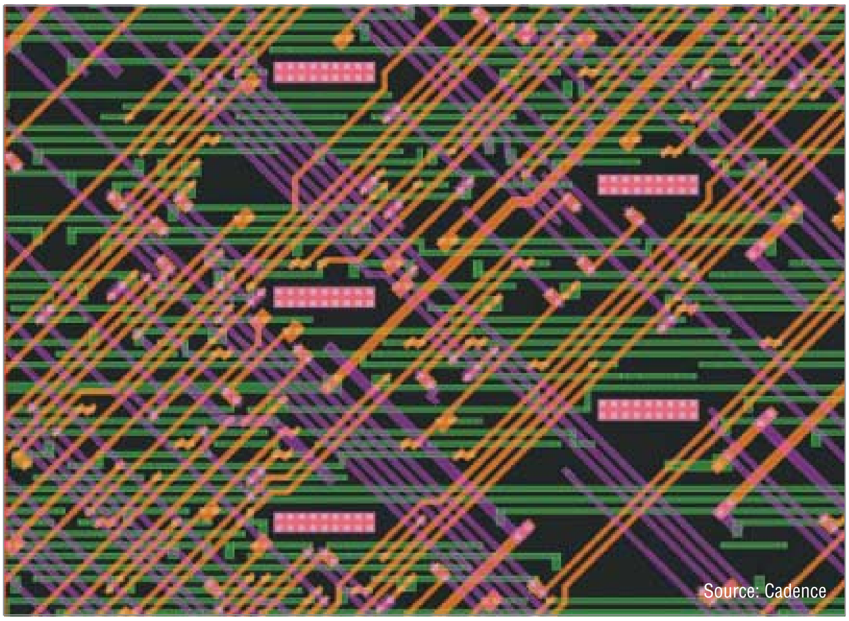
Also, introducing diagonal paths lowers the amount of wiring and concentrates more connections within each layer, which could decrease the number of layers. This would cut production costs. In addition, less wiring would reduce problems such as crosstalk and current leakage into the substrate, which typically decrease chips' performance and energy efficiency.

Agere Systems, which LSI Corp. recently acquired; ATI, which AMD purchased last year; and Teranetics use the X Architecture in some of their chips.

Cadence has had trouble convincing chip makers to use the X Architecture partly because it is largely untried, would force them to buy new equipment, and is supported by only one vendor, said Gabe Moretti, a veteran observer of the electronic-design-automation industry who operates the Gabe on EDA Web site (www.gabeoneda.com).

Also, he added, "I believe that what has been demonstrated so far has yielded only marginal improvements over traditional routing techniques."

According to Hsu, the companies trying the X Architecture now are aggressive, risk-taking, and pushing the state of the art. ■



A photograph of a cell-phone chip shows how Cadence Design Systems' X Architecture runs wiring diagonally —as well as horizontally and vertically—between elements on a processor. Currently, wiring on most chips runs only horizontally and vertically. Cadence says its approach increases performance, reduces manufacturing complexity, improves energy efficiency, and enables more design flexibility.

Published by the IEEE Computer Society

New Technology Transmits Data via Visible Light

Researchers are working on a technology that would utilize the LEDs in common light fixtures to transmit data over short distances.

"When LED light fixtures become commonplace, their use as a communications device will also become commonplace," predicted Shinichiro Haruyama, a Keio University professor, vice president of the Visible Light Communications Consortium (VLCC), and one of the original VLC researchers.

"LEDs are becoming popular," he explained, "because their light-emitting power is becoming large enough and their lifetime is longer than those of conventional incandescent lamps and fluorescent lights." Also, he noted, LEDs' power efficiency is greater than that of incandescent lamps and about same as that of fluorescent lights.

Tracking Troubled Turtles with Wireless Technology

US scientists are attaching small computers to snapping turtles and using their own TurtleNet network technology to track the reptiles' movements, as part of a project that could enable researchers to help the animals cope with threats to their habitat.

University of Massachusetts Amherst researchers designed this project to help them understand the turtles' regular movements, which could let them help the reptiles survive encroaching land development and an increase in predators, which threaten some other types of turtles.

For example, the scientists could inform developers where turtles frequently go so that they could avoid building there, explained Mark Corner, a University of Massachusetts assistant professor of computer science.

The project would also provide basic biological information about the turtles, he noted.

In general, turtles are subject to swift population declines because they live a long time and reproduce infrequently, noted Michael Jones, a University of Massachusetts PhD candidate in organismic and evolutionary biology.


Source: University of Massachusetts

A snapping turtle wears a solar-powered computer that University of Massachusetts researchers are using to track its movements. This is part of a project that could enable the scientists to help the animals cope with threats to their habitat.

Therefore, he explained, the researchers decided it would be prudent to research the snapping turtles as part of their larger Diverse Outdoor Mobile Environment (DOME) project.

The scientists use orthodontic cement and duct tape to attach small computers to turtles' backs and track their movements along western Massachusetts' Deerfield River.

The machines record information about the animals' location and the ambient temperature, which identifies when they are sunning themselves and when they are in the water. The computers are powered by the sun, which also recharges the batteries that provide power during the night or in overcast conditions, said Corner.

"The computers are about the size of a roll of coins but weigh less than 100 grams. We are always trying to make them smaller so that they won't interfere with the turtles' movements," he said.

When the turtles pass within 500 meters of one another, their radios exchange information via a disruption-tolerant network, which uses a data-transmission technology that the researchers developed for their DTN. This puts as much data as possible on each animal's computer.

Each turtle also transmits data when it is within about 500 meters of a centrally located base station. The station then uses cellular technology to send the data to the researchers about 24 kilometers away.

The turtle-mounted computers thus don't have to make long transmissions, which would require large, heavy batteries that need frequent recharging.

In the future, Jones said, the researchers hope to refine the technology for use with species that have critical conservation needs.

VLC—recently implemented in Japan—would be most useful in locations where cellular and other radio-based systems don't work optimally, such as dense urban areas.

Researchers created VLC modulators that turn the power to an LED on and off—too rapidly for the human eye to process—thereby creating binary data's ones and zeros.

A VLC system would include LED lights; photodetectors, which sense light and convert it to electricity; and a small transceiver chip that relays data between systems.

The technology could work with light sources other than LEDs. The researchers say they chose to experiment with LEDs in the belief that they will probably be used more than other types of lights in the future.

VLC's top transmission speed with common LEDs will probably be 20 or 30 megabits per second, although it could top out as high as 80 Mbps, according to Haruyama. Although the technology may be slower than some other types of short-range communications, the main benefit will be its pervasiveness, he explained.

A type of LED designed strictly for communications could offer data rates up to 500 Mbps in the future, although it would not provide a very bright light.

The technology's transmission range in tests conducted with an LED traffic light was 30 meters. However, Haruyama said, improving the receivers eventually could increase the range to a kilometer.

VLC could link to other types of wired or wireless communications networks.

In addition to communications, VLC could be used for location detection. The light in a building could send a unique location identifier, the address and room number, and the structure's longitude and latitude to a visible light receiver.

Haruyama said it remains to be seen whether adding communications features to LEDs would make them significantly more expensive and, if so, whether manufacturers would do so.

Light-based communications are not affected by electromagnetic interference. However, the technology is limited to locations at which transmitters could send light to receivers either directly or by bouncing it off another object.

Already, said Haruyama, a Hong Kong University professor is using an LED traffic light to send sound, and some Daimler Chrysler researchers are using LED head lamps and tail lamps for sending data between cars.

The Tokyo-based VLCC (www.vlcc.net)—which includes companies such as Mitsubishi Heavy Industries, NEC, NTT DoCoMo, Sony, and Toshiba—coordinates VLC research and standardization. The consortium is looking at potential business models for commercializing the technology. ■

Researchers Herd Computers to Fight Spyware

A joint research project of Harvard Law School's Berkman Center for Internet & Society and Oxford University's Oxford Internet Institute is using a technique called *herd computing* to fight spyware and various types of malware.

Herd computing uses multiple computers from voluntary participants. Software runs on each client, gathers data about the computer's vital signs—such as CPU performance, memory consumption, pop-up windows, system crashes, and the activities of processes running on the computer—and sends it to a server.

The server then collects and analyzes the data, presents it in a form viewers can see, and anonymizes the material to eliminate IP addresses and other information that identifies individual machines.

Participants and network administrators monitoring the herd can view the processed information via either a browser or a light client with browser functionality.

Data transfer between elements can occur via any of a number of wired or wireless Internet-based communications technologies.

In essence, herd computing lets participants who are considering running new code see how many other machines have chosen to run the software, as well as the effects it has had on the computers, explained Oxford professor Jonathan Zittrain, who cofounded the Berkman Center.

Using participant experience or the observed behavior of applications, members could decide whether to download the software, based on their organization's computing environment and usage policies, Zittrain said.

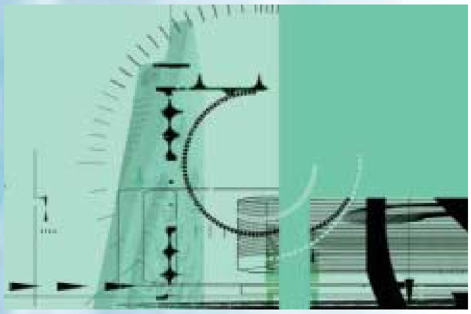
Herd computing lets participants configure their computers to decide the extent to which they want to participate and thus how much processor and bandwidth they will contribute, he noted.

The researchers plan to include open APIs in future pilot herd networks so that users can write widgets enabling them to work with collected data as they choose. ■

News Briefs written by ***Linda Dailey Paulson****, a freelance technology writer based in Ventura, California. Contact her at ldpaulson@yahoo.com.*

Editor: Lee Garber, *Computer*, l.garber@computer.org

A New Era of Performance Evaluation



Sean M. Pieper
University of Wisconsin-Madison

JoAnn M. Paul
Virginia Tech

Michael J. Schulte
University of Wisconsin-Madison

Long-standing techniques for performance evaluation of computer designs are beginning to fail. Computers increasingly interact with other computers, humans, and the outside world, leading to scenario-oriented computing, an emerging category of design that will enable future consumer devices and usher in a new era of performance evaluation.

Since 1943, researchers have used latency and throughput as the primary metrics to describe computer performance. These metrics served us well because we used computers in fairly simple ways.

The unspoken assumption is that data is available on demand and only its quantity and content can affect execution time. This implies batch-style execution, in which measuring each program's speed, including the operating system, in isolation can determine overall performance. For performance-evaluation purposes, programs are merely an extension of instructions—reduced latency and higher throughput are always better. This perspective informs the design of benchmark suites such as those from the Standard Performance Evaluation Corporation (SPEC)[1] and the Embedded Microprocessor Benchmark Consortium (EEMBC),[2] which are composed of batch-style jobs executed in isolation.

For many new computer systems, such evaluation is misleading. Computers increasingly interact with humans, the physical world, and each other—often simultaneously. Overall performance in this context is a function not only of individual applications, but also of their interactions as they contend for resources both internal and external to the device. Cell phones, for example, often perform some baseband processing in software. Wireless communications arrive over time rather than on demand, and strict requirements dictate when outputs must occur. Other tasks such as videoconferencing depend on this software, but they also can compete with it for memory and processing resources. I/O subsystems over which the processor has little or no control and interdependencies between unrelated programs break the batch processing model, but they are essential aspects of this new computing style.

Researchers must describe modern computer usage in terms of scenarios consisting of numerous I/O streams, timing information, and parallel tasks that enter and leave the system, rather than in terms of programs executing in isolation from the physical world and each other.[3] Such use represents a new style of computing, which we call *scenario-oriented* to contrast it with other well-established computing styles such as general-purpose and application-specific. Table 1 compares these three styles.

Evaluation methods designed for general-purpose and application-specific computing are insufficient for scenario-oriented computing. Existing benchmarks do not reflect modern usage, and their metrics fail to describe performance as perceived by end users.

COMPUTER USAGE EVOLUTION

As Figure 1a illustrates, with traditional computer usage, a single task, T1, enters the system, executes for some period of time,

Table 1. Comparison of general-purpose, application-specific, and scenario-oriented computing.

Computing style	User programmability	Design	Performance evaluation	Inputs
General-purpose	Complete programmability	Balanced performance	Each application evaluated individually	Sequenced by application
Application-specific	Limited or no programmability	Excellent performance for a single application	Compared against known requirements	Timed to external reference
Scenario-oriented	Can install software for new functionality	Variety of uses, but emphasizes performance of some	Holistic evaluation of scenario components and their interactions	Both sequenced by applications and timed to external reference

and then completes. Some time later, another task, T2, enters the system and takes its turn. This model abstracts schedulers and other operating system features to simplify performance evaluation. There is no contention between T1 and T2. Only program control flow sequences and interleaves data access; there is a single input stream and a single output stream.

In contrast, as Figure 1b depicts, modern usage is more complicated. Many tasks operate simultaneously, contend for resources, and communicate with each other. Unlike traditional usage, both asynchronous and streaming I/Os such as alerts and user inputs, webcams, and music are important to the functionality, and there is not necessarily a one-to-one mapping from inputs to outputs.

An arbitration layer uses preemptive scheduling to allow for interleaved execution and to enable multiple logical I/O streams to share a single physical link. This layer supports real-time requirements and user demand for concurrency. Advanced hardware also enables simultaneous execution. Complex interactions between tasks and hardware resources prevent describing an entire system's performance in terms of a single task evaluated in isolation or even of multiple tasks run one after the other. This break from traditional assumptions on computing causes Amdahl's law to fail for bounded systems with heterogeneous resources in that slowing down some tasks can actually improve overall performance.[4]

While Figure 1a accurately describes scientific and engineering usage, the computing industry has expanded beyond the small market of engineers and scientists who use computers to develop and run batch-style programs to an ever-growing group of nontechnical users. Software that uses processing power to deliver both new functionality and increasing ease of use has made this growth possible. These nontechnical users currently buy billions of processors every year in the form of cell phones, set-top boxes, and music players; and they expect these devices to make their lives easier and more enjoyable.

Examples such as e-mail, Web browsing, and gaming illustrate how researchers have historically harnessed increased processing power to create a virtual infrastructure unique to computing. These applications do more than simply facilitate problem solving; they actually create entirely new technological foundations that increase demand for computing. In turn, the applications themselves become ever more sophisticated. E-mail, for example, dates back to at least 1972, but increased memory and processing capability, as well as multithreaded operating systems, have expanded its capability far beyond the transmission of text messages.

Modern e-mail programs have integrated this base functionality with features such spell checking, junk-mail filtering, scheduling, sort and search functions, HTML support, image decompression, encryption, and virus checking. As a result, e-mail is significantly more useful, but it is also more complicated and computationally intensive.

This trend is not confined to general-purpose devices and their applications. Cell phones have reached the point where they no longer can be considered traditional application-specific devices. They now use multithreaded operating systems, such as Symbian and Windows CE, and they can run applications such as video-editing software and 3D games that would traditionally run on a PC. Users are thinking less in terms of general-purpose computers or single-purpose systems, such as phones, and more in terms of programmable convergence devices that integrate into various aspects of their lives. They expect such devices to facilitate common tasks and to enable novel ways of interacting with the world.

TURNING POINT IN HARDWARE DESIGN

Modern users' demands are rapidly outpacing the capabilities of existing design methodologies. The computer community has been in this position before. Vacuum tubes gave way to discrete transistors, then came simple integrated circuits, followed by very large-scale integration systems. In each case, the response was to create entirely new foundational principles. As uniprocessor design hits its limits, researchers must find new design methodologies to deliver next-generation functionality.

Sematech's most recent *International Technology Roadmap for Semiconductors* suggests that single-core designs cannot scale beyond 20 to 25 million transistors. Multiprocessor designs with "SOC-like integration of less efficient, general-purpose processor cores with more

Computer Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

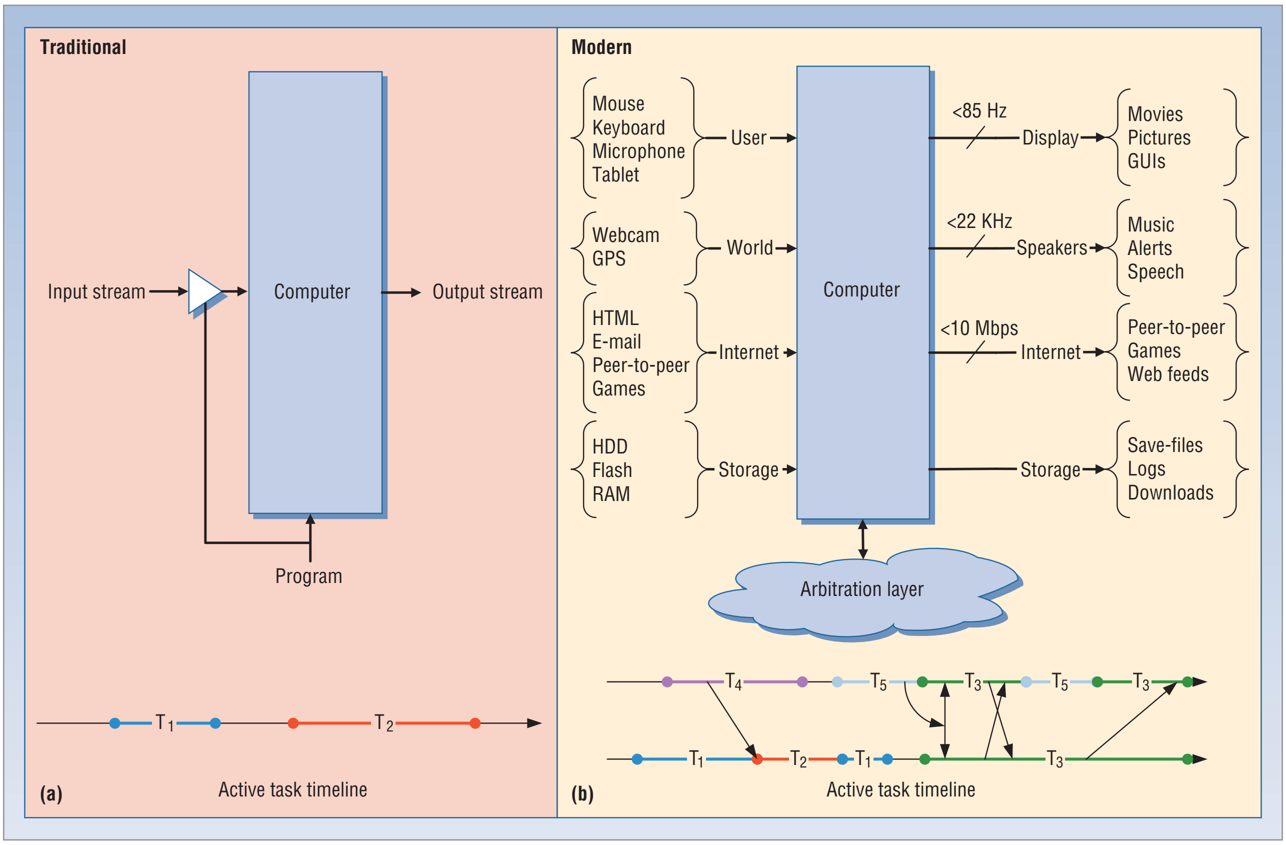


Figure 1. Traditional and modern computer usage. (a) Common traditional tasks include compilation, data compression, physics simulation, placement and routing, and discrete event simulation. (b) Common modern tasks include image manipulation, video and audio playback, e-mail, virus scanning, Web browsing with dynamic content, voice over IP, and 3D gaming.

efficient special-purpose 'helper engines'" are projected to be the next step in computer evolution.[5] Developers expect as many as 63 processing engines—cores and custom logic blocks—on a single chip by 2009. The migration to single-chip heterogeneous multiprocessors (SCHMs) will pick up over the next few years and ultimately allow exponential increases in performance to continue while reducing reliance on clock scaling.[5]

Two early SCHM architectures for commercial devices are the Sony, Toshiba, and IBM (STI) Cell and the Sandbridge Sandblaster.[6,7] Cell is geared toward set-top boxes and game consoles, while Sandblaster targets wireless handsets. In both of these areas, the most difficult problems, such as physics processing for games and baseband processing in cell phones, contain significant data- and thread-level parallelism. To exploit this parallelism, both Cell and Sandblaster combine a cluster of single-instruction, multiple-data processors with a single scalar processor. Cell uses its scalar processor to coordinate the SIMD units, and Sandblaster uses its to handle user-interface tasks. Neither architecture reserves processors for specific functions. As a result of their nontraditional design and programming model, processors such as Cell and Sandblaster have been described as system-on-chip designs, but this is not strictly accurate. Both Cell and Sandblaster are more accurately described as "processors of processors."

SoC descends from application-specific integrated circuit design and provides a methodology to rapidly develop integrated circuits for complex, but well-defined, task sets that are fixed at design time. The SoC design style divides the chip into several units; some of these units can be programmable, but each has a fixed purpose. Cell and Sandblaster diverge from this model by considering the entire chip as a programmable device that must be able to dynamically reallocate resources. They also are intended for devices that are marketable based on their compelling features, rather than sheer processing power.

SCENARIO-ORIENTED COMPUTING

The changes in usage combined with developments in technology point to a new organizing principle for design—rather than being general-purpose or application-specific, computing is becoming scenario-oriented. Consider an onboard navigation system that determines its current location using GPS, and receives verbal instructions, such as "Go to 1600 Pennsylvania Avenue."

In response to the user's command, the system connects to a map server and checks for traffic advisories,

Computer Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

calculates and displays an optimized route, and transmits the directions through speech synthesis software as the user nears the destination. If a traffic advisory arrives, the computer drops the speech synthesis and seeks an alternate route. Unlike application-specific computing, the processor performs different tasks over time, but unlike general-purpose computing, these tasks share a common goal.

In contrast with the assumptions of both general-purpose and application-specific design, actual usage of devices such as cell phones, PDAs, and set-top boxes is modal. Users view these devices differently according to their immediate purpose—they might use a smart phone as a scheduler, music player, game-playing device, digital camera, or simply as a phone. These devices also can implement a single mode in several ways—for example, a user might play songs in several different formats while using the device as a music player. The user's expectations distinguish the modes, rather than the actual hardware or software that enables them.

The processing power of heterogeneous programmable cores is intended for a wide range of tasks.

Because customers expect a variety of uses for a finite amount of silicon, heterogeneous programmable cores become the central elements in scenario-oriented hardware. In contrast to the fixed-purpose resources in SoC and other application-specific design styles, the processing power of these cores is intended for a wide range of tasks. Unlike general-purpose computers, scenario-oriented devices must accommodate varying demands for different types of processing within a finite amount of silicon and certain time constraints. Heterogeneity is a response to this challenge. Software designers can leverage modality to inform scheduling decisions and use heterogeneous cores more effectively.

FAILURE OF EXISTING METRICS AND BENCHMARKS

A benchmark suite is a set of applications that provide a representative sample of usage. SPEC CPU,[1] the primary benchmark suite computer architects use, contains a variety of real engineering and scientific applications that are selected and modified to run with minimal operating system support and interaction. The EEMBC benchmarks, which contain representative kernels and applications from the embedded domain, support embedded systems design.[2] The Stanford SPLASH benchmark suite, which measures the runtime of parallelizable algorithms, provides similar services for traditional multiprocessor architectures.[8] All applications in these benchmark suites are batch jobs and are executed in isolation. Figure 1a illustrates this type of usage.

Performance in SPEC is measured as speedup, $s = \tau_{reference}/\tau_{measured}$, over a reference system. Because the design goal is to provide excellent performance under arbitrary usage, each application's speedup is treated equally, using the geometric mean to generate a composite score. The geometric mean places greater weight on entries with low performance than those with high performance—if a single result is 0, the entire output is 0, giving this entry infinite weight. This rewards balanced performance, which is appropriate for general-purpose usage, but does not accurately describe scenario-oriented performance.

SPEC also includes SPEC_rate, a throughput measurement intended for multiprocessor systems. To generate SPEC_rate scores, the computer executes *n* copies of each task simultaneously and then measures the time to complete all *n* copies. This measurement anticipates homogeneous usage appropriate to industrial applications such as simulation, Web hosting, database processing, and supercomputing. Scenario-oriented design, in contrast, anticipates diverse usage, as is common in most consumer applications.

Some newer benchmarks such as Business Applications Performance's SYSmark and Futuremark's 3Dmark[9] are more representative of commercial use. SYSmark evaluates computer performance in a business setting. It uses common commercial applications such as Adobe Acrobat Reader, Macromedia Dreamweaver, McAfee VirusScan, and Microsoft Office in combination with input events and data generated by observing real users. Multiple applications execute together under different scenarios, such as communication (e-mail and Web browsing) and data analysis (database queries and spreadsheet operations). The benchmark reports separate scores for each scenario. SYSmark focuses on response times rather than runtimes, reflecting the fact that many applications are event-driven and can go idle while the user is not interacting with them.[10]

SYSmark comes closer than SPEC to describing modern usage, but it still does not include any real-time applications. Real-time tasks such as streaming media, baseband processing, and voice recognition are essential to multimedia, mobile usage, and human-computer interaction. Their absence limits SYSmark's ability to describe scenario-oriented usage. SYSmark's focus on current usage also limits its applicability to scenario-oriented hardware design. Several years can pass between the time developers make fundamental design decisions and when a new device hits the market. Designers need the ability to evaluate performance under anticipated future workloads.

3Dmark evaluates gaming performance under next-generation loads. It measures the real-time frame rate of a set of games with extremely demanding graphics.

3Dmark originally focused on graphics processing units, but recently it has added a CPU portion to model the impact of AI and physics calculations on frame rate. While 3Dmark can describe real-time performance of future gaming workloads, its dependence on frame rate as a figure of merit limits its applicability to other areas.

While researchers have invested much effort and creativity in the design of these benchmarks and their associated metrics, they are insufficient for guiding scenario-oriented design for the following reasons:

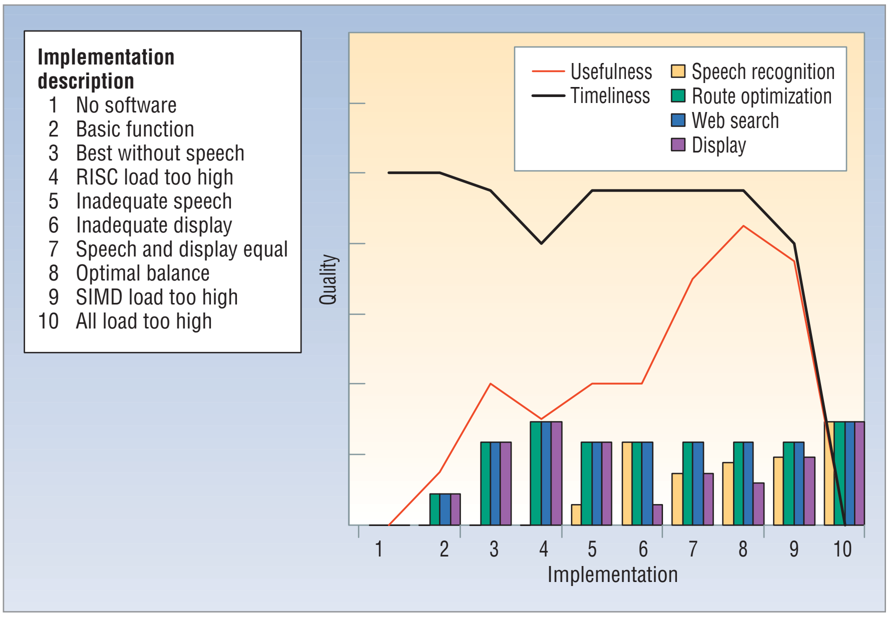


Figure 2. Timeliness and usefulness of various implementations of a navigation scenario.

- *Their composite metrics weight all applications equally*. This is a relic of sharing general-purpose processors for batch jobs. With interactive usage, fast responses to some events are more important than the response time to others.
- *They judge hardware on its ability to accelerate, rather than enable*. Customers expect increasing integration of new features such as speech recognition, rather than faster execution of existing features such as spell checkers.
- *They can't describe cooperation across tasks*. If several tasks operate toward a common purpose, the acceleration of some tasks is not necessarily beneficial and can even degrade overall performance.

Fundamentally, none of these approaches identifies the modifications that would improve a scenario-oriented design. Computer architecture has historically been the art of identifying performance bottlenecks and then identifying performance facilitators such as caches and branch predictors to alleviate these bottlenecks. Different performance facilitators will exist for future architectures, but these will facilitate critical cases—the instances when application software overloads the system and performance rapidly deteriorates. Scenario-oriented benchmarks must enable designers to identify critical cases and, in doing so, aid the discovery of new performance facilitators.

EVALUATING SCENARIO-ORIENTED PERFORMANCE

Figures 2 and 3 describe a hypothetical system's performance in an onboard navigation scenario in terms of usefulness and timeliness. Usefulness indicates the degree to which the device helps the user navigate, and timeliness indicates the device's ability to perform calculations in a timely manner. Although subjective, human or even marketing studies can measure usefulness. Timeliness is a complex metric—some deadlines can be more important than others (this relative importance, of course, is also subjective) and creating a composite can be difficult. The importance of such metrics lies in bringing performance evaluation in line with user satisfaction.

The navigation system is assumed to involve speech recognition, route optimization, Web search, and graphical display, executing on a SCHM with both reduced-instruction-set computer and SIMD cores. The speech and display software run better on the SIMD cores, while the route optimization and Web search run better on the RISC cores.

Figure 2 compares the performance of several possible implementations of a navigation scenario as the component algorithms' complexities vary. Each group of bars represents an implementation, and each bar—whose height indicates relative computational complexity—represents a task. Reasons for complexity changes can include algorithm selection, the amount of data the system is processing, or control dependencies on input values. We assume that, given infinite processing power, higher complexity results in increased usefulness.

Figure 2 illustrates three important points:

- Software and hardware are not evaluated independently.
- Adding a new feature can significantly increase a device's usefulness even if the individual quality of other features is sacrificed.
- The relative amount of computing power dedicated to each feature has a significant effect on usefulness and timeliness.

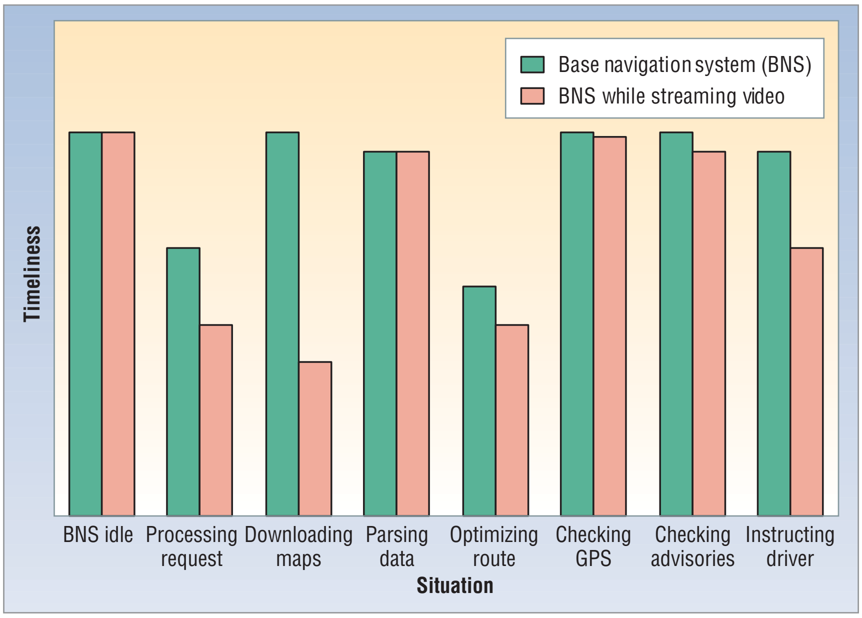


Figure 3. Navigation system timeliness. The green bar shows the timeliness of the navigation system executing alone, while the yellow bar describes timeliness when the system is downloading and displaying a streaming video at the same time.

In implementation 1 in Figure 2, no software is running. As a result, no deadlines are missed, and the timeliness rating is perfect. However, the lack of functionality brings the usefulness score to zero.

Implementation 8 does not have perfect timeliness, but it incorporates enough functionality in a sufficiently timely manner that the device is very useful. In implementation 10, the load is too high, and all deadlines are missed. Timeliness bottoms out, and this degradation destroys usefulness. On a more powerful processor, however, timeliness would improve, and the usefulness of these more complex implementations would increase with it. Software must match hardware to optimize performance.

Comparing implementations 1-4, which do not include speech recognition, with implementations 5-8 shows the impact of adding an additional feature. Usefulness hits a local maximum in implementation 3, and then begins to decrease because the requirements for route optimization and Web search are too high, and they must be performed on SIMD processors where they execute less efficiently. Using these processors to implement speech recognition, rather than improve existing features, will make the device far more useful overall.

Implementations 5-8 also demonstrate the importance of striking the correct balance. When the speech recognition is prioritized too much in implementation 6, the display must be sacrificed to maintain timeliness. In implementations 7 and 8, the balance adjusts to improve usefulness without reducing timeliness. This leads to an unequal division of computing resources.

Figure 3 illustrates a navigation system's dynamic behavior and the impact of adding an extra feature to this system. The figure shows different situations in roughly chronological order from left to right. The navigation system is idle until it receives a verbal request, which triggers a chain of computation that continues until the system finds a route. From then on, the navigation system periodically polls the GPS and traffic advisory Web sites and announces each turn. Two bars illustrate each situation. The green bar shows the timeliness of the navigation system executing alone, while the yellow bar shows the timeliness when the system is downloading and displaying a streaming video at the same time.

This type of graph can reveal unexpected consequences of adding new functionality that might not be apparent from the macroscopic view of Figure 2. For example, downloading map data is not a problem for the base navigation system, so we might not think about how adding a new feature affects performance. Figure 3, however, reveals that this is the worst case when simultaneously streaming video. Further analysis might reveal that limited bandwidth is to blame, which could lead to solutions such as compressing map data or designing around a more sophisticated wireless protocol.

Figures 2 and 3 demonstrate the potential for new performance representations to isolate critical cases and describe tradeoffs. When performance evaluation breaks free of traditional assumptions, many representations become possible.

TAKING PERFORMANCE EVALUATION INTO A NEW ERA

Because it differs so vastly from both general-purpose and application-specific computing, scenario-oriented computing requires an overhaul of performance evaluation. We can divide this challenge into benchmark selection and metric design.

Scenario-oriented benchmark criteria

Benchmark selection is the problem of defining a computational load in terms of inputs (programs and data) and timing information. New scenario-oriented benchmarks should satisfy the following criteria:

- *Include software and hardware interactions.* Tasks can wait for each other, communicate, spawn children, and leave the system. As a result, the processor's load can change dramatically during usage. Figure 3 illustrates this type of dynamic behavior as well as an example in which multiple tasks compete for a shared hardware resource.

- *Provide timing information for inputs and outputs.* Because a mouse click can spark a chain of calculations, its occurrence relative to other events is important. For example, Figure 3 shows how processing directions while streaming a video can hurt performance, but running the streaming video is fine when the base navigation system is idle. Outputs such as video and music have associated timing requirements that play an important role in determining perceived quality.
- *Exercise critical cases.* Scenarios should isolate the knees of performance curves. Because this will vary according to the underlying hardware, mechanisms must exist to adjust requirements in a variety of ways. Implementation 9 in Figure 2 illustrates this type of operating point.
- *Describe sets of usage modes.* Although they are geared toward certain uses, scenario-oriented processors can provide additional value in other modes. Quantitative evaluation of tradeoffs between primary and secondary modes is necessary. For example, is it better to have an excellent handheld TV or to sacrifice some TV functionality to allow using the same device as a phone?

We propose a fundamental change in the structure of benchmarks for scenario-oriented computing. Accordingly, metrics for evaluation must also change.

Scenario-oriented metric criteria

Metric design is the problem of quantitatively assessing the execution of a benchmark. Scenario-oriented performance metrics should satisfy the following criteria:

- *Differentiate application elements by relative impact on usefulness.* Users do not have equal performance requirements for all tasks or even all portions of a single task. In Figure 2, for example, speech recognition is slightly more important than display because drivers try to avoid looking away from the road, and speech recognition assists with this requirement.
- *Account for nonlinearity.* A hard real-time task will not perform correctly if it can't meet its deadlines. Once it meets all deadlines, however, there might be no benefit to further accelerating that task. Human interaction is similar—beneath some threshold, humans can't perceive faster response times. For example, a graphics task doesn't benefit from frame rates higher than the monitor can support.
- *Describe critical cases.* Understanding the tradeoffs in a scenario-oriented computer requires knowing when the interaction of time, data, functionality, and hardware causes overall performance to degrade. For example, there should be a way to describe exactly what happens in implementation 9 in Figure 2 that causes performance to rapidly drop or to explain the interactions that occur in Figure 3 when downloading a map while displaying a streaming video.
- *Have a visual representation.* Pictures and graphs are powerful tools for rapidly conveying complex information and tradeoffs. Developers often use bar charts in conjunction with SPEC to demonstrate improvements in throughput and latency. Scenario-oriented designers will frequently need to select one approach from numerous alternatives when a single "best" choice is not apparent. Intuitive ways of expressing the results of future benchmarks are necessary to guide such decisions.

These properties for future benchmarks and metrics depart sharply in structure and focus from those that historically have guided computer design. This overhaul is necessary for computer designers to develop and evaluate the new principles required to deliver compelling devices to end users.

Developers must reconsider performance evaluation in light of emerging hardware and software trends. A new era of scenario-oriented computing is dawning. The means to evaluate new performance facilitators and decisions shape, both directly and indirectly, approaches to design and their ultimate success—or failure. The advancement of scenario-oriented design, therefore, hinges on the development of appropriate evaluation methods. We invite the community to join us in considering this challenge. Contact us at soar@ece.wisc.edu or visit www.ece.wisc.edu/~soar for more information. ■

Acknowledgments

This work was supported in part by the National Science Foundation under grants 0607934 and 0606675. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

References

1. J.L. Henning, "SPEC CPU2000: Measuring CPU Performance in the New Millennium," *Computer*, July 2000, pp. 28-35; www.spec.org.
2. M. Levy, "Evaluating Digital Entertainment System Performance," *Computer*, July 2005, pp. 68-72; www.eembc.org.

3. J.M. Paul, D.E. Thomas, and A. Bobrek, "Scenario-Oriented Design for Single-Chip Heterogeneous Multiprocessors," *IEEE Trans. VLSI*, Aug. 2006, pp. 868-880.
4. J.M. Paul and B.H. Meyer, "Amdahl's Law Revisited for Single Chip Systems," *Int'l J. Parallel Programming*, Apr. 2007, pp. 101-123.
5. Sematech, *International Technology Roadmap for Semiconductors* (ITRS), 2005; www.itrs.net/Links/2005ITRS/Home2005.htm.
6. H.P. Hofstee, "Power-Efficient Processor Architecture and the Cell Processor," *Proc. 11th Conf. High-Performance Computing Architectures*, IEEE CS Press, 2005, pp. 258-262.
7. M.J. Schulte et al., "A Low-Power Multithreaded Processor for Software Defined Radio," *J. VLSI Signal Processing Systems*, June 2006, pp. 143-159.
8. S.C. Woo et al., "The SPLASH-2 Programs: Characterization and Methodological Considerations," *Proc. 22nd Ann. Int'l Symp. Computer Architecture*, ACM Press, 1995, pp. 24-36.
9. N. Renqvist and M. Kallinen, "3DMark06," white paper v1.0.2, Jan. 2006; www.futuremark.com/products/3dmark06.
10. J.M. Sammons and C.H. Sauer, "Measuring the Performance of Personal Computers," *Proc. 37th IEEE Computer Society Int'l Computer Conf.* (Compcon 92), IEEE CS Press, 1992, pp. 311-313.

Sean M. Pieper *is a PhD student in the Department of Electrical and Computer Engineering at the University of Wisconsin-Madison. His research interests include scenario-oriented and power-efficient computer architectures. Pieper received an MS in electrical and computer engineering from Carnegie Mellon University. Contact him at spieper@wisc.edu.*

JoAnn M. Paul *is an associate professor in the Department of Electrical and Computer Engineering at Virginia Tech. Her research interests include the design, modeling, simulation, and evaluation of single-chip heterogeneous multiprocessors. Paul received a PhD in electrical engineering from the University of Pittsburgh. She is a member of the IEEE and the IEEE Computer Society. Contact her at jmpaul@vt.edu.*

Michael J. Schulte *is an associate professor in the Department of Electrical and Computer Engineering at the University of Wisconsin-Madison. His research interests include high-performance embedded processors, computer architecture, domain-specific systems, and computer arithmetic. Schulte received a PhD in electrical engineering from the University of Texas at Austin. He is a senior member of the IEEE and the IEEE Computer Society. Contact him at schulte@ece.wisc.edu.*

Sign Up Today

For the IEEE Computer Society Digital Library E-Mail Newsletter

- Monthly updates highlight the latest additions to the digital library from all 23 peer-reviewed Computer Society periodicals.
- New links access recent Computer Society conference publications.
- Sponsors offer readers special deals on products and events.

Available for FREE to members, students, and computing professionals.

Visit **http://www.computer.org/services/csdl_subscribe**

Computer Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page qMags

INTERNATIONAL CONFERENCE ON PARALLEL PROCESSING

Organizing & Program Committees

General Chair
Wu-chi Feng, Portland State Univ., USA

Program Chair
Yuanyuan Yang, SUNY Stony Brook, USA

Program Vice-Chairs

Architecture
Angelos Bilas, Univ. of Crete, Greece

Algorithms and Applications
A. Yavuz Oruc, Univ. of Maryland, USA

Cluster Computing
Olav Lysne, Univ. of Oslo, Norway

Compilers and Languages
Rudolf Eigenmann, Purdue Univ., USA

OS / Resource Management
Wu-chun Feng, Virginia Tech Univ., USA

Network-Based / Grid Computing
Qianping Gu, Simon Fraser Univ., Canada

Peer-to-Peer Technologies
Jiannong Cao, Hong Kong Polytechnic Univ., China

Performance Evaluation
Azzedine Boukerche, Univ. of Ottawa, Canada

Software Systems and Tools
Xin Yuan, Florida State Univ., USA

Wireless and Mobile Computing
Samir Das, SUNY Stony Brook, USA

Program Committee Members
(Please see the conference web page.)

Workshops Chair
Gagan Agrawal, Ohio State Univ., USA

Awards Co-Chairs
D.K. Panda, The Ohio State Univ., USA
Jose Duato, Univ. of Valencia, Spain

Publications Chair
Yanyong Zhang, Rutgers Univ., USA

Publicity Chair
Feng Qin, The Ohio State Univ., USA

International Liaison Co-Chairs
Ten H. Lai, Ohio State Univ., USA
Makoto Takizawa, Tokyo Denki Univ., Japan
A Min Tjoa, Vienna Univ. of Tech., Austria

Local Arrangements Chair
Su-Hui Chang, Portland State Univ., USA

Registration Chair
Elizabeth O'Neill, The Ohio State Univ., USA

Steering Committee Chair
Mike Liu, The Ohio State Univ., USA

CALL FOR PAPERS

- 37th Annual Conference -

2008 International Conference on Parallel Processing (ICPP-2008)

http://www.cse.ohio-state.edu/~icpp2008

Hilton Portland and Executive Tower
Portland, Oregon, USA
September 8-12, 2008

Sponsored by
The International Association for Computers and Communications (IACC)

In cooperation with
Portland State University, USA
The Ohio State University, USA

Scope

The International Conference on Parallel Processing provides a forum for engineers and scientists in academia, industry and government to present their latest research findings in aspects of parallel and distributed computing.

Topics of interest include, but are not limited to:

- Architecture
- Algorithms and Applications
- Cluster Computing
- Compilers and Languages
- OS / Resource Management
- Network-Based / Grid Computing
- Peer-to-Peer Technologies
- Performance Evaluation
- Software Systems and Tools
- Wireless and Mobile Computing

Paper Submission

Paper submissions should be formatted according to the IEEE standard double-column format with a font size 10 pt or larger. Each paper is strictly limited to 8 pages in length. Submissions should represent original, substantive research results. We will not accept any paper which, at the time of submission, is under review for or has already been published (or accepted) for publication in another conference or journal venue. See the conference website for electronic paper submission instructions.

Conference Timeline

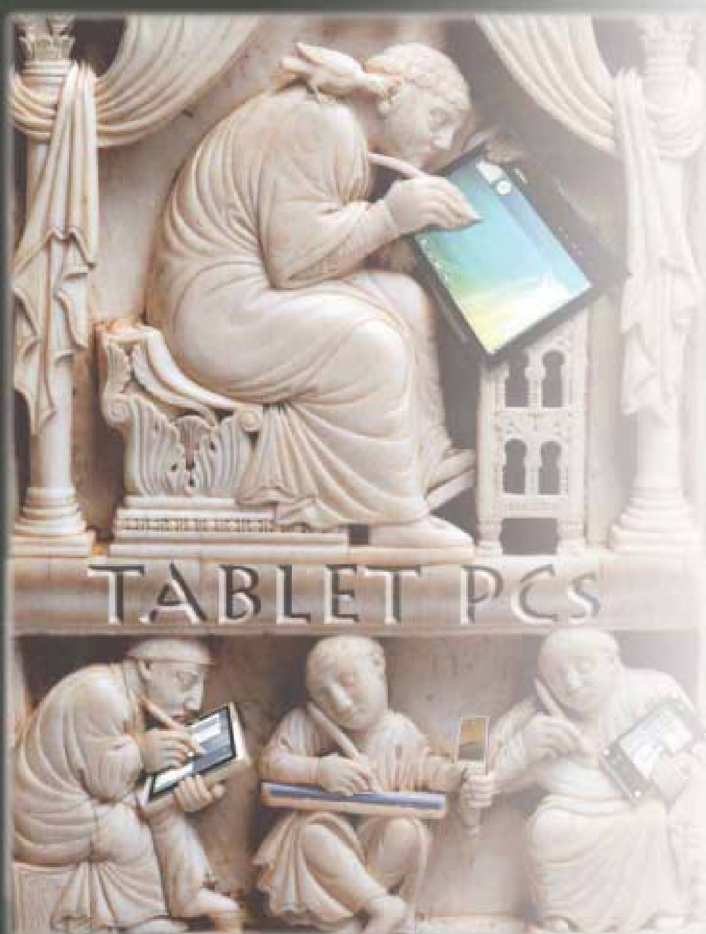
Paper Submission Deadline	February 4, 2008
Author Notification	May 12, 2008
Final Manuscript Due	June 2008

Workshops with more narrowly focused scope will be held on September 8 and September 12. Workshop proposals should be submitted to the Workshops Chair, Dr. Gagan Agrawal (agrawal@cse.ohio-state.edu), by **November 1, 2007**.

Proceedings of the conference and workshops will be published by the IEEE Computer Society in CD format only and will be available at the conference.

For Further Information please contact:
Professor Wu-chi Feng, Portland State University, wuchi@cs.pdx.edu
Professor Yuanyuan Yang, SUNY Stony Brook, yang@ece.sunysb.edu

GUEST EDITORS' INTRODUCTION



Tablet PC Technology: The Next Generation

Jane Prey
Microsoft Research

Alf Weaver
University of Virginia

Early adopters in higher education have developed Tablet PC teaching platforms that incorporate active learning techniques and support in-class collaborations.

Tablet PCs—what are they? Are they just a fad, or do they really play a significant role in the computing milieu?

Wikipedia defines a Tablet PC as "a notebook- or slate-shaped mobile computer. Its touch screen or digitizing tablet technology allows the user to operate the computer with a stylus or digital pen or a fingertip instead of a keyboard or mouse. The form factor offers a more mobile way to interact with a computer. Tablet PCs are often used where normal notebooks are impractical or unwieldy, or do not provide the needed functionality."

The Tablet PC concept is not new; the ideas behind pen-based computing hark back to visionaries such as Alan Kay, Butler Lampson, and Chuck Thacker and to innovative companies such as Xerox PARC in the late 1960s and early 1970s. Alan Kay's Dynabook from 1968, shown in Figure 1, looks remarkably similar to today's machines. But the technology just wasn't ready—many attempts at pen-based computing failed due to an unsuitable user interface, inaccurate handwriting recognition, and a poor digitizer.

Microsoft entered the field in 1999 with its proof-of-concept machine, shown in Figure 2, which was based on the Transmeta TM5800 processor. With 256 Mbytes of RAM, a 20-Gbyte disk, and a 10.4-inch slate form factor, it pushed the power efficiency envelope by providing 5 hours of runtime (200 hours standby). But the machine was not a commercial success because it was so slooooow.

Now all that has changed. As the Gartner Group's "Dataquest Insight: Tablet PCs Are Slowly Gaining Momentum" report indicates, Tablet PCs are here to stay (www.gartner.com). Whereas the insurance, healthcare, public safety, and real estate industries were the first users of tablet technology, Gartner suggests that the next big vertical enterprise to identify the usefulness of Tablet PCs will be higher education. "In the past two years, higher education faculty and students have been the main emerging users adopting the convertible tablet PCs," the report states.

Published by the IEEE Computer Society
0018-9162/07/$25.00 © 2007 IEEE

Computer | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

Early adopters in higher education have developed Tablet PC teaching platforms that incorporate active learning techniques and support in-class teacher/student and student/student collaborations. Novel software is available to grade assignments, conduct online office hours, tutor circuit design, and illustrate concurrent programming—among many other applications.

IN THIS ISSUE

This issue of *Computer* presents five articles that look at tablet technology from the perspectives of application development, research, and instruction.

"Magic Paper: Sketch-Understanding Research" by Randy Davis describes how sketch understanding systems let users draw more naturally and with a freedom not available with CAD systems so that they can create informal figures as a way of thinking or working through a problem.

In "Ink, Improvisation, and Interactive Engagement: Learning with Tablets," Jeremy Roschelle and colleagues discuss how today's approaches to teaching have moved beyond the notion of learning as recollection to new metaphors and approaches. They describe how communities that form around platforms such as Classroom Presenter, GroupScribbles, and related applications should provide an excellent forum for advances in the use of tablets in educational settings.

In "Handwriting Recognition: Tablet PC Text Input," Jay Pittman declares that pen-based computing is here now and describes how a large neural network trained on a very large training set containing ink samples from thousands of people with a diverse range of writing styles is being used to support poorly formed cursive script, providing amazing recognition accuracy, especially once users acclimate to writing with a plastic pen on a plastic screen.

"Classroom Presenter: Enhancing Interactive Education and Collaboration with Digital Ink" by Richard Anderson and colleagues discusses a Tablet PC-based interaction system that supports the sharing of digital ink on slides between instructors and students and describes its use in classroom interaction scenarios. They demonstrate that the technology offers flexibility and a range of expression that can achieve a wide range of educational goals and foster a more participatory classroom environment.

In "Facilitating Pedagogical Practices through a Large-Scale Tablet PC Deployment," Joe Tront describes the multifaceted, collaborative approach that faculty and students at the Virginia Tech College of Engineering are using. They have begun exploration of the use of Tablet PCs in engineering and computer science courses, working with an implementation process that includes computer acquisition, faculty training, infrastructure modifications, and multiple learning assessments. Preliminary results have been quite positive.

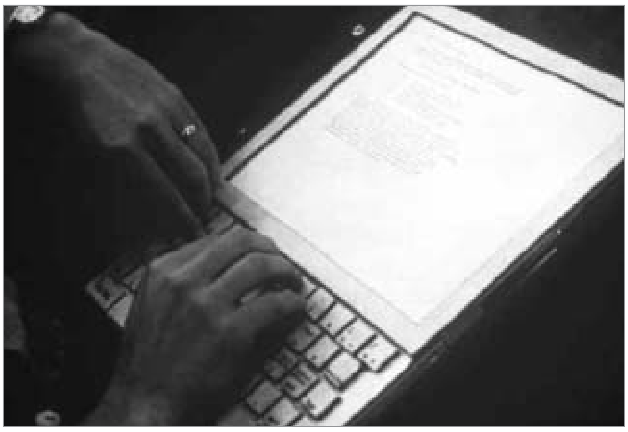


Figure 1. Alan Kay's Dynabook mockup, first shown in 1968.

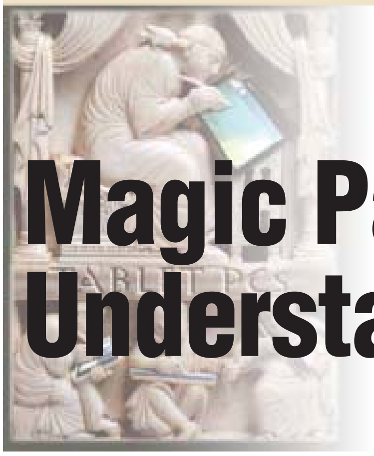
Figure 2. Microsoft's proof-of-concept Tablet PC based on Transmeta's early TM5800 processor (1999).

We hope that these articles will raise your awareness and curiosity about tablet technology and encourage you to think about how you might apply it in your own work. Thanks for your interest.

Jane Prey *leads the Tablet Technologies in Higher Education and the Gender/Pipeline initiatives at Microsoft Research. Prey received a PhD in instructional technology from the University of Virginia. She is a member of the ACM Education Board as well as a former member of the board for ACM SIGCSE. Contact her at jprey@microsoft.com.*

Alf Weaver *is a professor in the Department of Computer Science, School of Engineering and Applied Science, at the University of Virginia. His research interests include networks, e-commerce, computer security, and computer applications to medicine. Weaver received a PhD in computer science from the University of Illinois. He is a Fellow of the IEEE and a member of the ACM. Contact him at acw@cs.virginia.edu.*

COVER FEATURE



Magic Paper: Sketch-Understanding Research

Randall Davis
Massachusetts Institute of Technology

Sketches are hand-drawn informal figures often created as a way of thinking about or working through a problem. Sketch-understanding systems let users interact with computers by drawing naturally, offering a freedom not available with traditional CAD systems.

Sketching is ubiquitous: We draw as a way of thinking, solving problems, and communicating in a wide variety of fields, for both design (such as sketches of conceptual designs) and analysis (such as sketches drawn to help puzzle through problems in physics or electronic circuits).

Unfortunately in today's technology, sketches are dead—they're either graphite on slices of dead trees, or, if captured on a PDA or tablet computer, simply pixels of digitized ink. The Sketch Understanding Group at MIT has been working toward a kind of "magic paper"—that is, a surface that's as natural and easy to draw on as paper, yet that understands what you draw.

What does it mean for the paper to "understand"? One example, in Figure 1, shows some of our earliest work. We use Assist (A Shrewd Sketch Interpretation and Simulation Tool) to sketch simple 2D physical devices, then watch them behave.[1] Assist understands the raw sketch shown in Figure 1a in the sense that it interprets the ink the same way we do (Figure 1b), that is, as an inclined plane with a wheeled cart. As Figure 1c shows, it hands this interpretation to a physics simulator, which animates the device, giving the user the experience of drawing on intelligent paper.

One detail helps illustrate the sense in which the system understands the sketch in a manner similar to a human observer. The wheels (blue circles) are attached to the car's body with pin joints (pink circles), yet the user draws both the wheels and pin joints with the same geometric shape—a circle. The system interprets a circle differently based on context—a circle can be a pin joint only if it's drawn over two bodies that are already overlapping (in this case the car body and the wheel).

We've built sketch-understanding systems for a variety of domains, including the simple physics sketcher shown in Figure 1, Unified Modeling Language diagrams (Figure 2), analog circuits (Figure 3), and chemical structure sketches[2] (Figure 4).

TERMINOLOGY AND FOCUS

Sketches differ from diagrams. By "diagrams" we mean the more formal, at times draftsman-like figures that CAD systems produce, while sketches are the hand-drawn informal figures people create on paper, whiteboards, napkins, and, more recently, tablet computers. There is a significant body of work on understanding diagrams such as the International Conferences on Document Analysis and Recognition, but the task of understanding a sketch is significantly different.

Research on sketch understanding also differs from the sizable body of work on handwriting understanding. Sketch-understanding work proceeds largely from an attempt to recognize the shape of the objects drawn, using the same notion of shape that people use. Handwriting recognition, on the other hand, doesn't attempt to recognize each letter by its shape, and has successfully used machine-learning techniques that derive distinguishing features that might or might not correspond to what people attend to.

Published by the IEEE Computer Society
0018-9162/07/$25.00 © 2007 IEEE

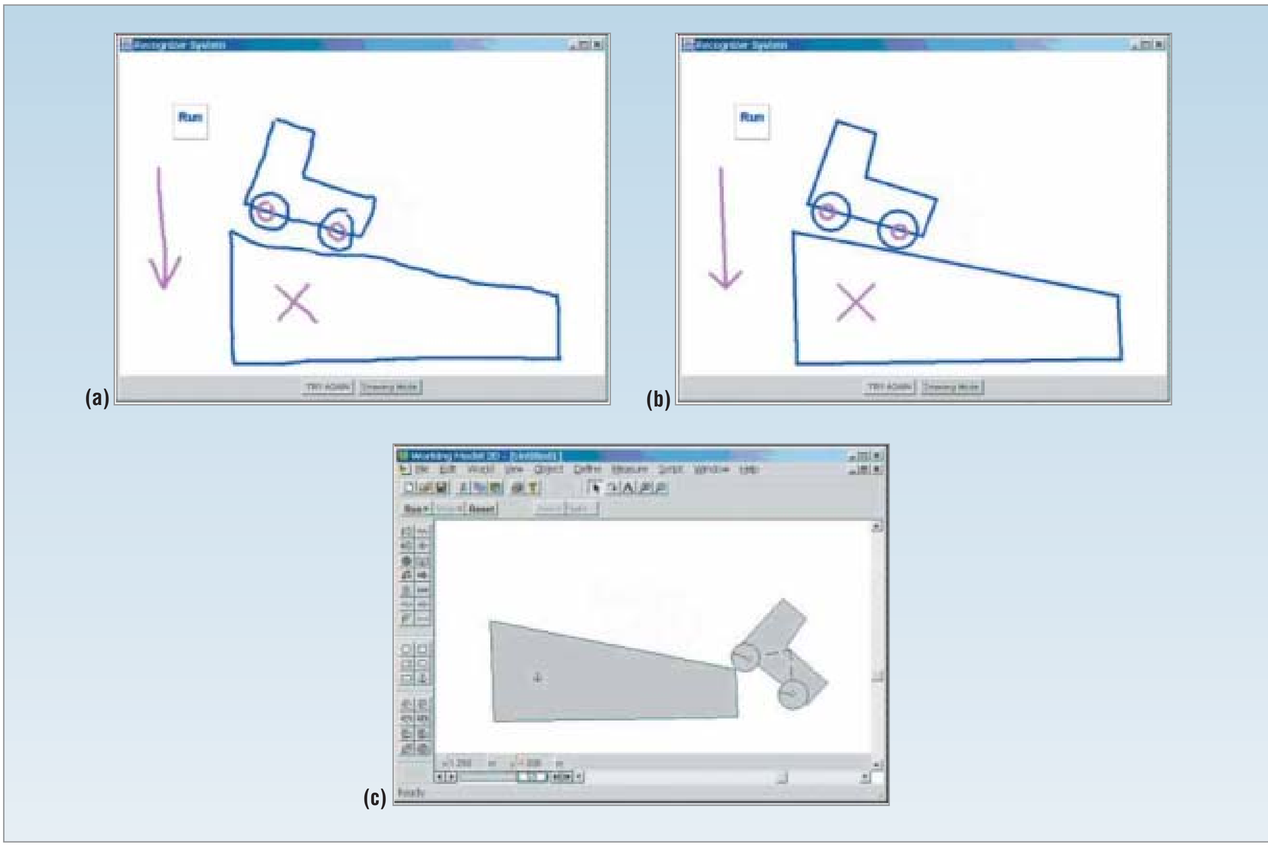


Figure 1. Assist system. (a) The initial sketch, (b) the sketch as cleaned up by Assist, and (c) the simulation, showing the consequences.

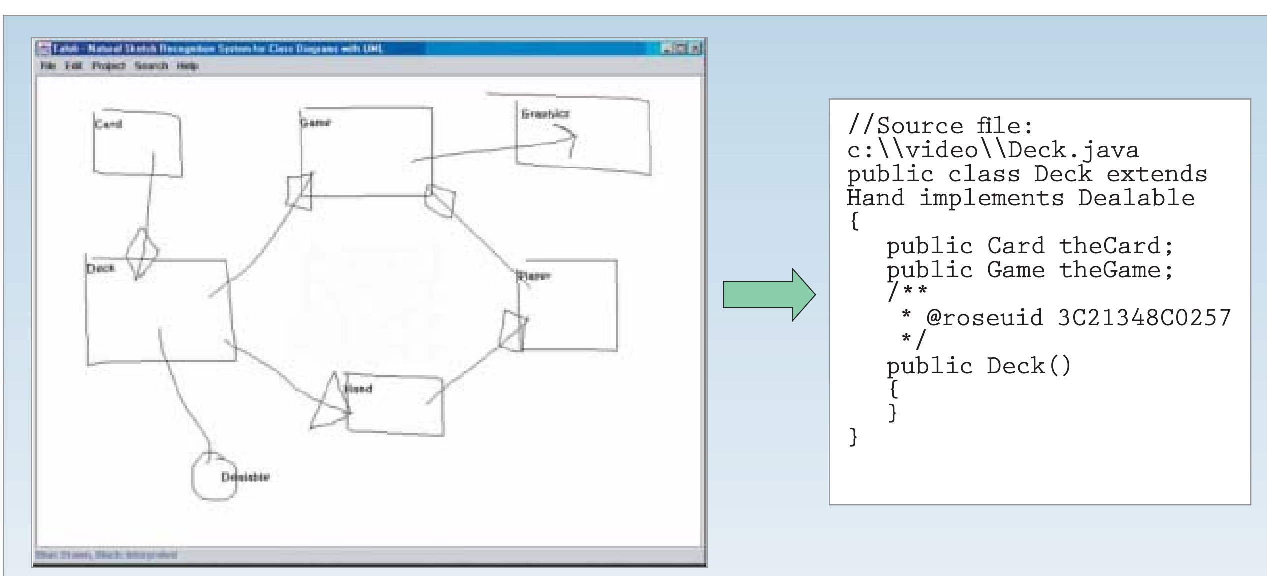


Figure 2. A UML sketch recognized then turned into code using RationalRose.

WHY SKETCH?

Given the prevalence and power of design-automation tools, it's reasonable to ask, why sketch when we have tools to create far more precise and polished drawings?

Two things motivate the desire to enable sketching.

First, picking up a pen is still more natural than using a keyboard and mouse, and free-form sketched figures seem more intuitive than the formal shapes a CAD program generates. There's a freedom of thought that seems to go with the ease of sketching, an intuition supported by evi-

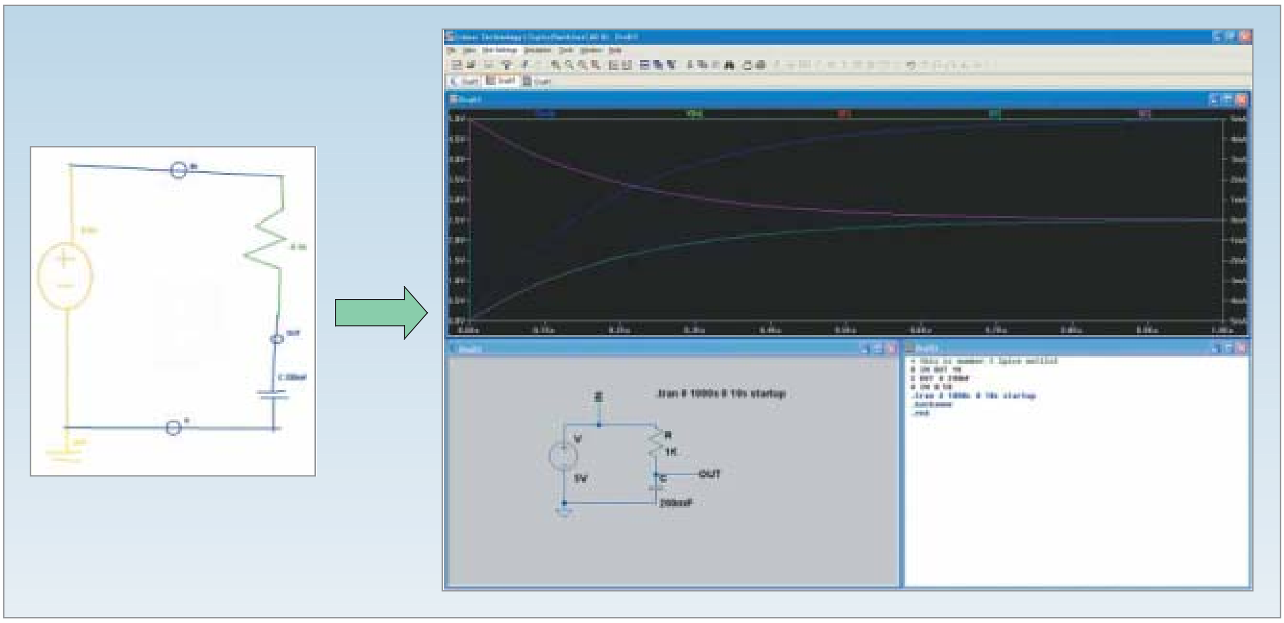


Figure 3. An analog circuit sketch recognized, then analyzed by Spice.

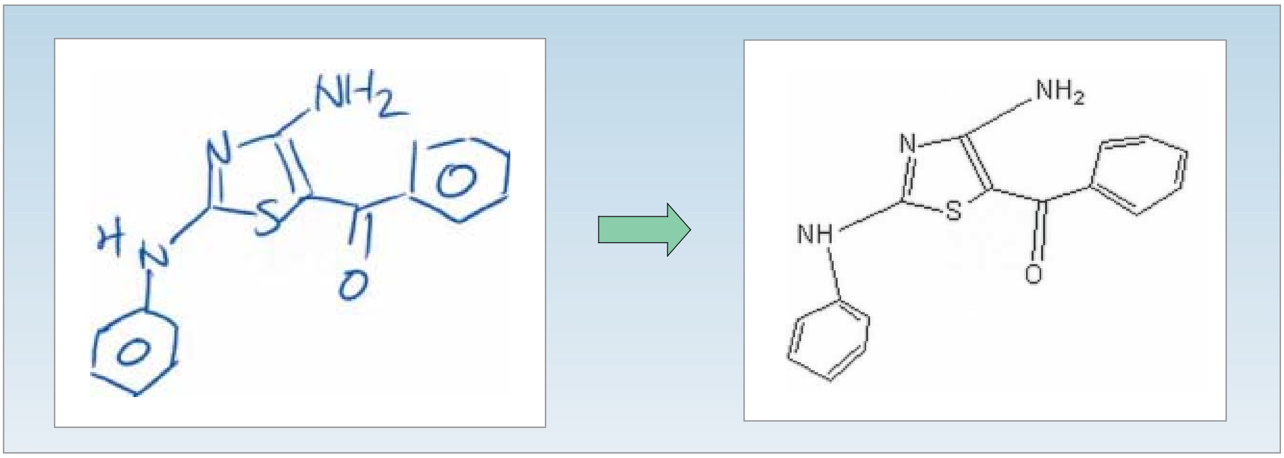


Figure 4. A chemical structure sketch recognized, then redrawn by ChemDraw.

dence from cognitive science research, which showed that designers who sketched produced more design alternatives than did those who were using a drafting tool.[3]

Second, and perhaps more important, CAD systems typically require commitments that the sketcher might not want to make, particularly at the early conceptual design stage, or when sketching a rough picture for an analysis problem. It is, for example, impossible simply to draw a line in, say, a mechanical CAD system: Every line has a precise dimension, angle, and so on. Yet, when you want to dash off an idea as to how a device might work, the need to specify precise sizes, angles, radii, and so on gets in the way.

DIFFICULTIES WITH SKETCH INTERPRETATION

Sketch understanding seems so quick and intuitive when we do it that we may wonder why creating software to do the same thing is so difficult. One way to see the difficulties in the task is to view the problem as one of signal interpretation (interpreting a time-stamped sequence of points). Many of the standard signal interpretation issues arise, along with some novel ones.

Sketch interpretation in general is difficult in direct proportion to the user's allowed degree of freedom—the less constrained the drawing style, the more difficult the interpretation task. At one end of the spectrum lie techniques such as Graffiti, which prescribes a specific set of gestures to be drawn in a specified manner. In these circumstances, recognition is considerably easier, and the computational demand relatively modest. The tradeoff is the necessity of learning and using a particular drawing style. An intermediate position is taken by work like that in Silk,[4] which suggests doing minimal interpretation of the strokes, in part because the system is

Computer | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

intended as a design platform, and in part to avoid interfering with the creative process.

Our work lies closer to the unrestricted end of the spectrum. We want people to be able to draw as they would normally, without specifying the number, order, or direction of strokes in a symbol, yet still have the system understand. Figure 5 illustrates several difficulties that arise from this.

First, our task is incremental. That is, we want the system to interpret users' strokes as they're drawn. This lets the system provide continuous feedback about its understanding of the sketch, so the user can make corrections when a misunderstanding arises. But it also means that interpretation must be continually revised. In Figure 5, for example, the user has not yet drawn the line connecting the circle and rectangle in the upper part of the sketch that will indicate a married couple.

In addition, as in any signal interpretation problem, there is noise. In this case, noise arises from our inability to draw with mechanical precision: Lines wiggle, curves intended to be closed might have a gap or an overlap, and so on.

Next, the drawing conventions in many domains permit variations. In analog circuits, for example, you can draw capacitors as two straight parallel lines or as a straight line and a curved line.

Individual styles also vary, across users and even within a sketch. In Figure 5, the user drew the three successive rectangles in the lower part of the figure, yet drew them in three different ways (using a single stroke, a squared C-shape closed by a line, and two L-shaped strokes).

Another issue is the difficulty of segmentation. Consider the arrow pointing toward the rectangle at the bottom left in Figure 5. The arrowhead touches the rectangle, but isn't connected to the shaft. Determining which strokes might belong together requires understanding the domain.

Next, difficulties arise when the user lays down extensive amounts of ink, as when strokes are overtraced or when figures are filled in—for example, the filled-in triangles used in drawing some arrowheads. The problem here is that the fundamental view of the sketch as a set of strokes or time-stamped points breaks down. A filled-in shape is a particularly clear example: It's of little use to represent the ink in the interior of the shape as a sequence of strokes. It might have been laid down that way, but the intent was to create a 2D area of continuous ink; the particular set or sequence of strokes used is irrelevant. Overtracing is similar. It's the ink's appearance that is intended to create the line. The numerous (possibly zigzagging) strokes used to lay down the ink are irrelevant. This calls for a change of representation, from the sketch as a sequence of strokes to an area of ink.

Finally, and perhaps most interesting, the signal is both 2D and nonchronological. It's 2D in the obvious sense that, unlike normal handwriting, it spreads across the page. In this it's similar to but more unrestricted than handwritten mathematical formulas.

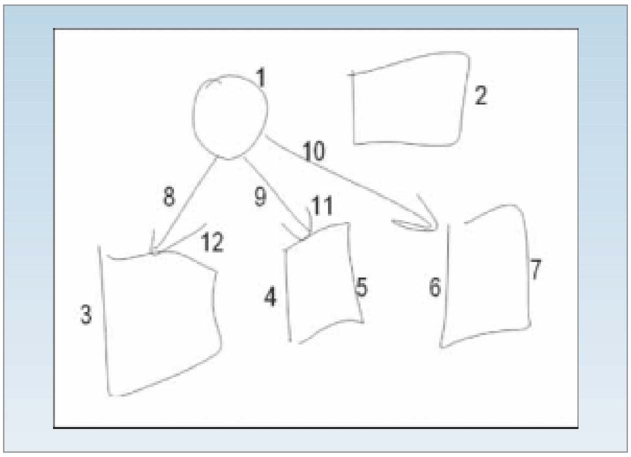


Figure 5. Snapshot of a family tree as a user is drawing it. Squares indicate males, circles indicate females, arrows indicate offspring, and a straight line (not yet drawn) indicates a marriage. Numbers have been added to indicate stroke order.

The signal is nonchronological in the sense that we don't require each object to be finished before the next is started, so a user might add strokes to a sketch to complete something started earlier. For example, as the numbers in Figure 5 indicate, the user drew the arrows' shafts in sequence, and only later added the heads.

This differs from other signal interpretation problems, such as speech recognition. When talking you might restate something, but you can't go back in time and change the sounds you made earlier. Yet in sketching, newly added strokes can change the interpretation of strokes made earlier.

SKETCH UNDERSTANDING

Two basic assumptions ground most work in sketch understanding.

First, the work is done in domains where there's a reasonably well-established graphical lexicon and grammar. The lexicon is the set of shapes used in a domain—for example, the standard graphical notation for digital circuit components. The grammar describes interrelations among shapes, indicating, for example, that a transistor symbol should be connected to three other components. Many scientific and engineering domains have such a graphical lexicon and syntax, but this is clearly not universal. The sort of impressionistic sketches drawn by architects early in the design process, for example, are not describable in these terms.

Second, much like work in speech understanding, sketch-understanding systems are built for a specific domain. Unrestricted sketch understanding is currently out of reach, for many of the same reasons that unrestricted speech understanding isn't particularly successful.

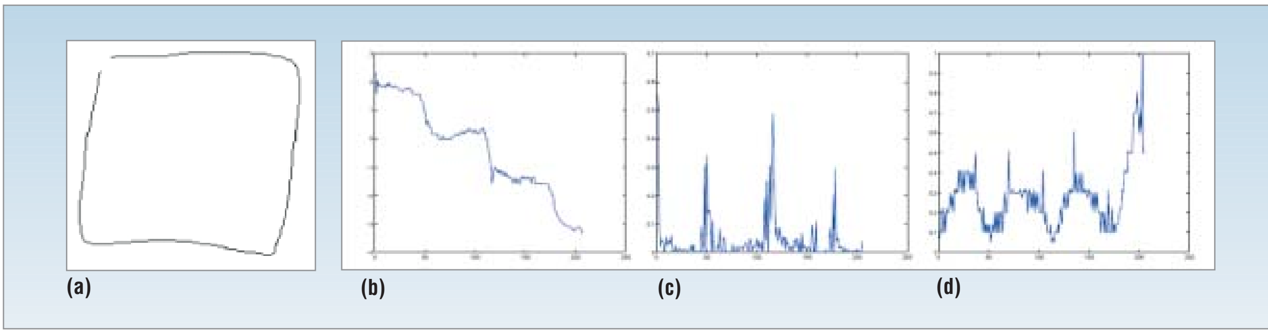


Figure 6. Characteristics of a drawing. (a) A hand-drawn square with corresponding graphs showing the drawing's (b) direction, (c) curvature, and (d) speed.

Representing a sketch

At the most basic level, a sketch is a collection of time-stamped coordinate points, grouped into individual strokes—that is, from pen down to pen up. As with handwriting recognition, online and offline sketching differ. Offline sketches must be scanned and lines located, introducing additional noise, while the raw data of online sketches is composed of strokes whose locations and widths are known precisely.

Finding primitives

One basic task common to much sketch-recognition research involves reinterpreting the raw data as primitive shapes—lines and arcs. This low-level processing is an interesting task by itself, given the noisiness of the data.

One common approach to finding primitives uses the data's temporal character, based on the observation that, when drawing by hand, people routinely slow down at corners, without consciously attempting to do so.[5] This lets the system locate corners more precisely by combining information about curvature and speed, looking for points that combine high curvature and low speed.

Although combining information from these two sources helps, the problem is still difficult, due to the fine-grained noisiness of the data (Figure 6). In response, we have explored the use of scale-space filtering to remove the noise, relying on the basic intuition (common to all scale-space approaches) that the signal and noise have different spatial frequencies.[6] As those frequencies are not generally determinable a priori, that work uses dynamic selection of the appropriate scale from the data itself.

After locating the corners, we represent strokes as piecewise linear approximations, with additional processing merging nearly collinear segments, and using arcs (for example, Bezier curves) where an arc is a better fit than a line or line sequence.[5]

We can then use the resulting lines and arcs as the raw material in any of the representation and recognition approaches described next.

Recognizing shapes

The variety of information available in online sketches enables three distinctly different representations, with three correspondingly different approaches to recognition:

- how the shape is defined—the set of geometric constraints the shape must obey to be an instance of a particular object;
- how the shape is drawn—the sequence of strokes used; and
- what the shape looks like—the traditional concept of image appearance.

We've built recognition systems to explore each of these representations individually and are working to understand how to combine them.

How it's defined. We represent a shape's definition in a language called Ladder.[7] As Figure 7 shows, Ladder descriptions list the basic components making up the shape and the geometric constraints that must hold between them. Recognizing individual shapes is then a process of matching the strokes drawn against a set of shape descriptions for a domain.[8]

The description is significant both for what it constrains—for example, to be an arrow the shaft must be longer than the heads—and for what it doesn't constrain—for example, it doesn't specify the lines' orientation or length, letting the system recognize arrows of any size and in any orientation.

How it's drawn. When drawing the standard symbol for an AND-gate (Figure 8a), most people start with the vertical stroke, add the arc, then draw the connecting wires. This sequence, along with just a few others, accounts for most of the routinely encountered stroke orders. This is true for a wide variety of shapes, not just those as simple as in Figure 8.

This observation provides a foundation for recognition by observing stroke sequence, using a variation on a hidden Markov model (HMM) called a dynamic Bayes' net. We train the DBN to recognize shapes based on the sequence of primitive geometric elements such as lines and arcs found in the strokes.[9]

```
components
 Line shaft
 Line head1
 Line head2
constraints
 longer shaft head1
 equal head1 head2
 coincident head1.p1 head2.p1
 coincident head1.p1 shaft.p1
 acuteMeet head1 shaft
 acuteMeet shaft head2
```

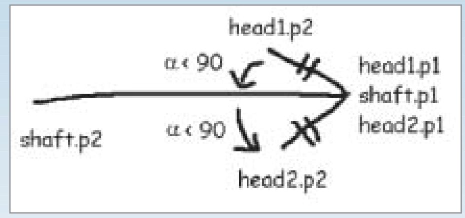


Figure 7. The Ladder description of an arrow. The definition includes the shape's basic components and their geometric constraints.

One standard simplifying assumption in an HMM-based approach is that each shape is completed before the next is started. This assumption puts a tractable limit on the number of states the HMM must have. If you can arbitrarily intersperse shapes, the number of states needed in the HMM increases exponentially. Although people don't typically intersperse shapes arbitrarily, neither do they reliably finish one object before drawing another. When drawing the transistor in Figure 8b, for example, people routinely draw the vertical connecting wire strokes before the arrowhead strokes that complete the transistor itself. We have extended the DBN approach to learn and then deal with the sort of interspersing encountered in real-world practice.[9]

How it looks. The long history of image-understanding research provides various approaches to recognizing a sketch by its appearance. Several of these techniques have been used in sketch understanding, typically by matching the user's strokes (viewed as bitmaps) against templates, using some combination of traditional distance metrics—for example, the Hausdorff distance, Tanimoto coefficient, and so on.[10]

One difficulty with sketches is that, unlike images, we can't assume that the template and image are related by an affine transform. Hand-drawn sketches of symbols routinely exhibit variations that are nonuniform over the symbol, making the matching task more challenging, inspiring efforts to develop variations on the traditional metrics.

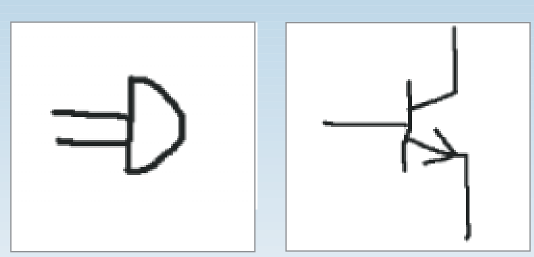
Recognizing a sketch by how it looks is useful in dealing with situations such as overtracing and filled-in shapes, where the stroke-based representation breaks down. Here we need to determine whether the ink looks like a line, no matter what sequence or collection of strokes produced the ink.

One recent approach inspired by work in image processing attempts to capture local visual features, rather than match an entire template.[11] It builds on the notion of shape contexts, measuring the ink's placement and orientation at various points in a shape using a small circular bull's-eye pattern. A collection of these patterns forms a characterization of the entire shape, which we can then compare to similar characterizations made for template figures.

SKETCH-ENABLED INTERFACES

Given the ability to translate strokes into object descriptions, we can connect sketch understanding to various back-end programs. We produced the simulation in Figure 1, for example, by sending our sketch understander's output to a physics simulator and having it compute and animate the resulting behavior.

We've experimented with several other back-end programs as well. We have, for example, connected a UML sketch-understanding system to RationalRose. Users sketch a class hierarchy and RationalRose generates class declaration code for that hierarchy. In another application, we linked a program that understands sketches of chemical structures to ChemDraw, which redraws the structure neatly so we can check our interpretation. We also plan to link our chemical sketch understander to one or more databases of information about chemical compounds. These databases contain massive amounts of useful information, but currently must be searched by providing one of the text-based structural encodings (such as Smiles or the International Chemical Identifier), which are nonintuitive, to say the



(a) (b)

Figure 8. Hand-drawn shapes. (a) A hand-sketched AND-gate. Most people start this drawing with the vertical line, then add the arc, and the connecting wires. (b) A hand-drawn transistor.

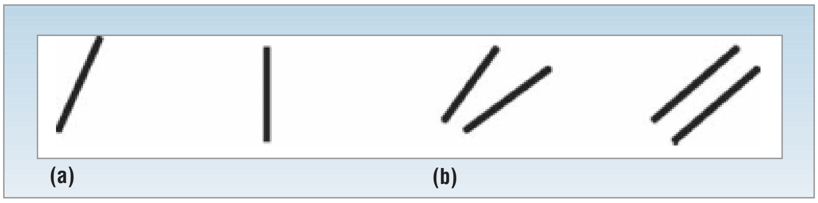


Figure 9. Line orientations. (a) A slanted and a vertical line. Rotate a slanted line slightly and it's still a slanted line, but rotating a vertical line changes its verticality property. (b) Two slanted lines and two parallel lines. Rotating one of a pair of parallel lines can change the parallelism property.

least. For example, the InChI language encodes the elementary compound L-ascorbic acid as `InChI=1/C6H8O6/c7-1-2(8)5-3(9)4(10)6(11)12-5/h2,5,7-10H,1H2/t2-,5+/m0/s1`. The complexity of this notation arises from the need to capture in linear text information about connectivity, 3D orientation, and so on. It will be far easier and more natural simply to draw a molecule and use that as the query.

Linking sketch-understanding systems to back-end applications allows us to sketch-enable those applications, expanding the modalities used to interact with them, and consequently vastly simplifying their use.

LEARNING A NEW DOMAIN

When sketch understanding is done by matching descriptions against strokes, creating a sketch understander for a new domain can be as straightforward as writing a new set of shape definitions—much as some speech recognizers let users write lexicons and grammars for a new domain. This is considerably easier than writing custom code for recognizers and lets nonprogrammers develop sketching interfaces.

Writing shape descriptions for a new domain, however, isn't always easy. It is challenging at times because even some simple shapes might involve multiple lines of text (for example, the 11 lines needed to describe an arrow). Writing descriptions is also challenging because shape definitions have inevitable subtleties, in the form of constraints that are typically not obvious until they're violated (for example, the barbs of an arrow need to meet the end of the shaft at an acute angle). In practice, people find it difficult to think of all the constraints a shape must meet, leading to definitions that are underconstrained. Other times, they write a description while thinking of a particular instance that isn't general enough, leading to overconstrained definitions.

We've explored a variety of techniques for learning shape definitions by example, attempting in particular to learn from a single example. The problem is intriguing because going from a specific drawing to an appropriately generalized description is a classic problem in learning. The fundamental question is, what about the specific example is essential and what is accidental?

This is a particularly challenging question where drawings are concerned because they contain so much information. When viewed geometrically, every line has a length and orientation, intersection points with other lines, numerous pairwise relations—*parallel*, *perpendicular*, *near*, and *far*—as well as higher arity relations, such as *between*. Separating the essential from the accidental thus means selecting the important properties from numerous irrelevancies.

Consider the drawing in Figure 8a. If we're told that this is an example of an AND-gate, precisely which geometric properties are essential to its being an AND-gate? Does it matter that the arc is connected to both ends of the vertical line? Does it matter that the ratio of the lengths of the two parallel lines is 1.034? Even someone unfamiliar with the domain would likely guess that the answers are yes and no, respectively. But how might a program even begin to arrive at these answers?

One useful insight comes from work by some early 20th-century psychologists, who suggested that our perceptual systems are drawn toward detecting what they termed *singularities*—geometric properties that are in a sense fragile and hence unlikely to have been accidental. Figure 9 illustrates the concept: Most people would describe the first line in Figure 9a as slanted—and the next line as vertical—its verticality jumps out at us. The verticality property is "fragile" in the sense that the slanted line, if rotated slightly, is still a slanted line, while a vertical line rotated slightly loses its verticality property. The situation is similar for properties like horizontal, parallel (Figure 9b), and perpendicular.

We've used this insight to select from among the profusion of properties present in a sketch those relations that people are likely to attend to, using them as a reasonable first-pass guess at the essential properties.[12] The intuition here is that a graphical notation is more likely to be usable if it depends on the geometric properties that our eyes and brain tend to pick out. Hence, those properties are more likely to be the sketch's essential properties.

We've built on this work by engaging the user in a description-refinement process, using an interesting variation on near-misses as a learning vehicle.[13] Where past work on near-misses has relied on the user to offer the system near-miss examples, our research has shown that the program can generate its own near misses, querying the user about their status as examples or nonexamples of the concept. This has the well-known advantage of speeding learning, while having the system take on the difficult task of determining what example would be most informative to consider next.

We've also explored learning from multiple examples. This has the familiar problem of requiring numerous such examples, as well as the less familiar difficulty of getting examples that explore a sufficiently wide range of possibilities. As one trivial example, when asked to draw examples of arrows and near-miss arrows, people rarely think of the near-miss arrow that has its shaft shorter than the arrowheads.

SKETCHING AND MULTIMODALITY

Watch anyone draw in a routine environment—for example, on a whiteboard during a meeting—and you'll almost invariably observe that they speak and gesture as they draw. Not infrequently, their gestures and words are essential to making sense of the sketch, adding information not representable in the drawing. As a consequence, researchers have shown considerable interest in the ability to use all these modalities, leading to several efforts that have explored various modality combinations, producing proof-of-concept demonstrations of their value and plausibility.[14, 15]

Sketch recognition today is at a stage roughly analogous to the early years of speech recognition, when isolated words could routinely be recognized, but continuous speech was still a research goal. Several sketch-understanding systems and approaches deal with individual symbols reasonably well, even those of moderate complexity, with a reasonable amount of noise, overtracing, and so on. No correlate yet exists to speech-recognition system grammars—in sketching, the inter-symbol relations—that approaches the complexity manageable in speech work, but this is one important direction of effort and progress in the field.[7]

Future work also seeks to extend recognition capabilities beyond line drawings to the more impressionistic sketching of engineers, where the profusion of strokes and the depiction of 3D structure present a set of difficult challenges.

Efforts at sketch understanding are in some ways an attempt to get back to the future—we want to return to the world where we can pick up a drawing implement and freely sketch out an idea, inspiration, or depiction of a problem to be solved, but at the same time, have a sketching surface that is intelligent enough to understand what's being drawn and thus can facilitate the design and analysis process. It's an intriguing undertaking, one with rapid current progress and the promise of considerable payoff. ■

Acknowledgments

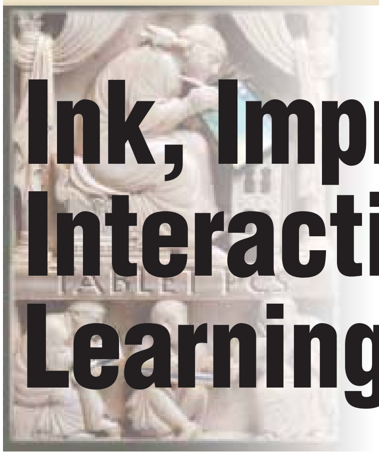
I'd like to acknowledge the work of numerous collaborators whose impressive contributions are summarized briefly here, as well as the organizations that have supported this work, including Ford Motor Co., the MIT/Oxygen Project, Microsoft, Intel, and the Pfizer Corporation.

References

1. C. Alvarado and R. Davis, "Resolving Ambiguities to Create a Natural Sketch-Based Interface," *Proc. Int'l Joint Conf. Artificial Intelligence* (IJCAI), AAAI Press, 2001, pp. 1365-1371.
2. T.Y. Ouyang and R. Davis, "Recognition of Hand-Drawn Chemical Diagrams," *Proc. AAAI 2007*, CD-ROM, AAAI Press, 2007.
3. V. Goel, *Sketches of Thought*, MIT Press, 1995.
4. J. Landay and B. Myers, "Sketching Interfaces: Toward More Human Interface Design," *Computer*, Mar. 2001, pp. 56-64.
5. T.M. Sezgin, T. Stahovich, and R. Davis, "Sketch-Based Interfaces: Early Processing for Sketch Understanding," *Proc. Workshop Perceptive User Interfaces*, ACM Press, 2001, pp. 1-8.
6. T.M. Sezgin and R. Davis, "Scale Space-Based Feature Point Detection for Digital Ink," *Proc. AAAI Fall Symp. Series 2004: Making Pen-Based Interaction Intelligent and Natural*, AAAI Press, pp. 145-151.
7. T. Hammond and R. Davis, "Ladder: A Language to Describe Drawing, Display, and Editing in Sketch Recognition," *Proc. Int'l Joint Conf. Artificial Intelligence* (IJCAI), AAAI Press, 2003, pp. 461-467.
8. T. Hammond and R. Davis, "Automatically Transforming Symbolic Shape Descriptions for Use in Sketch Recognition," *Proc. AAAI*, AAAI Press, 2004, pp. 450-456.
9. T.M. Sezgin, "Sketch Interpretation Using Multiscale Stochastic Models of Temporal Patterns," PhD thesis, Dept. of Electrical Eng., MIT, 2006.
10. L.B. Kara and T.F. Stahovich, "An Image-Based, Trainable Symbol Recognizer for Hand-Drawn Sketches," *Computers & Graphics*, vol. 29, no. 4, 2005, pp. 501-517.
11. M. Oltmans, "Envisioning Sketch Recognition: A Local Feature-Based Approach to Recognizing Informal Sketches," PhD thesis, Dept. of Electrical Eng., MIT, 2007.
12. O. Veselova and R. Davis, "Perceptually Based Learning of Shape Descriptions," *Proc AAAI*, AAAI Press, 2004, pp. 482-487.
13. T. Hammond and R. Davis, "Interactive Learning of Structural Shape Descriptions from Automatically Generated Near-Miss Examples," *Proc. Intelligent User Interfaces*, ACM Press, 2006, pp. 37-40.
14. E. Kaiser et al., "Demo: A Multimodal Learning Interface for Sketch, Speak and Point Creation of a Schedule Chart," *Proc. Int'l Conf. Multimodal Interfaces* (ICMI), ACM Press, 2004, pp. 329-330.
15. A. Adler and R. Davis, "Speech and Sketching for Multimodal Design," *Proc. 9th Int'l Conf. Intelligent User Interfaces*, ACM Press, 2004, pp. 214-216.

***Randall Davis** is a professor in the Department of Electrical Engineering and Computer Science at the Massachusetts Institute of Technology. His research interests include sketch understanding and multimodal interaction. Davis received a PhD in artificial intelligence from Stanford University. He is a Founding Fellow of the Association for Artificial Intelligence and served terms as both councilor and president. Contact him at davis@csail.mit.edu.*

COVER FEATURE



Ink, Improvisation, and Interactive Engagement: Learning with Tablets

Jeremy Roschelle, SRI International
Deborah Tatar, Virginia Tech
S. Raj Chaudhury, Christopher Newport University
Yannis Dimitriadis, University of Valladolid
Charles Patton and Chris DiGiano, SRI International

Instructional models that reflective educators develop and share with their peers will primarily drive advances in the use of tablets in education. Communities that form around platforms such as Classroom Presenter and Group Scribbles should provide an excellent forum for such advances.

> *Mark now the farther development. I shall only ask him, and not teach him, and he shall share the enquiry with me: and do you watch and see if you find me telling or explaining anything to him, instead of eliciting his opinion. Tell me, boy, is not this a square of four feet which I have drawn?*
>
> Socrates[1]

In one of the earliest recorded reflections on pedagogy, Socrates drew a figure in the sand to demonstrate to Meno that the slave already knew how to construct a square with double the area of a given square. Socrates aimed to show that he didn't need to teach, only to engage his student's memory of knowledge from a past life.

Today, we no longer think of learning as merely "recollection." Instead, we see learning as an active, constructive process that builds on prior knowledge. More generally, the learning sciences have led to important advances in our understanding of the process of learning as involving

- learners' active engagement,
- a focus on knowledge construction,
- feedback and formative assessment leading to adaptive instruction, and
- participation in a community of learners.[2]

Despite our changed view of learning, necessitated by recent scientific research, Socrates' pedagogical use of learning technology remains relevant. The Greek philosopher improvised an informal sketch in the sand to structure an interactive, engaging learning experience. Why didn't Socrates prepare his points before class, using the PowerPoint of his day: a chisel and block of stone? A few possible reasons come to mind.

First, his hand-drawn sketch was likely more expressive of the key concepts Socrates wished to communicate than a neat, chiseled presentation would have been. Second, an informal sketch might have invited the participation of his student in active reasoning more effectively than a more formal, fixed diagram would have. Third, the act of drawing, gesturing, and speaking in close synchrony let Socrates focus his student's attention on the meaning of the diagram he was preparing. Fourth, by asking probing questions, Socrates learned much about his student's existing state of knowledge, letting him adapt his instruction to his student's needs.

This example highlights why today's classroom teachers might prefer Tablet PCs with their constellation of affordances, and the right software, over desktop or laptop computers. Compared to typing, ink can express important ideas more vividly.[3] For example, handwrit-

Published by the IEEE Computer Society
0018-9162/07/$25.00 © 2007 IEEE

Computer Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page qMags

ing expresses mathematical notation more naturally than typing.[4] With ink, teachers can also highlight and annotate over words and diagrams, thus focusing student attention on the key features of those visual representations while gaining the efficiency of preparing complex visual aids in advance.[5]

Tablet PCs can also help with adapting instruction and inviting participation. Today's classrooms are becoming fully wireless, which lets teachers harvest and aggregate students' contributions.

Transforming classroom practices around collected and aggregated student work could have profound learning implications.[6,7] In particular, many educators have experimented with student response systems, often called *clickers*. In a model application of such systems,[8] a lecturer asks students a probing multiple-choice question. At first, students anonymously respond with their answers and results aggregated into a histogram that lets the students and teacher see the pattern of responses in the classroom. In the most common case, there are some right and some wrong answers. Students are then encouraged to discuss the question with their neighbors and to convince each other of the answer. The teacher then takes a follow-up poll. Depending on the results of this poll, the teacher then adapts instruction. If only a few students got the right answer, the teacher knows to reteach the material via a complementary approach. If most students got the right answer, the teacher can move on. Hence, the heart of the method is the combination of probing questions and engaging students in peer instruction.

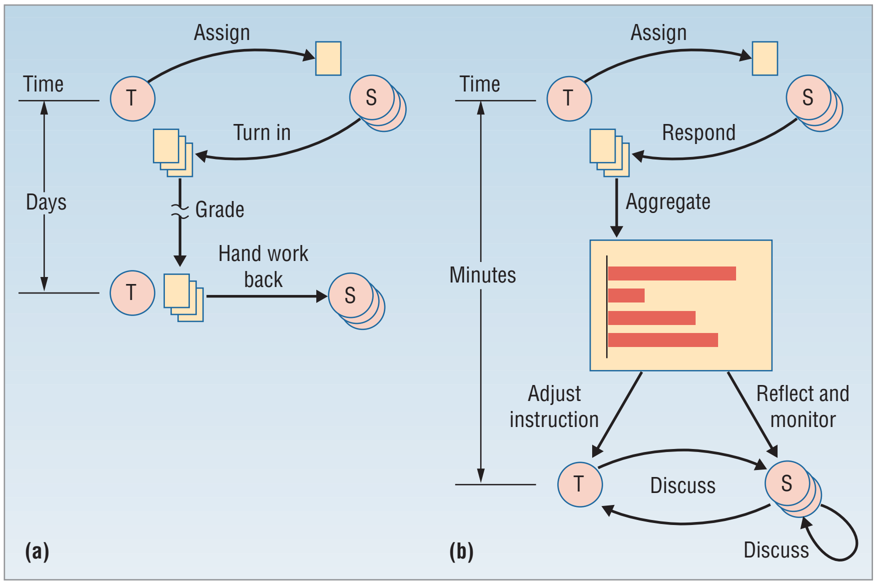


Figure 1. Interacting around student work. (a) Traditional methods require extensive paperwork and can take days to complete; (b) networked classrooms allow a much higher level of interaction and can accomplish an equivalent workload in minutes.

As Figure 1 shows, this method can improve teaching and learning by

- decreasing from days to minutes the time it takes teachers and students to get feedback,
- enabling the teacher to adapt instruction,
- encouraging students to reflect and monitor their own progress, and
- engaging students in arguing for their point of view.

Tablet PCs allow richer interactions than those possible with a clicker. One application that illustrates this, Classroom Presenter (www-cse.ucsd.edu/users/bsimon/pubs/papers/wipte06.pdf), takes as its starting point a prototypical classroom situation in which an instructor presents prepared PowerPoint slides. The instructor can gather students' annotations or sketches on a particular slide and use this collected information to drive further classroom discussion. For example, the instructor can ask students to sketch the next step in the visualization of an algorithm and then discuss the varied possibilities.

Both clickers and Classroom Presenter are powerful classroom innovations, but both presume that the teacher coordinates all classroom interactions explicitly; they do not support coordinated use of the technology among students. Thus, we set out to design a software platform that supports generalized coordination among students and a teacher.

INTRODUCING GROUP SCRIBBLES

Group Scribbles starts from the desire to maximize the power of ink, improvisation, and interactive engagement in a wireless, tablet-based learning environment. improvisation is the central design goal: We intend Group Scribbles to be a platform that supports teachers in inventing and enacting new forms of collaboration and coordination in their classroom without resorting to additional programming. To support improvisation, Group Scribbles offers a powerful metaphor based on familiar physical artifacts from the classroom or office: adhesive notes, bulletin boards, whiteboards, stickers, pens, and markers.[9]

The fundamental unit of expression in Group Scribbles, the Scribble Sheet, is a small square of virtual paper just large enough to express a single thought or concept, whether via a quick sketch or a few words. Scribble Sheets can be posted to public boards, visible to all participants. Multiple sheets can be arranged to express ensemble ideas, such as groupings, chronologies, or hierarchies. A smaller sheet, termed a *label*, can be attached to the larger Scribble Sheets as an annotation. In addition, each participant has

Computer Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

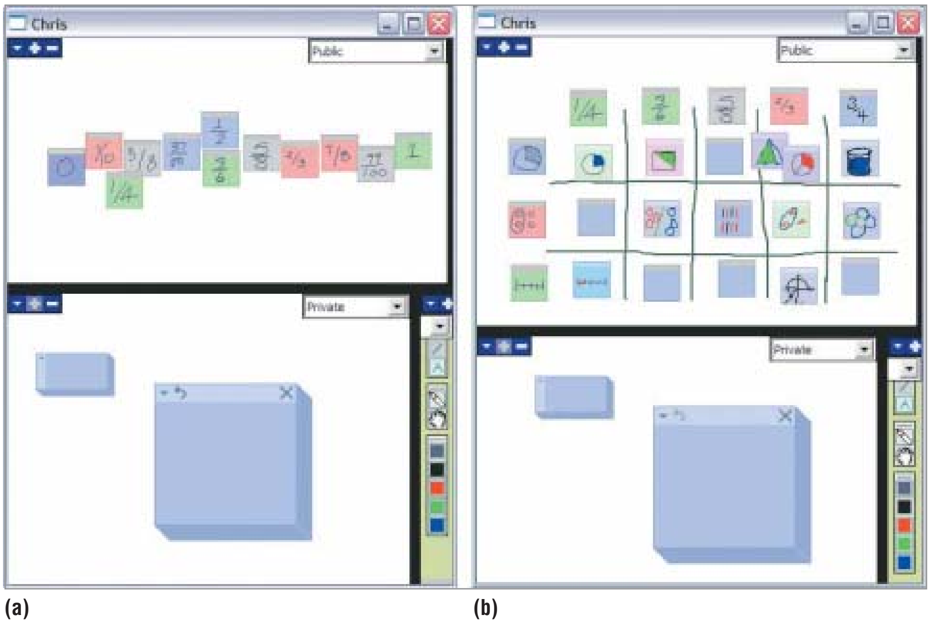


Figure 2. Group Scribbles. This application (a) helps students place their fractions in order, from the public board's left to right side, and (b) helps the teacher organize examples of the different representations into a table's row header and move some interesting fractions to the table's column header.

The Take operation is especially important because it allows coordinated, synchronous activity among students without requiring the teacher to provide detailed centralized instructions.

Group Scribbles, used by researchers across the globe, can be downloaded for free (http://groupscribbles.sri.com). Our close colleagues in Spain, Singapore, and Taiwan continue to tell us about new educational applications for Group Scribbles. Along with our own Group-Scribbles-based activities,[10] these applications offer compelling examples of the ways in which using the Tablet PC with the right software can transform teaching and learning. Two examples show the expressive power of this constellation of affordances in highly interactive teaching.

a private board on which to create and arrange Scribble Sheets. A given classroom instance of Group Scribbles will have one or more named public boards accessible to all users. The screen is divided to show the user's private board in one region and a public board in another.

On the private board, a user finds a Scribble Pad, an endless source of fresh Scribble Sheets. Users can pull sheets off the pad and write or type on them to generate new content. Users can zoom out several levels to help arrange and maintain their Scribbles. When users are ready to publish a Scribble Sheet, they simply drag it onto the public board, where all participants can view it immediately.

On a public board, any user can reposition any Scribble Sheet so that, while individual sheets express individual thoughts, the entire board expresses collective ideas. In this way, users can sort, group, or otherwise arrange Scribbles to express interdependent meaning. A user can take a Scribble Sheet from the public board, bring it onto a private board—for example, for activities calling for exchange or to take a token representing a turn in a sequence—and optionally return it to the public board.

Underlying the Group Scribbles visual metaphor is a "tuple spaces" architecture that supports the three classic operations required by a coordinated, distributed computing system:

- *Write*—dragging a Scribble Sheet from a private to a public board.
- *Read*—viewing the Scribble Sheets on a public board.
- *Take*—dragging a Scribble Sheet from a public to a private board.

INK AND IMPROVISATION

The following lesson, designed for teaching fractions in elementary school mathematics, has structural features that could be reused for many different levels of instruction. In presenting the lesson, we emphasize the role of ink and improvisation as they relate to learning science principles. Imagine this lesson enacted in a wireless classroom where the teacher uses an electronic whiteboard and the students have tablet devices.

To begin, the teacher asks the students to scribble fractions between 0 and 1 and post the Scribble Sheets to the public board. Thus, at the lesson's onset, the students—actively engaged—populate the whiteboard with their handwritten fractions.

Already, the lesson has accumulated many possible branch points. For example, if the teacher finds that some students have used a decimal or percentage, she could invite students to replace those Scribble Sheets with the equivalent fraction expressed as a ratio. The students themselves coordinate this operation through Group Scribbles. Whereas at a physical whiteboard, to avoid chaos the teacher must regulate which students come to the board, in Group Scribbles the Take operation executes atomically. This prevents more than one student from taking the same sheet and thus enables self-coordination.

Let's assume instead that all students understood the teacher's intent and produced fractions like 1/4 or 2/6. The teacher can then continue the lesson by asking stu-

Computer | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

dents to attach a label to the biggest fraction, or the smallest, or the closest to 3/4, and so on. This is like a clicker activity, and the teacher could follow in the style of peer instruction by asking students to convince their neighbor of which fraction is biggest. In the case where fractions like 5/7 and 5/6 are present, knowledge-rich conversations can result: Which is bigger and why?

Figure 2 shows how Group Scribbles also facilitates flexible use of space to organize student work. For example, the teacher can take advantage of this to ask students to place their fractions in order, from the public board's left to right side, as Figure 2a shows. Group Scribbles supports multiple simultaneous drag operations, enabling all students to actively order their fractions at the same time. Students can self-organize the visual space in meaningful ways, a feature not possible with clickers or other tablet applications. Again, there might be opportunities for the teacher to ask students to label fractions placed in the wrong order, rather than pointing these out to the students herself. There might also be opportunities to discuss equivalent fractions, like 1/3 and 2/6.

Let's assume the teacher next wishes to discuss the topic of equivalent fractions. She could observe that 1/3 and 2/6 are equivalent and ask, "Are there any additional equivalent fractions you can think of? Please try to write down at least three and keep them on your private board." The teacher asks students to keep these Scribble Sheets private because if 20 students each submitted three additional fractions, the public board would become cluttered quickly.

The teacher can manage the space in the public board using the following rhetorical style: "Asher, would you share one of your equivalent fractions by dragging it to the public board now? Thanks; 3/9 is also equivalent. How many of you had 3/9? Did anyone have a different equivalent fraction?"

In this way, the teacher can gather a rich diversity of equivalent fractions from many different students, one at a time, without overwhelming the space. Ideally, the teacher would use this as a basis for engaging the students in explaining how they know that fractions are equivalent, which could lead to a discussion of how to prove fractions are equivalent—showing they can be reduced to a common fraction, for example.

Alternatively, the teacher might choose to move into an activity about different representations of fractions. This might start as did the equivalent fraction exercise. The teacher could ask students to generate two or three ways of depicting 2/6, then collect the variety of emergent depictions, such as pie charts, divided bars, or a number line. This leads smoothly to an activity that uses Group Scribbles' Take feature to coordinate a distributed activity.

Figure 2b shows how the teacher can organize examples of the different representations into a table's row header and move some interesting fractions to the table's column header. Blank sheets can be posted to the table's cells.

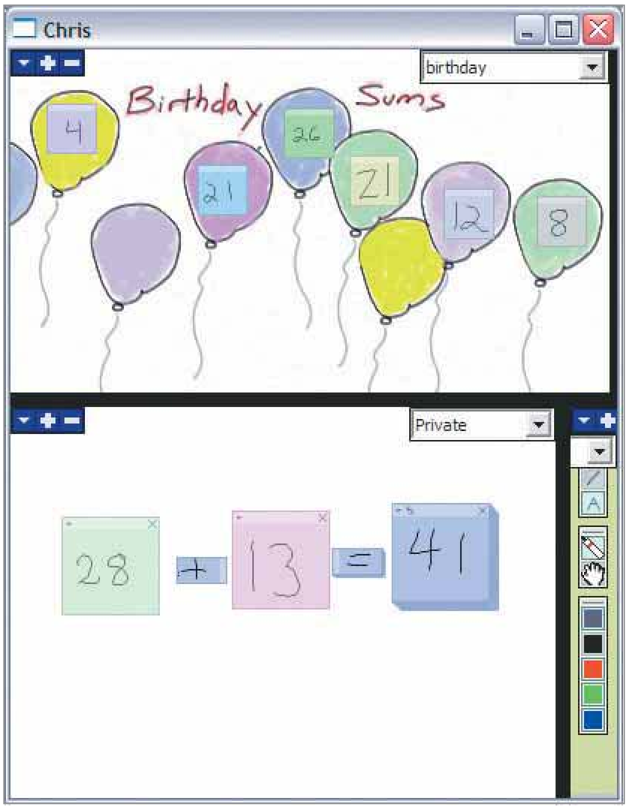


Figure 3. Birthday Sums. This Group-Scribbles-enabled participatory game explores the design of distributed algorithms.

Assume this results in 20 cells and that the class has approximately 20 students. The teacher could now ask students to replace a cell with a Scribble Sheet that depicts the given fraction in the given representation. The teacher does not need to tell individual students which cell they should take; students can self-select and the Take operation will coordinate student activity such that each cell is only taken once. The resulting matrix of fraction representations, shown in Figure 2b, is both a powerful formative assessment of what the class knows and a prop for further discussions.

This lesson exemplifies key learning-science principles. For example, students are actively engaged in creating, organizing, and representing the lesson's mathematical content. Space is used in ways that reflect mathematical knowledge—ordering the fractions from left to right, for example. The lesson is centered on mathematical relationships like ordering, equivalence, and representation. Formative assessment and adaptive instruction are possible throughout the lesson, as both the teachers and students receive rich and continuous feedback on what students know. Finally, at several points in the lesson students act more like a community than they would in a conventional classroom—for example, when they convert their peers' decimal and percentage numbers to conventional fractions or when they jointly complete the table in the last part of the lesson.

Although this lesson focuses on elementary school fraction content, we have found that similar lesson structures are appropriate for university courses. For example, a chemistry professor can ask students to produce Scribble Sheets with different molecules and forms of representation, such as formulas, electron-dot diagrams, or ball-and-stick diagrams. A physics professor can have students sketch a ball at different places in its trajectory and place these in order, from most to least kinetic energy. A Chinese-language professor could use Scribble Sheets to organize a lesson on how "radicals" (components of a character) are put together to form different Chinese characters. Indeed, Scribble Sheets fit Chinese characters very neatly.

> **Group Scribbles can implement a variety of participatory games, including hangman, *Password*, and *Apples-to-Apples*.**

COORDINATING INTERACTIVE ENGAGEMENT

Our experiences with Group Scribbles have led us to conceptualize their key benefit for education as making group learning more playful.[11,12] Ink and improvisation support play by allowing the more informal representations of sketches and gestures and by enabling a smoother flow of activity among participants. However, ink and support for improvisation are not enough: A key feature of play is coordination.

The role of coordination in play can be observed by watching children involved in schoolyard games. Starting from the simple game of catch, the elemental skill students learn is to synchronize their attention and actions with those of a partner to accomplish a shared outcome.

Regrettably, in a classroom setting, the cost of coordination is traditionally a major barrier to engaging in more playful learning experiences. A teacher who must guide and arbitrate all transactions in a game can easily become overburdened.[13] This leads many teachers to quickly revert from playful experiences to standard lecture and discussion formats. Even though collaborative and coordinated activities have high learning value, they have been too hard to implement.

We found that Group Scribbles can radically simplify the coordination of classroom games. It's no longer necessary for the teacher to centrally manage all transactions. The right configuration of boards and sheets, combined with a few verbal instructions to the class, can naturally enable students to self-organize their activities. The use of Group Scribbles in this context builds on prior success with a genre of activity called *participatory simulations*, which employs group play to actively engage students in serious content issues.[14,15] In a participatory simulation, technology brokers the exchange of information among classroom participants, letting the teacher focus on the pedagogical role. Group Scribbles makes it possible for teachers to construct these participatory simulations by themselves and even to invent a participatory simulation on the fly in the classroom.

The learning activity we call *Birthday Sums* offers an example of a Group-Scribbles-enabled participatory game. As Figure 3 shows, although not initially apparent, Birthday Sums is an exploration of the design of distributed algorithms. The instructor begins by asking group members to write their birth date on a Scribble Sheet, which is then posted to a public board. The instructor then challenges the class to come up with methods by which they could add up all the birthday numbers.

After a while, someone in the classroom suggests the distributed algorithm: "Everyone take two numbers, add them, and put the result back. Repeat until only one number is left." Enacting this algorithm as a class is fun, but it also reveals important conceptual issues at the heart of algorithm design. For example, the participants must determine how to

- prevent a deadlock where everyone has taken exactly one number and thus no one can add and no one can get another number;
- know they added all the numbers, or determine that one processor stalled before returning its partial result; and
- make sure they detect and correct errors if one processor adds incorrectly.

Enacting such algorithms as a class generates a palpable excitement, maximizing cognitive engagement in a situation where the class as a whole acts as a distributed machine that tries out variant algorithms.

In the Group Scribbles project, we found the system can implement a variety of participatory games, including hangman, *Password*, and *Apples-to-Apples*. In addition, we have used Group Scribbles to transform single-player games such as crossword puzzles and Sudoku into collaborative games. With Group Scribbles, these games leverage the synthesis of the computer screen's representational power and the wireless network's collaborative capacity.[16] While these games lack the kinesthetic qualities of playground games, they share their highly interactive, negotiable, and appropriable properties.

REFLECTIVE TEACHING WITH TABLETS

Wireless tablet computers can offer new affordances for informal sketches, improvisation, and interactive engagement that take this form factor beyond that possible with prior technologies. Realizing this potential requires navigating design tensions.[13] For example, in the Group Scribbles project, we struggled with the tension between planned and improvised activities and the tension between informal ink and the notations a computer might more easily recognize, such as typed text or mathematical notation.

Another salient tension arises between supporting classrooms as they are today and nurturing classrooms that feature more student-initiated activity and collaboration among students without a teacher at the hub. Learning science research consistently shows that technology can improve student understanding only when students and teachers use it to do meaningful work with course content. Thus, we find it prudent to conceptualize technologies such as Group Scribbles as a potent infrastructure component that must be activated by new teaching practices. These practices can build on lessons from the learning sciences: active engagement, a focus on knowledge construction, feedback leading to adaptive instruction, and group play in a community of learners.

Today's approaches to teaching and learning have moved beyond Socrates' notion of learning as recollection to new metaphors and approaches. Nonetheless, Socrates provides a fine example of a reflective practitioner who engages in scholarship about his teaching practice.

Given the central role of teaching practice in learning outcomes, advances in the use of tablets in education will be driven not primarily by technology features but rather by instructional models that reflective educators develop and share with their peers. Communities that form around platforms such as Classroom Presenter, Group Scribbles, and related applications should provide an excellent forum for such advances.

Acknowledgments

This material is based on work supported by the National Science Foundation under grant no. 0427783. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. We thank the many additional contributors to the Group Scribbles project, including John Brecht, Krista Davis, Zaz Harris, Robin Lin, Chee Kit Looi, and Patti Schank.

References

1. B. Jowett, *The Dialogues of Plato*, 2nd ed., vol. 1, Clarendon Press, 1875.
2. National Research Council, *How People Learn: Brain, Mind, Experience, and School*, National Academy Press, 1999.
3. M. McCullough, *Abstracting Craft: The Practiced Digital Hand*, MIT Press, 1996.
4. J.J. LaViola and R.C. Zeleznik, "MathPad2: A System for the Creation and Exploration of Mathematical Sketches," *Proc. SIGGRAPH 04*, ACM Press, 2004, pp. 432-440.
5. M. Wilkerson, W.G. Griswold, and B. Simon, "Ubiquitous Presenter: Increasing Student Access and Control in a Digital Lecturing Environment," *Proc. SIGCSE Technical Symp. Computer Science Education*, ACM Press, 2005, pp. 116-120.
6. J. Roschelle, W.R. Penuel, and L.A. Abrahamson, "The Networked Classroom," *Educational Leadership*, vol. 61, 2004, pp. 50-54.
7. S.R. Chaudhury et al., "Coordinating Student Learning in the Collaborative Classroom with Interactive Technologies," *Program Abstracts of the 3rd Int'l Society for the Scholarship of Teaching and Learning Conf.*, Int'l Society for the Scholarship of Teaching and Learning, 2006, p. 84.
8. C.H. Crouch and E. Mazur, "Peer Instruction: Ten Years of Experience and Results," *The Physics Teacher*, vol. 69, 2001, pp. 970-977.
9. C. DiGiano, D. Tatar, and K. Kireyev, "Learning from the Post-It: Building Collective Intelligence through Lightweight, Flexible Technology," *Conf. suppl. ACM SIG Conf. Computer-Supported Cooperative Work* (CSCW 2006), ACM Press, 2006, pp. 65-66.
10. J. Brecht et al., "Coordinating Networked Learning Activities with a General-Purpose Interface"; http://groupscribbles.sri.com/publications/GSMLearn.pdf.
11. S. Lin et al., "Learning When Less Is More: 'Bootstrapping' Undergraduate Programmers as Coordination Designers," *Proc. Participatory Design Conf.*, ACM Press, 2006, pp. 133-136.
12. D. Tatar and S. Lin, "Playground Games and the Dissemination of Control in Computing and Learning," *Learning about Learning Technology Design*, C. DiGiano, S. Goldman, and M. Chorost, eds., Lawrence Erlbaum, in press.
13. D. Tatar, "The Design Tensions Framework," *J. Human-Computer Interaction*, in press.
14. T. Moher, "Embedded Phenomena: Supporting Science Learning with Classroom-Sized Distributed Simulations," *Proc. SIGCHI Conf. Human Factors in Computing Systems*, ACM Press, 2005, pp. 691-700.
15. W.M. Stroup, "Learning through Participatory Simulations: Network-Based Design for Systems Learning in Classrooms," *Proc. Computer-Supported Collaborative Learning Conference*, Unext.com, 1999, pp. 667-676.
16. P. Orozco et al., "A Decoupled Architecture for Action-Oriented Coordination and Awareness Management," CSCL/W Frameworks, *Groupware: Design, Implementation and Use*, LNCS, vol. 3198, Springer-Verlag, 2004, pp. 246-261.

Jeremy Roschelle *is the director of the Center for Technology in Learning at SRI International. His research interests include collaborative learning, mathematics education, and scaling up educational interventions. Roschelle received a PhD in education/cognitive science from the University of California, Berkeley. Contact him at jeremy.roschelle@sri.com.*

Deborah Tatar *is an associate professor of computer science and, by courtesy, psychology, at Virginia Tech. Her research interests include augmenting face-to-face learning, conceptualizing complex sociotechnical systems, and technology to support coordination. Tatar received a PhD in psychology from Stanford University. Contact her at tatar@cs.vt.edu.*

S. Raj Chaudhury *is an associate professor of physics at Christopher Newport University. His research interests include learning technologies, scientific visualization, and teacher preparation. Chaudhury received a PhD in physics from the University of California, Los Angeles. Contact him at schaudhury@cnu.edu.*

Yannis Dimitriadis *is an associate professor of signal theory, communications, and telematics engineering at the University of Valladolid, Spain. His research interests include technological supports for learning and work processes, machine learning, and the use of technology in learning scenarios. Dimitriadis received a PhD in telecommunications engineering from the University of Valladolid. Contact him at yannis@tel.uva.es.*

Charles Patton *is a principal scientist at SRI International. His research interests include technology-supported mathematical representations, coordination structures in collaborative learning, and wireless Internet learning devices. Patton received a PhD in mathematics from SUNY Stony Brook. Contact him at charles.patton@sri.com.*

Chris DiGiano *is a senior scientist at SRI International and an adjoint professor at the University of Colorado at Boulder. His research interests include design processes for learning tools, collaborative learning systems, and end-user development. DiGiano received a PhD in computer science from the University of Colorado at Boulder. Contact him at chris.digiano@sri.com.*

Practical Support for ISO 9001 Software Project Documentation: Using IEEE Software Engineering Standards

WILEY
Publishers Since 1807

IEEE
IEEE computer society

www.wiley.com/ieeecs

978-0-471-76867-8 • October 2006
418 pages • Paperback • $89.95
A Wiley-IEEE Computer Society Press

To Order:
1-877-762-2974 North America
+ 44 (0) 1243 779 777 Rest of World

ISO 9001 provides a tried and tested framework for taking a systematic approach to software engineering practices. Readers are provided with examples of over 55 common work products. This in-depth reference expedites the design and development of the documentation required in support of ISO 9001 quality activities. Also available:

- Practical Support for CMMI© - SW Software Project Documentation: Using IEEE Software Engineering Standards
- Jumpstart CMM©/CMMI© Software Process Improvements: Using IEEE Software Engineering Standards

15 % off for CS Members

COVER FEATURE

Handwriting Recognition: Tablet PC Text Input

James A. Pittman
Microsoft

To support a wide range of writing styles and poorly formed cursive script, the Tablet PC input panel uses a time-delay neural network working with a lexicon. High-end versions of Microsoft's Vista now include Tablet PC software, with an improved recognizer that supports both personalization and error reporting.

Data-entry problems persist for portable electronic devices. PDA keypads, cell-phone-size qwerty keyboards, and onscreen keyboards are all difficult to use, and full-size plug-in keyboards are hard to maneuver in constrained spaces such as airplanes. In specific situations, machine-recognized handwriting serves as a useful alternative for entering small volumes of text.

HANDWRITING-RECOGNITION CHALLENGES

Practical handwriting recognition isn't easy, however. Not only must handwriting-recognition systems handle many different shapes and styles for each letter, but humans also rarely write technically correct letter shapes. Neat and correct handwriting takes time. Most people reduce their adherence to letters' defined shapes to speed up their writing, producing *sloppy script*. Typically, writers maximize their throughput speed by dropping their readability to an estimated minimum, adjusted occasionally as a result of negative feedback.

Employing context

Humans can read sloppy handwriting because of context, both *within-word context* (most letter sequences aren't words) and *between-word context* (most word sequences aren't meaningful). To support the sloppy script needed to maintain reasonable throughput, the handwriting recognizer must employ the same context that human readers employ. The use of between-word context is still largely a research topic, but the Tablet PC recognizer makes heavy use of within-word context, primarily in the form of a lexicon.

However, employing a lexicon creates its own difficulties. In addition to providing thorough coverage of the language, a system must also cover each specific user's jargon and slang, being careful about potentially offensive words. This means not only sexually taboo words, but racial and ethnic slurs, and in some cultures, blasphemy.

Pen-based computing

Microsoft has pursued pen-based computing since the early 1990s. It released Windows for Pen Computing in 1992, and Pen Services was a component of Windows 95, released in 1995. While these products ultimately failed in the marketplace, a small group at Microsoft continued working on recognition technology. That group shipped recognizers for the East Asian market, focusing mostly on Pocket PCs running Windows CE. In 1997, the group expanded into handling cursive English and then added recognizers for cursive French and German.

Microsoft Chairman Bill Gates continued to believe in the dream of pen-based computing, predicting that technical advances and cost reductions would eventually create a successful product. In 2000, the company formed the Tablet PC group to design a pen-centric version of the next OS, Windows XP. The recognition team moved into the new group, and, with a new burst of

0018-9162/07/$25.00 © 2007 IEEE

Figure 1. Tablet PC input panel. The input panel lets users write directly on the Tablet PC.

energy (and funding), expanded its work on cursive writing in a variety of western languages.

Tablet PC software is no longer separate, but is now included in most high-end versions of Vista, which Microsoft released in January 2007. The Vista recognizer is vastly improved, and it supports both personalization and error reporting.

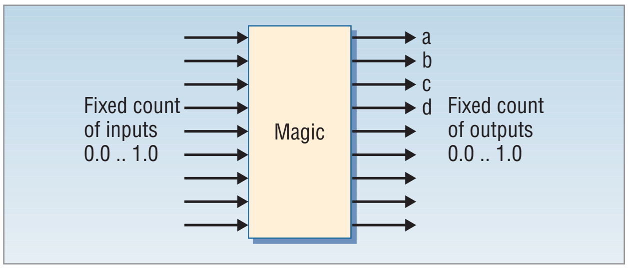


Figure 2. Outsider's view. This view of a neural network shows the information fed into and obtained from the network.

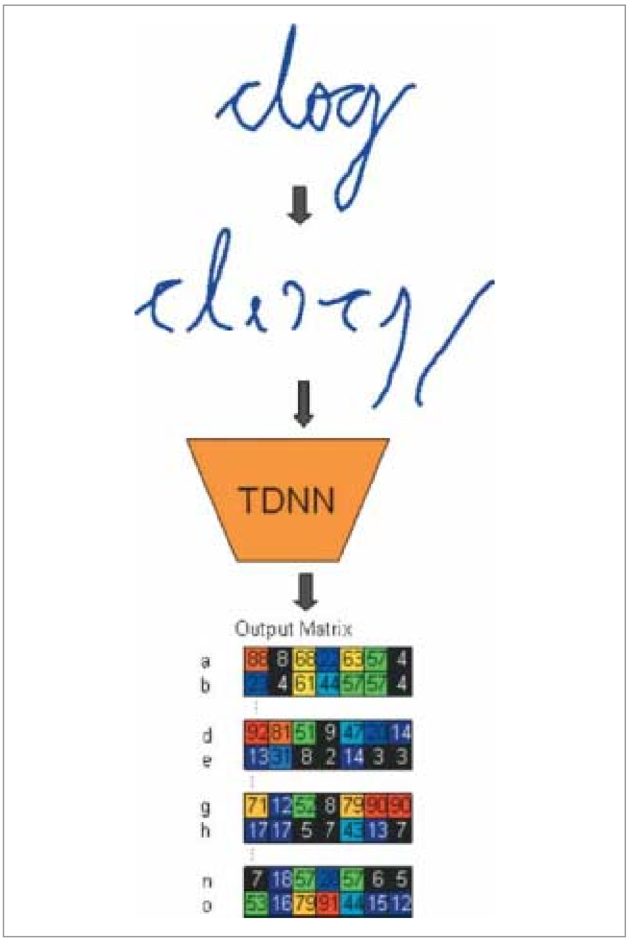


Figure 3. Input and output of the time-delayed neural network. A TDNN repeats the sets of inputs and outputs, along with the internal mechanisms.

Tablet PC input panel

Most users experience the handwriting recognizer via a special handwriting window called the *Tablet PC input panel*. As Figure 1 shows, the TIP will appear anytime a pen is brought near the tablet's surface.

But the handwriting recognizer isn't restricted just to the TIP; users can call it up from any software. Application developers can obtain a free download of the Vista software development kit at http://microsoft.com/downloads/details.aspx?FamilyId=7614FE22-8A64-4DFB-AA0C-DB53035F40A0&displaylang=en. Once the kit is installed, the documentation is available at Start/All Programs/Microsoft Windows SDK/Windows SDK Documentation. Tablet PC code samples are in %ProgramFiles%\MicrosoftSDKs\Windows\v6.0\Samples\TabletPC.

NEURAL NETWORKS

To support such a wide range of writing styles, and in particular to support poorly formed cursive script, we employ a large neural network trained on a very large training set. The training set contains ink samples from thousands of people with a diverse range of writing styles.

For our purposes, we can explain a neural network as a view from the outside—a description of what information we feed into and obtain from the network. Readers interested in how a neural network works will find hundreds of books written on the topic. A reasonable starting point might be the Wikipedia entries on artificial neural networks (http://en.wikipedia.org/wiki/Artificial_neural_network) and parallel distributed processing (http://en.wikipedia.org/wiki/Parallel_distributed_processing).

Figure 2 presents an outsider's view of a neural network. The network has a fixed number of inputs and outputs. Each input or output is a simple floating-point number, in our case ranging from a fixed minimum (0.0) to a fixed maximum (1.0).

The inputs are features (measurements) computed from the ink itself. The input features' exact composition is a trade secret, but generally the measurements are of the direction and curvature of the ink trace, along with various measurements of size. The outputs are estimates of probability for each letter, digit, or character the recognizer supports.

Computer Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

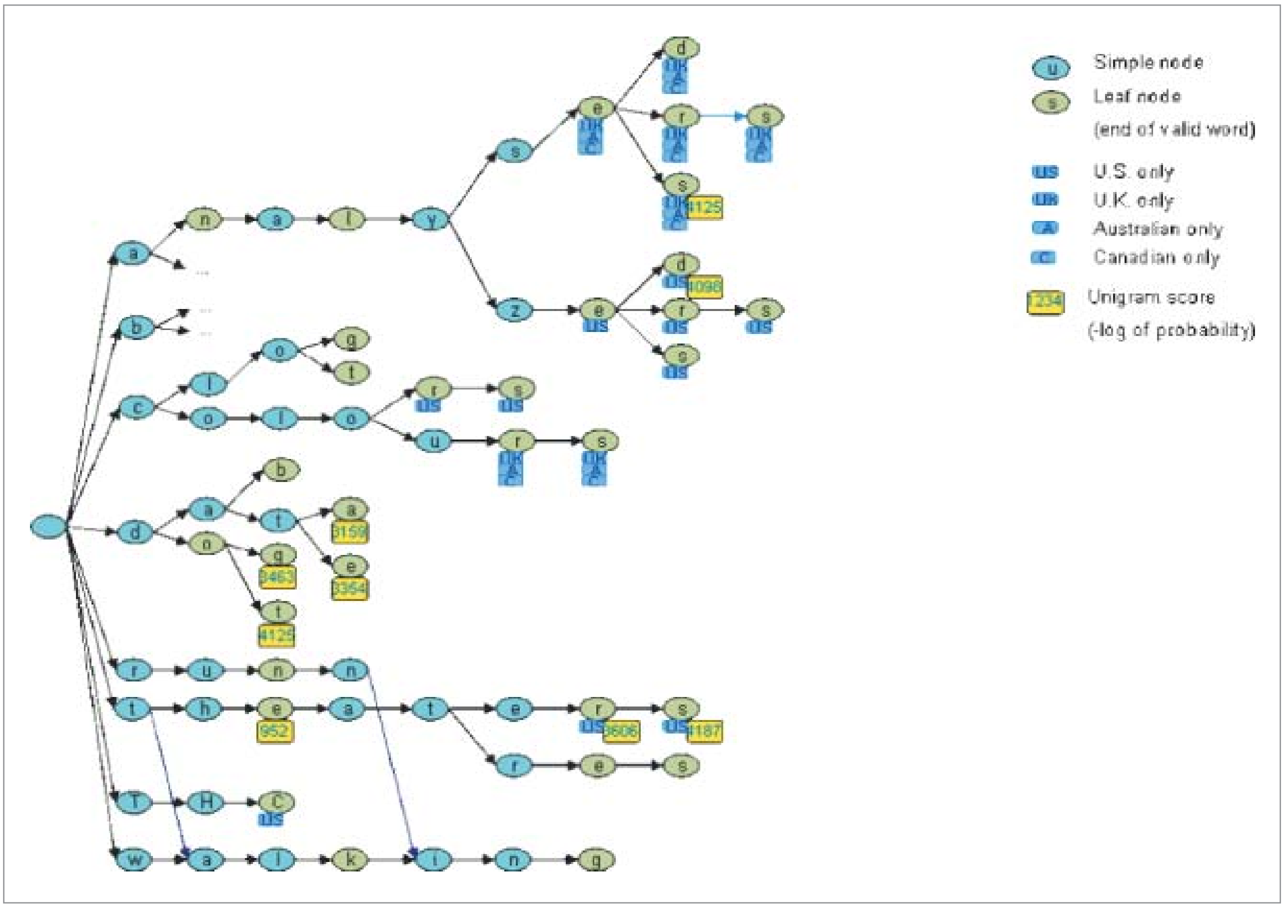


Figure 4. Lexicon stored as a trie. The majority of language-model context comes in the form of a lexicon, a simple list of the allowed spellings in the language.

The neural network that Figure 2 depicts will handle isolated character recognition, but it won't handle recognition of connected letters in cursive script. For this, we use a variation called the *time-delay neural network*. A TDNN repeats the sets of inputs and outputs, along with the internal mechanisms. A TDNN has a *fan-in factor*. In our case, any one set of outputs sees connections from a window of five sets of inputs. This allows the TDNN to output a set of estimates of probabilities of letters, digits, and characters for each segment of ink, based on its observation of that segment and the two segments just before and just after.

A simple rule segments the ink: The software cuts the ink into separate segments anytime it drops downward in the Y dimension and then reverses direction back upward. Also, it cuts the ink anywhere the writer lifts the pen. Figure 3 shows the ink before and after segmentation, the TDNN, and the output, which consists of a vector of probability estimates per segment, to form a matrix.

LANGUAGE MODEL

If the TDNN were forced to recognize sloppy cursive handwriting by itself, its performance would be weak at best. When cursive writing is sloppy, letter shape is only a weak predictor of the intended letter. To approach human recognition accuracy, we must employ the extra information that human readers employ: the *language-model context*.

The majority of this language-model context comes in the form of a lexicon, a simple list of the allowed spellings (character sequences) in the language. This is the same lexicon a spellchecker employs. Figure 4 depicts the lexicon, organized in the form of a *trie*. In our case, the nodes contain a Boolean flag indicating whether this letter is the end of a word. Optionally, some such end-of-word nodes also contain a word unigram probability (stored in the form of a negative log), or a set of dialog bits indicate that the spelling is only valid in some specific dialects.

The lexicon is combined with the output of the TDNN using a beam search, an approach that most commercial speech-recognition systems use. Figure 5 shows how the ink is featurized at the top of the figure and fed through the TDNN. The TDNN outputs consist of a matrix, where each column represents one ink segment and each row represents one letter, digit, or character.

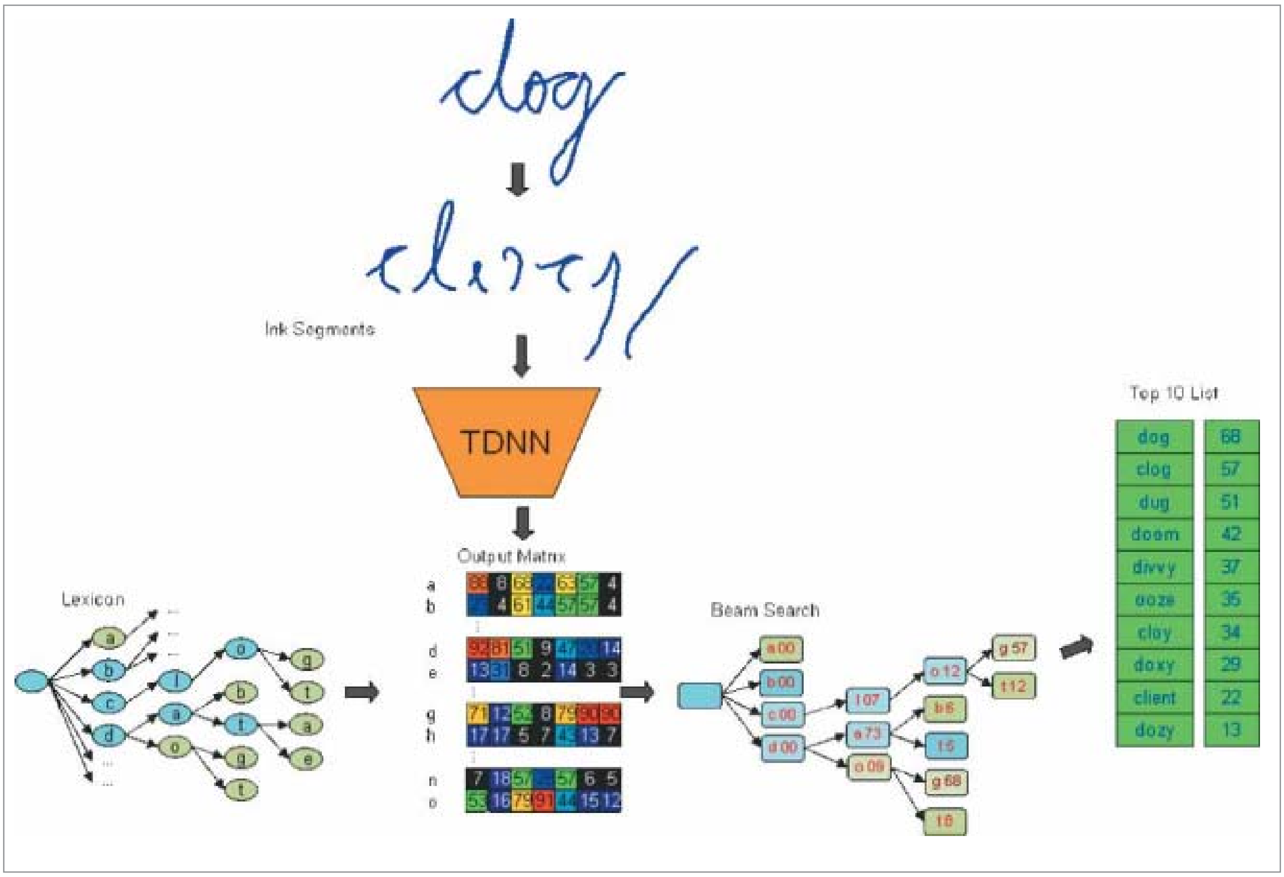


Figure 5. Beam search. The beam search moves through the TDNN output matrix, column by column, from left to right.

Figure 5 illustrates how the beam search moves through the TDNN output matrix, column by column, from left to right. At each column, it asks the lexicon on the left what letters might be next and looks up the probability estimates for those letters in the matrix. It builds a copy of the lexicon trie (shown on the right side in Figure 5), with the scores in the nodes of this copy trie. At any one time, the scores in that trie are the cumulative scores of each character and all characters up to that character, for each possible word in the lexicon.

For instance, look at the two nodes representing the letter "g" in that trie. One is in the last column, and one is in the next-to-last column. The score in the "g" node in the last column represents the cumulative score of that letter "g" and all the letters in the parent nodes, from the root down to this node, representing the word "clog." The score in the "g" node in the next-to-last column represents the cumulative score of the letter "g" and all the letters in its parent nodes, representing the word "dog."

When a parent node produces a child node, the score in the child node is computed from the current score in the parent node. But after that, they operate somewhat independently, even competing. Since we can't know beforehand how many ink segments any letter might occupy, we must consider the possibility that this child letter hasn't started and that we're still in a column (a segment) representing a part of the parent letter. So the scores in a parent node don't represent a component of the child's score. Instead, they represent a competitor of the child.

At any given time, the nodes in the copy trie contain scores as of the current column. Early on, the scores higher in the trie will be stronger, representing shorter words. As we progress left to right in the output matrix, the scores further down in the trie will become stronger, and the scores of previously strong letters will weaken, as we see too many segments for such short words. Once we've completed the last column in the matrix, the trie contains our final hypotheses about what the word might be. At this point, it's simply a matter of walking the trie to find the 10 best-scoring words.

PERSONALIZATION

The handwriting recognizer can now train on a user's handwriting samples to adapt to that user's writing style. The training that's done is the same training that happens before Microsoft ships the product. The only difference is that the customer, rather than thousands of people, contributes the training data.

There are two forms of personalization training. In *explicit training*, the user decides to personalize and runs

Computer | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

a training wizard that prompts for specific text. The user writes the requested phrases to provide sample ink. In *implicit training*, the user enables the feature and then continues to use the product in the usual manner. Ink that's written in the writing window is stored away for future training.

Explicit training gives the user a chance to confirm that the ink is what the label says it is. But the volume is low because of the tediousness of copying phrases. Implicit training has the advantage of eventually amassing a much greater supply of ink samples for a more representative sample of this customer's handwriting style. But the ink hasn't been checked so that the labels are correct. The system protects itself via an internal confidence measure and by a pipeline of quarantined stores. Samples move up in status depending on their confidence, and, as the system retrains the recognizer on higher-confidence ink, it gradually adapts to the user's style and is then used to rescore the confidence metric on the ink samples.

Shipping a commercial product containing a lexicon leads to new and somewhat unexpected issues and responsibilities.

ERROR REPORTING

The Vista release has a new feature called "report handwriting recognition errors." After correcting a word in the TIP, the system enables a menu item that allows the user to send the ink sample to the Microsoft group that developed the recognizer. Users can specify what they believe the recognition result should have been.

Optionally, there's a "comment" field that supports venting frustration. This error-reporting mechanism is meant to focus the handwriting-recognition team on those specific errors that are causing the most pain in the user community. The feature is implemented as an extension to Microsoft's original Dr. Watson error-reporting system, which was designed for reporting operating system and application crashes.

OFFENSIVE TERMS

Shipping a commercial product containing a lexicon leads to new and somewhat unexpected issues and responsibilities. One issue that has required significant thought and debate is offensive words and terms. The customer community views some words as patently offensive. Since recognition accuracy isn't 100 percent, it's easy to imagine that someone who isn't intending to write an offensive term might suddenly find that the ink has been misrecognized as one such term.

For instance, my own letter "p" is somewhat idiosyncratic, containing an (apparently) overly tall spike at the top of the letter relative to the majority of writers in the US. This letter is often misrecognized as the letter "f." It's understandable that if I were employed as a sports writer covering hockey, every time I used the word "puck" there would be a chance a profanity might appear in the output. Such a situation would create an unreasonable proofreading burden to avoid an embarrassing mistake.

SOLUTIONS TO THE PROBLEM

The Microsoft team obtained its lexicon from the spellchecker, which unfortunately contains many offensive terms. Without their presence in the lexicon, the spellchecker marks every incidence of any offensive term as misspelled, which could lead to ridicule and also becomes a form of censorship. But if we add the words to the spellchecker's lexicon, when someone misspells some similar word, there's a chance that one of the offensive words might show up in a list of suggested corrections, possibly embarrassing some customers. The spell-checker avoids this problem with the addition of a single bit per word; with this "restricted" bit on, the list contains the offensive words and they're no longer flagged as misspelled, but the list of suggested corrections won't include them.

Life isn't so easy for the handwriting recognizer. It doesn't perform two jobs; it just outputs possible recognitions into a top-10 list for display in the writing window. If it marks a word as restricted, the handwriting recognizer can't use it, because using it at any time constitutes offering it as a recognition. Thus, the only (admittedly weak) solution is to remove the word from the lexicon. The Microsoft team filtered out a selection of roughly 1,000 words (the count varies by language) covering sexual and scatological terms, racial and ethnic slurs, and blasphemy.

Because the lexicon supports an out-of-dictionary system—to cover such things as part numbers, foreign names, and so on—users can still write offensive terms, but they must handprint them very neatly. All that's actually removed is support for sloppy cursive writing of offensive terms.

In the spirit of avoiding censorship, the user can add any word to the user dictionary, a user-specific list of additional terms for the lexicon. Customers offended by the lack of offensive terms can add their favorite obscenities back into the lexicon.

NEW LANGUAGES

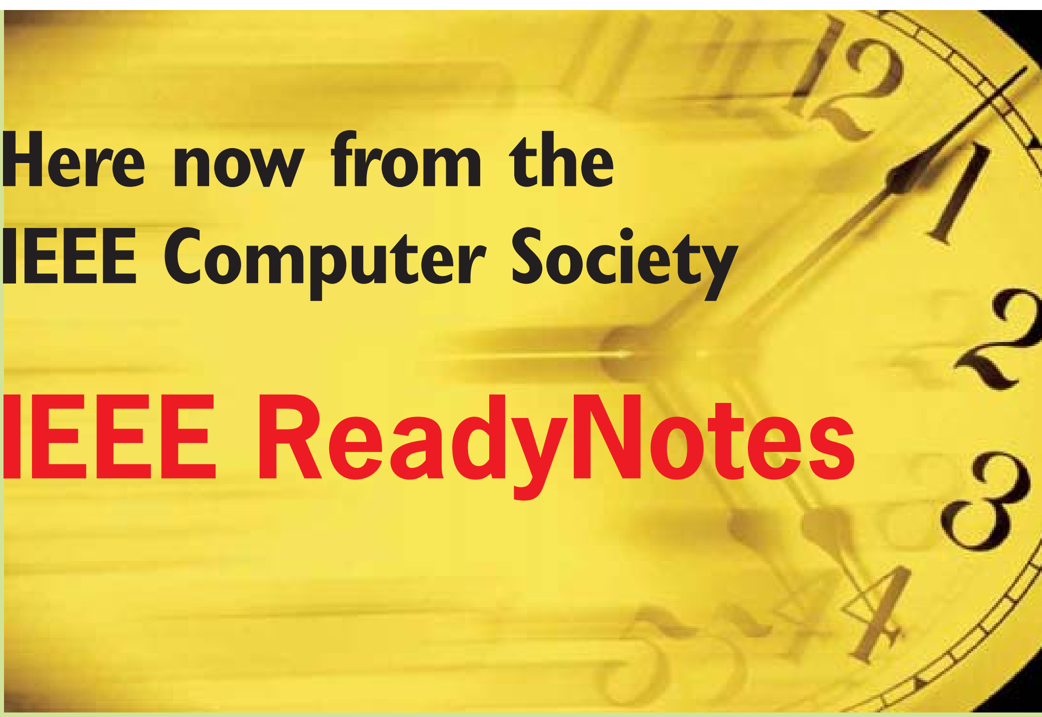
The current Tablet PC handwriting recognizers support Brazilian Portuguese, Chinese (with separate support for simplified and traditional Chinese), Dutch, English (with separate support for US, UK, Canadian, and Australian versions), French, German, Italian, Japanese, Korean, and Spanish.

The team is continually working on new languages. As of June 2007, work continues on Catalan, Croatian, Czech, Danish, Finnish, Norwegian, Polish, Portuguese (for Portugal), Romanian, Russian, Serbian (with separate support for Latin and Cyrillic scripts), and Swedish. Shipping occurs when specific quality criteria, rather than schedule deadlines, are met.

The team has started research on three new languages outside the orthographic families of previous products: Arabic, Hebrew, and Hindi. Each of these new scripts brings with it new challenges not seen in the Cyrillic, East Asian, or Latin orthographies.

Pen-based computing is here now. Cheaper and better batteries, CPUs, displays, memory, handwriting-recognition software, and note-taking applications have combined to make Tablet PCs practical and cost-effective. American English users with typical sloppy cursive or semicursive script tend to see amazing recognition accuracy, especially once they acclimate to writing with a plastic pen on a plastic screen. The Tablet PC is a laptop that shifts from mobility between uses to mobility while in use. It's becoming the laptop of choice for many students, professors, doctors, pilots, police, military personnel, inspectors, construction workers, delivery drivers, and workers in traditional office cubicles. ■

James A. Pittman *is a software development lead with Microsoft. His research interests are in neural-network pattern recognition, human interfaces, and handwriting recognition. He received a PhD in industrial engineering and operations research from Virginia Tech. Contact him at jay.pittman@microsoft.com.*



Looking for accessible tutorials on software development, project management, and emerging technologies? Then have a look at ReadyNotes, another new product from the IEEE Computer Society.

ReadyNotes are guidebooks that serve as quick-start references for busy computing professionals.

Available as immediately downloadable PDFs (with a credit card purchase), ReadyNotes sell for $19 or less.
www.computer.org/ReadyNotes

IEEE | IEEE computer society

IEEE computer society

PURPOSE: The IEEE Computer Society is the world's largest association of computing professionals and is the leading provider of technical information in the field.

MEMBERSHIP: Members receive the monthly magazine *Computer*, discounts, and opportunities to serve (all activities are led by volunteer members). Membership is open to all IEEE members, affiliate society members, and others interested in the computer field.

COMPUTER SOCIETY WEB SITE: www.computer.org

OMBUDSMAN: To check membership status or report a change of address, call the IEEE Member Services toll-free number, +1 800 678 4333 (US) or +1 732 981 0060 (international). Direct all other Computer Society-related questions—magazine delivery or unresolved complaints—to help@computer.org.

CHAPTERS: Regular and student chapters worldwide provide the opportunity to interact with colleagues, hear technical experts, and serve the local professional community.

AVAILABLE INFORMATION: To obtain more information on any of the following, contact Customer Service at +1 714 821 8380 or +1 800 272 6657:

- Membership applications
- Publications catalog
- Draft standards and order forms
- Technical committee list
- Technical committee application
- Chapter start-up procedures
- Student scholarship information
- Volunteer leaders/staff directory
- IEEE senior member grade application (requires 10 years practice and significant performance in five of those 10)

PUBLICATIONS AND ACTIVITIES

Computer. The flagship publication of the IEEE Computer Society, *Computer*, publishes peer-reviewed technical content that covers all aspects of computer science, computer engineering, technology, and applications.

Periodicals. The society publishes 14 magazines, 9 transactions, and one letters. Refer to membership application or request information as noted above.

Conference Proceedings & Books. Conference Publishing Services publishes more than 175 titles every year. CS Press publishes books in partnership with John Wiley & Sons.

Standards Working Groups. More than 150 groups produce IEEE standards used throughout the world.

Technical Committees. TCs provide professional interaction in over 45 technical areas and directly influence computer engineering conferences and publications.

Conferences/Education. The society holds about 200 conferences each year and sponsors many educational activities, including computing science accreditation and certification.

Next Board Meeting: 9 Nov. 2007, Cancún, Mexico

IEEE

EXECUTIVE COMMITTEE

President: Michael R. Williams*
President-Elect: Rangachar Kasturi*
Past President: Deborah M. Cooper*
VP, Conferences and Tutorials: Susan K. (Kathy) Land (1ST VP)*
VP, Electronic Products and Services: Sorel Reisman (2ND VP)*
VP, Chapters Activities: Antonio Doria*
VP, Educational Activities: Stephen B. Seidman†
VP, Publications: Jon G. Rokne†
VP, Standards Activities: John Walz†
VP, Technical Activities: Stephanie M. White*
Secretary: Christina M. Schober*
Treasurer: Michel Israel†
2006–2007 IEEE Division V Director: Oscar N. Garcia†
2007–2008 IEEE Division VIII Director: Thomas W. Williams†
2007 IEEE Division V Director-Elect: Deborah M. Cooper*
***Computer* Editor in Chief:** Carl K. Chang†
Executive Director: Angela R. Burgess†
** voting member of the Board of Governors* *† nonvoting member of the Board of Governors*

BOARD OF GOVERNORS

Term Expiring 2007: Jean M. Bacon, George V. Cybenko, Antonio Doria, Richard A. Kemmerer, Itaru Mimura, Brian M. O'Connell, Christina M. Schober
Term Expiring 2008: Richard H. Eckhouse, James D. Isaak, James W. Moore, Gary McGraw, Robert H. Sloan, Makoto Takizawa, Stephanie M. White
Term Expiring 2009: Van L. Eden, Robert Dupuis, Frank E. Ferrante, Roger U. Fujii, Ann Q. Gates, Juan E. Gilbert, Don F. Shafer

EXECUTIVE STAFF

Executive Director: Angela R. Burgess
Associate Executive Director: Anne Marie Kelly
Associate Publisher: Dick Price
Director, Administration: Violet S. Doan
Director, Finance and Accounting: John Miller

COMPUTER SOCIETY OFFICES

Washington Office. 1730 Massachusetts Ave. NW, Washington, DC 20036-1992
Phone: +1 202 371 0101 • Fax: +1 202 728 9614
Email: hq.ofc@computer.org

Los Alamitos Office. 10662 Los Vaqueros Circle, Los Alamitos, CA 90720-1314
Phone: +1 714 821 8380
Email: help@computer.org
Membership and Publication Orders:
Phone: +1 800 272 6657 • Fax: +1 714 821 4641
Email: help@computer.org

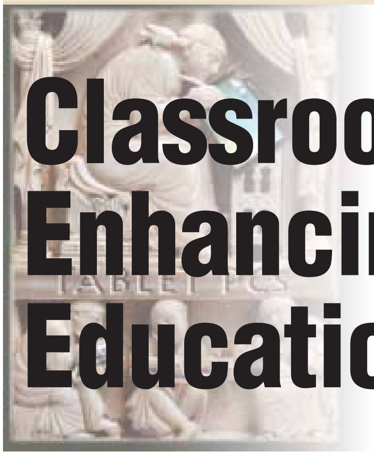
Asia/Pacific Office. Watanabe Building, 1-4-2 Minami-Aoyama, Minato-ku, Tokyo 107-0062, Japan
Phone: +81 3 3408 3118 • Fax: +81 3 3408 3553
Email: tokyo.ofc@computer.org

IEEE OFFICERS

President: Leah H. Jamieson
President-Elect: Lewis Terman
Past President: Michael R. Lightner
Executive Director & COO: Jeffry W. Raynes
Secretary: Celia Desmond
Treasurer: David Green
VP, Educational Activities: Moshe Kam
VP, Publication Services and Products: John Baillieul
VP, Regional Activities: Pedro Ray
President, Standards Association: George W. Arnold
VP, Technical Activities: Peter Staecker
IEEE Division V Director: Oscar N. Garcia
IEEE Division VIII Director: Thomas W. Williams
President, IEEE-USA: John W. Meredith, P.E.

revised 25 June 2007

COVER FEATURE



Classroom Presenter: Enhancing Interactive Education with Digital Ink

Richard Anderson, Ruth Anderson, Peter Davis, Natalie Linnell, Craig Prince, Valentin Razmov, and Fred Videon
University of Washington

Classroom Presenter is a Tablet PC-based interaction system that supports the sharing of digital ink on slides between instructors and students. Initial deployments show that using the technology can achieve a wide range of educational goals and foster a more participatory classroom environment.

As classroom environments become ever more device-rich, higher education faces a dilemma: Will myriad distractions and external demands for communication cause students to lose attention, or can the devices be harnessed to improve learning?

We believe that technology can drive major positive changes in the classroom and address problems associated with traditional lecture-based pedagogy.[1] In our vision, students will have access to a heterogeneous collection of networked devices that can be used for a wide range of classroom applications including note taking, interaction with presentation materials, simulations, in-class communication, and accessing outside resources.

Our work in this space focuses on the deployment of networked Tablet PCs to facilitate student interaction with electronic lecture slides. The Tablet PCs are used for communication between students and instructor to enhance student engagement and active learning.

CLASSROOM PRESENTER

The widespread adoption of electronically projected slides has caused a technological and pedagogical revolution in the classroom.

On the positive side, slides provide structure for class sessions and let instructors display high-quality illustrations and examples to students. On the other hand, this structure conflicts with many educational objectives. Slide-based pedagogy relies on fixed, linear content and is geared toward one-to-many communication with limited feedback from the audience. However, good teaching requires awareness of students' level of understanding, dynamic adjustment of delivery and content, and the active engagement of students in their learning.[2]

In our work in the classroom, we use the slide-based lecture as a baseline and augment the technology to provide instructors with tools to

- enable the flexible delivery of lecture content;
- lower barriers to contributing so that more students feel comfortable participating in class;
- promote peer learning by integrating student work into classroom discussions;
- increase student engagement and understanding via in-class student activities; and
- make instructors more aware of their students' level of understanding.

To support these goals, we have developed and deployed Classroom Presenter (http://cs.washington.edu/education/dl/presenter), a distributed Tablet PC-based classroom interaction system.[3-5] As Figure 1 shows, Classroom Presenter supports the sharing of digital ink on electronic slides among instructor, student, and public displays. The user interface varies by device—on a public display, only the slide shows, without any additional controls.

Published by the IEEE Computer Society
0018-9162/07/$25.00 © 2007 IEEE

In a basic usage scenario, the instructor writes on a slide, and the ink annotations appear simultaneously on a public display. The ability to augment slides in real time to highlight or clarify a point provides dynamicity to a traditional slide-based presentation.

More importantly, in the case where students also have Tablet PCs, Classroom Presenter supports bidirectional sharing of data between instructor and student devices. In addition to receiving the instructor's slides and ink in real time, the students can write on slides and send them back to the instructor anonymously. The instructor can then show a selection of the submitted student work on a public display.

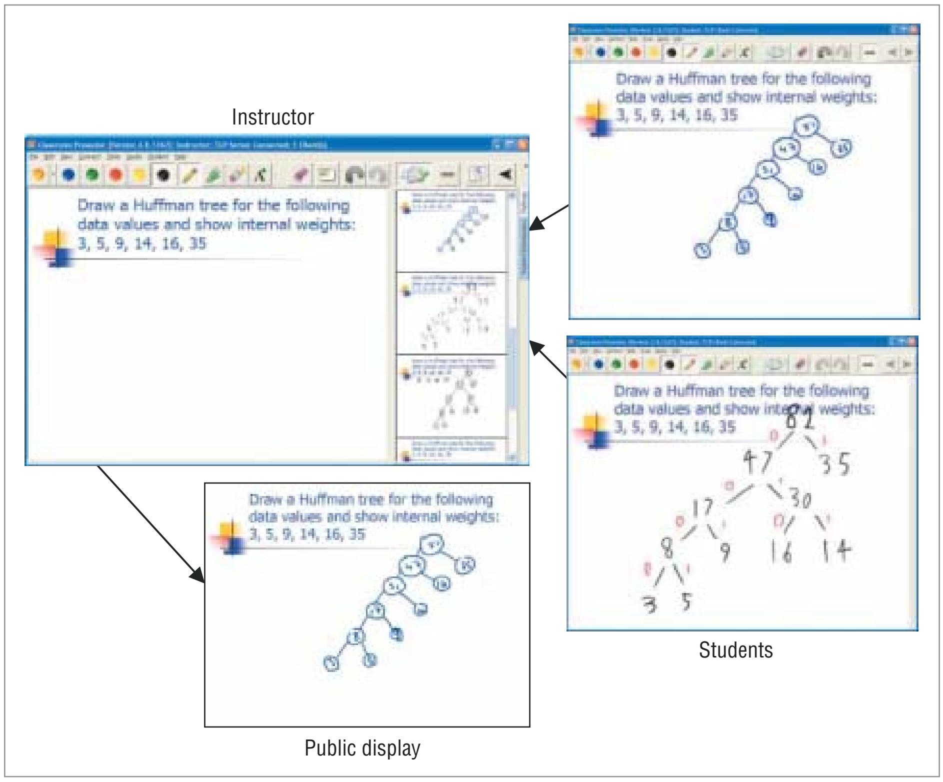


Figure 1. Classroom Presenter. The system runs on instructor, student, and public displays, each with its own user interface. In the scenario shown here, the instructor presents a slide with an activity. The students write solutions to the activity on their Tablet PCs and submit their answers to the instructor. The instructor can preview the student solutions in a film strip (shown on the right of the instructor view) and then selectively show student answers on the public display for class discussion.

This student submissions scenario is central to the pedagogy we have developed around the use of interacting devices. In the scenario shown in Figure 1, the instructor has created a slide-based lecture with activities on some slides. Upon reaching a slide, the instructor asks students to write answers in digital ink on the slide and return it. The instructor can then quickly and privately preview the submissions and selectively show some on a public display.

This lets the instructor introduce diverse ideas, show novel solutions, and discuss misconceptions in student answers. Using a public display creates a focus of attention and provides a mechanism for integrating student work into the lecture discussion, a powerful aspect of the student submission process.

CLASSROOM EXAMPLE

A student submission activity from an actual lecture on Huffman coding illustrates a typical use of Classroom Presenter.

The instructor began by introducing the Huffman tree algorithm, and then demonstrated the algorithm by sketching on a slide a Huffman tree for a given set of data values. Next, she advanced to a slide with an example intended for the students to work through. The instructor felt that having students work through an example was a better use of class time because it would promote understanding of the algorithm more effectively than additional lecturing.

A secondary goal in having students do the activity was to allow the instructor to get feedback on whether students had understood the algorithm. Depending on the results of the activity, the instructor planned to either address students' confusion or move to the next topic.

As soon as the instructor displayed the activity slide, shown in Figure 2a, the classroom tone changed. Working in pairs, students shifted their focus to the tablet in front of them and began talking to their partners about the problem and possible solutions. The instructor moved around the room, occasionally answering questions or clarifying details of the activity and getting an idea of the types of responses the students would submit.

As students completed the problem, conversations moved to other topics, and a few students began to doodle. From previous experience, the instructor knew that it was time to ask students to submit their answers and move on to class-wide discussion.

Before displaying responses to the class, the instructor quickly previewed the submissions on her tablet to get a feel for students' level of understanding. She had

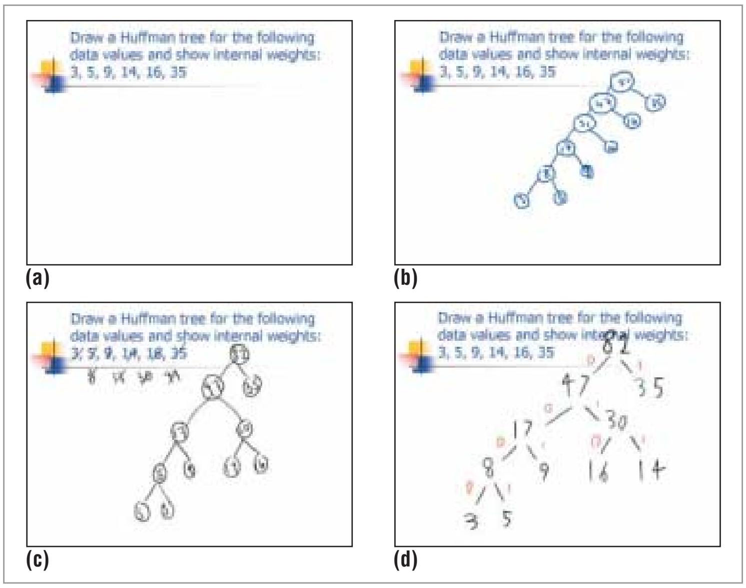


Figure 2. Typical use of Classroom Presenter. (a) Original activity slide. (b) Incorrect student submission. (c) First correct student submission. (d) Second correct student submission.

intentionally designed this activity so that it would be easy to identify correct solutions visually.

Following one of her usual patterns, the instructor first briefly displayed several responses to the class. This allowed some students to see their own work displayed and gave the entire class a chance to see the types of answers their peers had submitted.

Then the instructor chose to focus on one submitted solution, shown in Figure 2b, that demonstrated a specific, common misunderstanding about the algorithm. Since the solution was displayed anonymously, she could discuss the error without embarrassing the students who had submitted it.

Next, the instructor chose a correct solution, shown in Figure 2c, to display and discuss. This example was particularly appropriate because it demonstrated the process students had used to solve the problem (the crossed-out numbers) and thereby enabled the instructor to draw attention to that process without taking the time to re-create it herself.

The instructor then displayed another solution, shown in Figure 2d, and asked the class to help her explain why it was also correct despite the fact that the nodes with values 14 and 16 were reversed. Displaying the second solution also naturally led to a discussion of how to construct code words from a particular Huffman tree by labeling the edges with zeros and ones (as this pair of students had done in their solution even though it was not part of the question).

Satisfied that she had made the relevant points and confident that most students understood the algorithm, the instructor moved on to the next group of slides in the lecture to begin a new topic.

CLASSROOM EXPERIENCES

To explore our vision of interacting devices in classrooms of the future, we have deployed Classroom Presenter extensively in various courses at the University of Washington. We have used the system primarily in computer science courses and in environmental science courses ranging in size from 8 to 60 students. Instructors at other universities, including MIT, Virginia Tech, the University of Virginia, and the University of Massachusetts, have also used Classroom Presenter, as have several high schools and community colleges.

Ubiquitous Presenter,[6] an extension of Classroom Presenter with similar capabilities developed at the University of California, San Diego, has been used in numerous deployments as well.

Our experiences have taught us many lessons about the use of student devices in the classroom and about digital ink as a medium for communication between students and instructor. While we have focused on science courses, we expect many aspects of the pedagogy instructors have developed around our system, the advantages gained from the flexibility and expressiveness of digital ink, and the impact on classroom dynamics to be applicable in other fields as well.

Pedagogy

The interaction model that Classroom Presenter supports is simple: Students submit ink-annotated slides to the instructor, and the instructor selectively incorporates student responses into the discussion.

Multiple goals. One discovery from our deployments was the variety of creative ways in which instructors harnessed this simple interaction mechanism to achieve a wide range of instructional goals. Figure 3 shows example in-class activities used with Classroom Presenter from several courses and the associated pedagogical goals for each activity.

The brainstorming activity presented in Figure 3a was from an environmental science course. The instructor planned to collect a wide range of answers, possibly all valid, to use in the discussion.

Figure 3b illustrates a digital-design course activity that was used to generate rich student artifacts for discussion. The instructor felt that discussing student-generated examples would be more likely to uncover common misconceptions and would have more impact than if the instructor were to discuss self-manufactured examples of such common errors.

Figure 3c shows an activity from an algorithms course that was designed to lead students to discover for themselves a particular point of knowledge. It relied on the fact that students could work independently and not merely

Computer Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

watch the instructor solving the problem. The instructor designed this activity to lead students to an unexpected "A-ha!" moment, when they would realize the impossibility of finding a solution under the given constraints.

Figure 3d demonstrates a software-engineering course activity intended to assess students' prior knowledge, providing the instructor feedback on how many students understood the concepts and how well, and serving as a catalyst for discussing commonly held misunderstandings about testing.

Instructor awareness. We have also been impressed by how aware instructors have been of their pedagogical goals while designing activities using Classroom Presenter. This awareness comes about naturally when designing an activity-based lecture: The process causes the instructor to focus on developing activities to promote specific learning goals, as opposed to merely covering material.

As with the adoption of any new classroom technique, instructors unfamiliar with designing lectures in this manner might encounter an initial increase in their preparation time, but experience with the medium and reuse of materials has shown that this effect can be ameliorated over time.

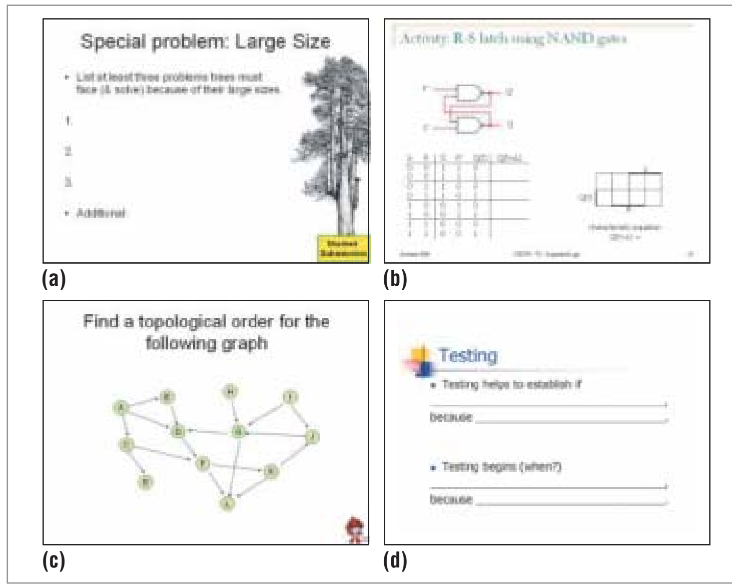


Figure 3. In-class activities used with Classroom Presenter. (a) Brainstorming activity from an environmental science course. (b) Activity from a digital-design course used to generate rich student artifacts for discussion. (c) Algorithms activity designed to lead students to discover for themselves a particular point of knowledge. (d) Activity from a software-engineering course intended to assess students' prior knowledge and serve as a catalyst for discussing commonly held misunderstandings about testing.

Digital ink

The choice of the Tablet PC platform was central to the design of Classroom Presenter; this decision was motivated by a desire to provide users with the flexibility of writing instead of typing as the primary input mechanism.

Student expression. Digital ink lets students express themselves in various ways, offering instructors insight into student thought processes and generating interesting artifacts for discussion. The effects of this freedom are evident throughout an activity's workflow.

Figure 4 illustrates the range of expression that digital ink has supported in student submissions.

In the activity shown in Figure 4a, the ability to annotate an existing diagram was critical. The instructor was able to evaluate solutions quickly by visual inspection of the annotation.

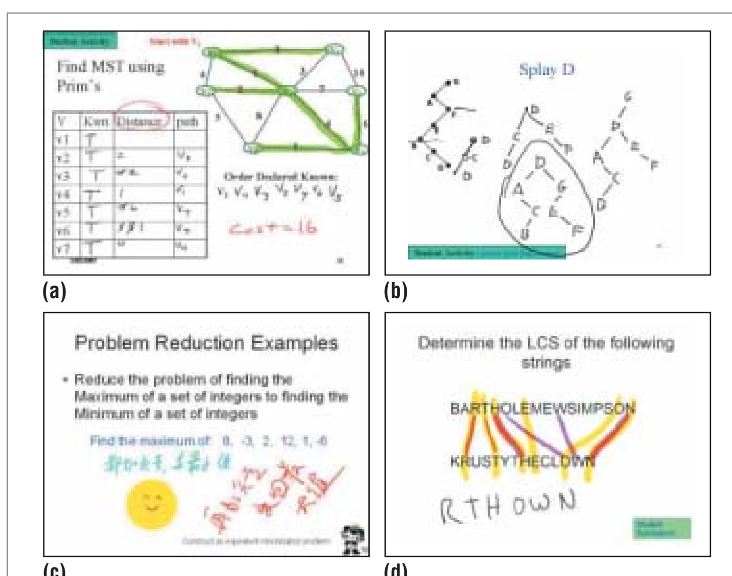


Figure 4. Student submissions. (a) The ability to annotate an existing diagram lets the instructor evaluate solutions quickly by visual inspection. (b) Diagrams like this one would be very difficult to produce using textual input only. (c) Using digital ink lets students respond in their own language as well as differentiate their responses using color. (d) Digital ink can be used to highlight the existence of multiple answers using color.

Diagrams like that shown in Figure 4b would be very difficult to produce using textual input only. Here the student also demonstrates the series of steps taken to transform an original graph into the final answer.

Figure 4c presents a submission from a class in China; the use of digital ink allowed students to respond in their own language as well as differentiate individual responses using color. The face motif appears regularly on these students' submissions (though not always smiling).

In Figure 4d, the student used digital ink to highlight the existence of multiple answers using color. This was an unexpected response: The instructor anticipated receiving only one solution per submission.

Range of activities. Digital ink lets instructors create activities beyond what would be appropriate for textual input or multiple-choice selection tools like "clickers."[7] Its flexibility of expression enables students to include diagrams, mathematical symbols, or arbitrarily complex drawings in their responses. Meanwhile, the ability to distribute rich prepared slides to student devices and have students annotate these images with digital ink allows the instructor to provide structure for the activity.

Problem solving. Students' use of digital ink also strongly influences the discussion around their submissions. Digital ink is a natural medium for solving problems, not just reporting solutions. Being able to see the steps that students have taken to solve a problem helps both the instructor—by providing valuable insight into student learning—and other students—by suggesting alternative approaches to their own. Digital ink also lets students express their answers in unique and unanticipated ways, yielding rich artifacts for discussion.

Classroom atmosphere. Perhaps most important is the impact of using digital ink on the classroom atmosphere. Digital ink uniquely identifies work as having originated from a student, allowing peers to relate to the artifact in a fundamentally different way than they might to instructor-generated content. In addition, students appreciate seeing their own work displayed; even though the system provides anonymity, students often use digital ink to personalize their responses with emotive symbols and phrases, doodles, and identifiable icons.

Classroom dynamics

Successful use of Classroom Presenter creates a new classroom dynamic—one based on collaboration and discussion.

Public display of student work. One of the central ideas of the student submissions scenario is the integration of student work into the discussion by showing it on a public display, effectively putting student efforts on the same footing as instructor content. The potential for showing their work is a strong motivator for students to contribute. Survey results indicate that students enjoy seeing their work displayed and that anonymity is also very important to them.

Giving more students a voice. In a traditional classroom, when students are prompted to answer questions verbally, typically only a few eager and vocal students contribute. In contrast, the use of Classroom Presenter gives each student time to consider the question and a chance to submit a response. This interaction model encourages instructors to allow enough time for all students to respond, while the anonymity that the system provides motivates shy or less confident students to participate too. The focus on the content value of submitted responses rather than on the individual contributors effectively prevents a small number of students from dominating the conversation.

Collaboration. Our initial deployments were with one tablet per student when possible; however, in the few cases when students needed to share tablets, a different dynamic emerged. The tablet form factor and the sharing of a pen facilitated student collaboration, including considerable discussion, and increased engagement in activities. Indeed, having students work together has been so successful that we now prefer to have multiple students share a tablet even when equipment is available for one-to-one deployment.

We have developed and deployed Classroom Presenter to explore a set of classroom interaction scenarios, particularly in student submissions, for enhancing student engagement in class. As an application targeted for the Tablet PC, Classroom Presenter capitalizes on the flexibility and range of expression that digital ink affords. Initial deployments of our system have shown that instructors can exploit this technology not only to achieve a wide range of instructional goals successfully but also to create a more participatory and collaborative environment. ■

Acknowledgments

We greatly appreciate support from Microsoft Research External Research and Programs, Hewlett-Packard, and the National Science Foundation, which has made this work possible.

References

1. J.D. Bransford, A.L. Brown, and R.R. Cocking, eds., *How People Learn: Brain, Mind, Experience, and School,* expanded ed., National Academy Press, 2000.
2. W.J. McKeachie and M. Svinicki, *McKeachie's Teaching Tips: Strategies, Research, and Theory for College and University Teachers,* Houghton Mifflin, 2006.
3. R.J. Anderson et al., "A Study of Digital Ink in Lecture Presentation," *Proc. SIGCHI Conf. Human Factors in Computing Systems,* ACM Press, 2004, pp. 567-574.
4. V. Razmov and R. Anderson, "Pedagogical Techniques Supported by the Use of Student Devices in Teaching Software Engineering," *Proc. 37th Technical Symp. Computer Science Education,* ACM Press, 2006, pp. 344-348.
5. R. Anderson et al., "Supporting Active Learning and Example-Based Instruction with Classroom Technology," *Proc. 38th Technical Symp. Computer Science Education,* ACM Press, 2007, pp. 69-73.

6. M. Wilkerson, W.G. Griswold, and B. Simon, "Ubiquitous Presenter: Increasing Student Access and Control in a Digital Lecturing Environment," *Proc. 36th Technical Symp. Computer Science Education,* ACM Press, 2005, pp. 116-120.
7. R.J. Dufresne et al., "Classtalk: A Classroom Communication System for Active Learning," *J. Computing in Higher Education,* Mar. 1996, pp. 3-47.

***Richard Anderson** is a professor in the Department of Computer Science and Engineering at the University of Washington. His research interests include educational technology, pen-based computing, and computing for the developing world. Anderson received a PhD in computer science from Stanford University. He is a member of the IEEE, the IEEE Computer Society, and the ACM. Contact him at anderson@cs.washington.edu.*

***Ruth Anderson** is a lecturer in the Department of Computer Science and Engineering at the University of Washington. Her research interests include educational technology and computer science education. Anderson received a PhD in computer science and engineering from the University of Washington. She is a member of the ACM. Contact her at rea@cs.washington.edu.*

***Peter Davis** is a research assistant in the Department of Computer Science and Engineering at the University of Washington. His research interests include software architecture, computational linguistics, and programming languages. Davis received a BS in computer science from the University of Washington. Contact him at pediddle@cs.washington.edu.*

***Natalie Linnell** is a graduate research assistant in the Department of Computer Science and Engineering at the University of Washington. Her research interests include educational technology, computer science education, and computing for the developing world. Linnell received a BS in mathematics and computer science from the University of Minnesota. Contact her at linnell@cs.washington.edu.*

***Craig Prince** is a graduate research assistant in the Department of Computer Science and Engineering at the University of Washington. His research interests include human-computer interaction, pen-based technology, and educational technology. Prince received an MS in computer science and engineering from the University of Washington. Contact him at cmprince@u.washington.edu.*

***Valentin Razmov** is a graduate research assistant and software engineering instructor in the Department of Computer Science and Engineering at the University of Washington. His research interests include human aspects in software engineering and management, computer science education, and methods for effective teaching and learning. Razmov received a PhD in computer science and engineering from the University of Washington. He is a member of the ACM and the American Society for Engineering Education. Contact him at valentin@cs.washington.edu.*

***Fred Videon** is a software engineer in the Department of Computer Science and Engineering at the University of Washington. His work focuses on distance learning applications. Videon received a BS in electrical engineering from Stanford University. Contact him at fred@cs.washington.edu.*

REACH HIGHER

Advancing in the IEEE Computer Society can elevate your standing in the profession.

Application to Senior-grade membership recognizes

- ✔ ten years or more of professional expertise

Nomination to Fellow-grade membership recognizes

- ✔ exemplary accomplishments in computer engineering

GIVE YOUR CAREER A BOOST ■ UPGRADE YOUR MEMBERSHIP

www.computer.org/join/grades.htm

COVER FEATURE

Facilitating Pedagogical Practices through a Large-Scale Tablet PC Deployment

Joseph G. Tront
Virginia Tech

The Virginia Tech College of Engineering has begun to explore the use of Tablet PCs in engineering and computer science courses. Using a multifaceted, collaborative approach, the faculty developed an implementation process that includes computer acquisition, faculty training, infrastructure modifications, and multiple evaluations and assessments demonstrating positive initial results.

Undoubtedly, when Socrates and Plato met for their many conversations, they discussed ways to improve student learning. Although they didn't have to ponder the effect of using modern computing and communications technology, they must have discussed ways to actively involve students in the learning process.

While today's outstanding teachers still rely on Socrates' techniques of drawing students into the learning process, now many of them are turning to technology to help facilitate these active and collaborative exercises. Mobile computing and communication devices like the Tablet PC, along with a high-bandwidth communication infrastructure, help increase the quantity and quality of teaching/learning interactivity with the expectation of improving student learning.

In the fall of 2006, the Virginia Tech College of Engineering became the first public college of engineering to require all 1,400 incoming students to own a Tablet PC. The purpose of this requirement program is to better facilitate pedagogical practices that are expected to improve learning—practices not readily accomplished in previous teaching/learning environments. This program is expected to support highly interactive classroom presentations, as well as student-student and instructor-student collaborations. It will also foster comprehensive note taking and review and emphasize more process-oriented lectures as opposed to simple information broadcasting.

This large multifaceted deployment requires the enthusiasm and support of numerous stakeholders. Decisions on hardware and software choices require input from across the university. Training of faculty and support personnel is central to the initiative's success. Physical plant challenges include infrastructure improvements such as network connectivity, additional classroom projection systems, and increased availability of power connections. Sound and frequent assessment of the program's successes and failures and identification of potentially rewarding future possibilities have been part of the overriding deployment strategy from the beginning.

BACKGROUND

The Virginia Tech College of Engineering faculty has a long history of continuously seeking ways to improve the teaching and learning environment to effectively provide students with a high-quality engineering education. Many teaching innovations have been implemented with support from the college administration, alumni, and various research agencies such as the National Science Foundation (NSF). Innovations include incorporating freshman hands-on mechanical dissection labs, integrated subject material courses, and multidisciplinary projects. Most notable among the teaching/learning

Published by the IEEE Computer Society
0018-9162/07/$25.00 © 2007 IEEE

innovations are the college's efforts to foster the effective use of computing and communication technology in the curriculum.

In 1984, the Virginia Tech College of Engineering was the first public institution to require all entering engineering freshmen to own a personal computer. In the early 1990s, Virginia Tech participated in the NSF-sponsored Southeastern University and College Coalition for Engineering Education (SUCCEED) and assumed the lead role in the association's effort to conduct research on technology's effects on engineering education. By 1996, the computer requirement had been scaled up to the so-called multimedia computer, which incorporated features that were advanced for the time, including a CD-ROM reader, a high-resolution graphics system, and a sound card—features we take for granted in today's computers.[1]

In 2002, the college moved to a laptop requirement, and many of its academic buildings were outfitted with a wireless communication system that gave students high-speed Internet access from anywhere on campus. Laptop technology was selected so that students could perform computing and communication operations in a completely mobile environment. Today's ubiquitous computer use in students' everyday learning practices and lifestyles provides anecdotal evidence that these technology requirement programs have been fruitful.

The college once again steps out on the technology forefront by requiring all students to own computationally powerful and well-connected Tablet PCs. In addition, the college is making a stronger effort to assess specific effectiveness measures. The assessment goal is to understand how to improve important pedagogical and learning practices and to identify general learning advancements that occur as a result of these practices.

TABLET PC DEPLOYMENT

In 2002, the faculty began pilot projects seeking ways to take advantage of the Tablet PC's electronic ink (e-ink) capabilities in the engineering education environment. Much like the standard blackboard or whiteboard, the instructor can use this technology to make dynamic and adaptive presentations that are more responsive to student interaction than a simple PowerPoint presentation.

PowerPoint presentations offer advantages over blackboards in that the instructor can easily organize them, and they can contain images that help bring real-world situations to the classroom. PowerPoint also aids in broad distribution of classroom notes, which for engineers and scientists can contain complex drawings that would be nearly impossible to copy during a lecture.

Free software like Classroom Presenter,[2] a Tablet PC presentation tool, combines PowerPoint's advantages with the flexibility and spontaneity of traditional blackboard lectures. Using Classroom Presenter, an instructor can prepare drawings and graphics in ready-made form and then annotate discussion points on the electronic slides during the lecture.

Instructors normally leave schematic drawings of problems incomplete and finish them during the presentation. This causes students to pay better attention during class instead of occasionally glancing up from their stupor as chock-full PowerPoint slides glide by on the screen. Using this new paradigm that essentially combines PowerPoint and a blackboard can better elicit and answer typical student "what-if" questions. Most importantly, students can take home a composite of the predrawn PowerPoint presentation supplemented by the in-class annotations for review and study.

Collaboration improves desirable learning outcomes in academic programs, including academic achievement, interpersonal interactivity, self-esteem, and learning retention.

Initial use of Classroom Presenter produced positive responses in student polls taken after the tablet-based classroom presentations. Attendance increased as students found the lectures more interesting. Given the early success of using Tablet PCs in simple presentations, the faculty began to identify other opportunities for using this technology to facilitate pedagogical practices that are known to improve learning.

TARGETED PEDAGOGICAL IMPROVEMENTS

Engaging students in the learning process by having them participate in an active discussion or problem-solving session with the instructor and with their peers has been shown to improve learning. According to Michael J. Prince and Richard M. Felder,[3,4] "The core elements of active learning are student activity and engagement in the learning process. Active learning is often contrasted to the traditional lecture where students passively receive information from the instructor." Richard Hake[5] used pre- and post-test data to examine more than 6,000 students in introductory physics courses and reported significantly improved performance for students in classes with substantial use of interactive engagement methods.

Collaborative learning consists of two or more students working together to solve a problem or understand a concept as opposed to individual work on a topic.[6] Studies show that collaboration improves desirable learning outcomes in academic programs, including academic achievement, interpersonal interactivity, self-esteem, and learning retention.[7-9]

When used appropriately, the Tablet PC's rich communications and multimodal input capabilities can increase learning interaction in the classroom. This technology can facilitate intense collaborative

Table 1. Tablet PC hardware requirements.

Item	Detail
Platform	Tablet PC convertible
OS	Windows XP Pro Tablet Edition
Processor	Pentium Core 2 Duo 1.8 GHz
Memory (RAM)	2 Gbytes
Hard disk	100 Gbytes; 5,400 RPM
Video card	128 Mbytes
Optical drive	DVD/CD+–R writable DVD
Input/output	USB 2.0
Wireless	802.11 a,b,g
NIC/Ethernet	10/100/1000 Ethernet
Warranty	3 years for accidental damage
External backup	USB external backup drive, 160 Gbytes

activities using software that is either currently available or under development.

The characteristic that differentiates between notebook and tablet technology is the user's ability to more naturally jot down ideas and sketch drawings that can be communicated with other collaborators on shared electronic surfaces. Meaningful tablet-based collaborations can take place either locally or over distances separated by the Internet.

Comprehensive, organized, and easy-to-review note taking is an effective learning behavior that increases subject cognition.[10-12] Keiichi Kobayashi's research on the impact of note taking and reviewing on student learning revealed that properly performed note taking substantially improves learning outcomes and demonstrated that assisting students in improving their note-taking skills can produce additional positive benefits.[13] Tablet PCs can improve student learning by allowing note taking in a natural manner and by improving the ability to review notes either through ease of search or increased organizational capabilities.

Achieving the expected outcome of enhanced student learning is based on improving three key pedagogical practices:

- increased active learning,
- incorporating collaborative exercises into the learning process, and
- improved note collection and note searching/review.

To support these practices, the hardware and software selected for student and faculty use must be sufficiently capable, faculty must be trained in the use of the technology as well as in appropriate pedagogical practices, students must have a baseline understanding of the technology and its expected use, and sufficient infrastructure and support personnel must be available. The overarching umbrella to all of this effort must be an assessment operation that formatively measures the accomplishments of the program, identifying the most likely avenues for success as the initiative progresses.

COMPUTER SPECIFICATIONS

In the fall of 2007, incoming Virginia Tech engineering students will purchase their Tablet PCs on the open market using a set of specifications that the college issued in mid-April. As Table 1 shows, the minimum computer specifications are a compromise among price, capability, longevity, and reliability.

For example, specifying the Pentium Core 2 Duo processor and the associated 2 Gbytes of RAM means that the computers will be capable of running all of the required software at reasonable speeds, while remaining affordable to the majority of the entering class.

Including wireless access cards capable of 802.11a, b, and g formats ensures the best opportunity to avoid overlapping broadband communications in the classroom and dormitory wireless infrastructure. While these minimum specifications might seem high initially, they are intended to ensure that the computer will be usable four years later, when the senior student will likely be performing computationally intense calculations and simulations.

Historically, about 40 percent of the entering students purchase the minimum hardware package, while the remaining 60 percent add higher-level capabilities to their systems such as a flat-panel monitor, increased RAM, more disk drive space, or extra video RAM.

An often-overlooked consideration is the Tablet PC's weight. The tablet cannot be too cumbersome because the student is expected to bring it to class every day. This leads to a conflict between the desire to have a large display screen and the additional weight of the upsized screen.

Typically, the 12-inch-screen machines weigh about 4.5 pounds, which is considerably less than the 14-inch-screen machines, which can weigh 8.5 pounds. The smaller devices are generally encouraged not only because of their lighter weight, but also because of their form factor, which lends itself to easier use on a typical classroom desktop.

Students also must purchase the engineering software bundle, which complements the hardware by providing the computing capability necessary in typical engineering learning environments. The minimal software suite for the student's Tablet PC is similar to what practicing engineers in industry might have access to in their design environments, including

- Matlab,
- Autodesk Inventor and Mechanical Desktop,
- PDF Annotator,
- Labview, and
- Microsoft Campus Agreement including Office Professional, OS upgrades, OneNote, Visual Project, Visual Studio, and Client Access Licenses.

The process of selecting hardware and software considered both the educational program's needs and the hardware vendors' expected offerings. Discussions took place in nondisclosure meetings conducted early in the year, taking into account whether vendors could deliver the hardware in the July/August timeframe. A vendor's unfulfilled promise can be disastrous if students are left without a computer to start the semester. To avoid these difficulties, the college established working relationships with reputable vendors and provided pertinent information to students and their families.

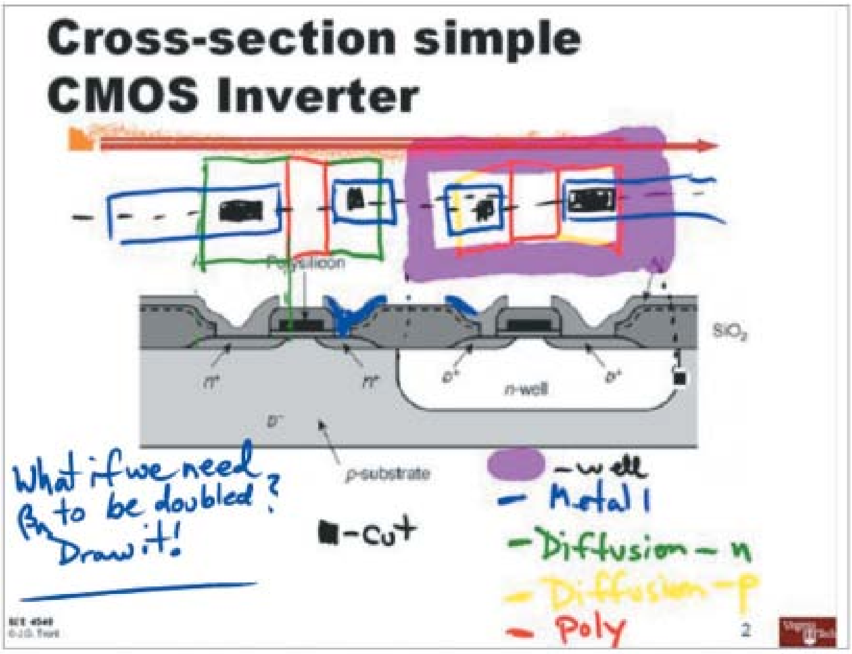


Figure 1. Annotated presentation. The instructor's slide accommodates dynamic annotations and challenge questions. The presentation is no longer limited to static, preplanned information.

TRANSFORMING CLASSROOM PRESENTATIONS

To effectively use Tablet PC technology, instructors must transform their teaching style and modify their instructional materials. They are using several e-ink-enabled software tools, including Classroom Presenter[2] and Dyknow Vision (www.dyknow.com), two mainstay tools that support more dynamic presentations and increase student interactivity.

Using these tools, a teacher generates instructional material by producing a set of PowerPoint-based slides to act as a framework for the lecture. The slides consist of pictures, diagrams, equations, Web page clippings, and other electronic materials to support the lecture.

The instructor then modifies the slides, hiding portions of diagrams or sections of equations to be filled in during the classroom discussion. Hidden portions are visible on the instructor's screen, but not in the public view or on the students' screens. The instructor uses e-ink to fill in the missing material dynamically in class. This encourages students to reflect on the material during the lecture rather than simply listening to a verbal reiteration of what is already shown on the screen or a reading of preprinted notes.

E-ink's power lies in its ability to show dynamically the process of developing a schematic or inking an equation's terms just as a practicing engineer does. The instructor's value-added factor is in showing the personalized process of development; without this personalization, the students might as well just read the book and skip the lecture.

Students become more engaged in the natural discussions in class because now they can view and participate in the design process rather than simply receiving a solution. In addition, they are not overwhelmed by the perceived need to mechanically copy notes.

Integrating interactivity is a second-level effect that the new tools facilitate. During the lecture, the instructor poses an open-ended question and asks the students to respond with a solution that is typically graphical in nature. Students write a solution on their tablet and submit it electronically to the instructor, who then chooses submissions to display anonymously and discuss with the class. Students become particularly engaged when instructors use this technique—they look forward to responding and are disappointed if the class doesn't discuss their solution.

Figure 1 shows an example of an annotated presentation screen in which the instructor spontaneously generates a question and asks the students to respond with answers to be displayed and discussed. The key to success here is the use of e-ink and the high-speed communication that facilitates the interactivity. Using this technique accommodates various student personality traits, ranging from those who are outgoing and are the first to volunteer an answer to those who are shy and rarely proffer a comment. The faculty has observed a more even distribution of students paying attention in class and more active discussion both during and after the lecture.

Another advantage to this presentation paradigm is the student's ability to generate a local, personalized electronic version of class notes. Both Classroom Presenter and DyKnow can broadcast notes and the instructor's e-ink in real time to students in the classroom. Students can then add their own e-ink and save the composite notes on their machines for later review.

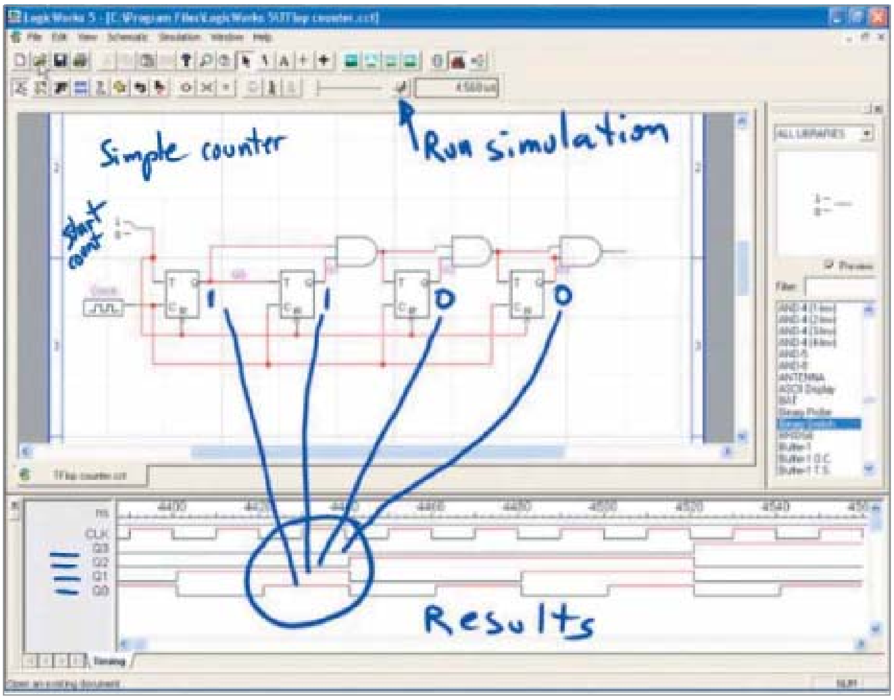


Figure 2. WriteOn tablet-based tool. The instructor can annotate directly on the operating simulator's output window, and students can save annotations and simulator output for later review.

This is a powerful mechanism that encourages in-depth thinking and enables reflections that are key to understanding complex concepts.

Engineering and computer science classes typically use dynamic simulators or other visualization tools. Instructors usually show the simulation's visual aspects in class and ask students to operate the simulator outside class, requiring them to remember the material presented during the lecture. With WriteOn, a new tablet-based software tool, the instructor can make notes on top of a dynamically running program and capture the annotations along with the simulation display as either a set of still images or as a movie of all onscreen activity.[14]

Figure 2 shows a logic simulator on which the instructor has made notes about the behavior shown in the simulator-generated waveform. WriteOn lets the instructor provide value-added information that previously might have been presented in a rather dry and difficult-to-remember demonstration.

Several other similar tablet tools are under development or are currently available to support classroom interaction in the Microsoft Tablet PC Education Pack (www.microsoft.com/windowsxp/downloads/tabletpc/educationpack/default.mspx).

Note taking and collaboration

Microsoft OneNote, a software tool for general note taking, provides an electronic notebook that mimics a pencil and paper paradigm but has several advanced capabilities. For example, with OneNote the user can rapidly search the entire notebook and locate handwritten words that relate to a concept. The built-in handwriting recognition facility does this with a high level of correctness.

Students also can use OneNote to record the audio portion of lectures while they take notes. In postclassroom sessions, students can click on e-ink objects in OneNote and play back the relevant audio clip, allowing very specific review of classroom information.

Collaboration is a powerful mechanism for reinforcing learning and preparing students for real-world design and development activities. OneNote's built-in collaboration facility lets participants share the electronic notebook's common sections.

Figure 3 shows a typical exercise in which several students work together to solve a problem. When they join a shared session, they can see the e-ink that other participants generate. The students can contribute their own e-ink, typed information, or any other electronic object they can extract from the electronic clipboard. A collaborator can modify or erase any object on the screen. Participants can be local or remote—with Internet access being the only requirement.

Initial classroom collaboration experiences indicate that most students are willing to participate in this type of interaction, but they do not have the refined skills required to derive maximum benefit from the exchange of ideas. However, after exposure, students quickly learn the basic requisites for effective electronic collaboration such as personal identification, appropriate sequencing, and idea formulation.

Electronic homework submission

Electronic homework submission has typically been difficult for engineering students since much of what is submitted consists of not just text as in a typical English or history class, but sketches intermingled with text along with the occasional picture. Several tablet-based tools offer students more flexibility in producing submissions. Word, OneNote, PDF Annotator, and Adobe Acrobat all allow applying e-ink annotations to typed documents, making it easier for students to produce electronic homework submissions.

For instructors, electronic homework submissions generally are easier to handle and grade. For example, upon receiving a submission either through e-mail or a classroom management system, the instructor can mark it up and return it to the student without having to use class time for paper collection or distribution. This also more readily preserves student privacy. Several faculty members who have used this scheme for a few semesters are pleased with the way it has streamlined the process and identify this practice as significantly increasing their efficiency.

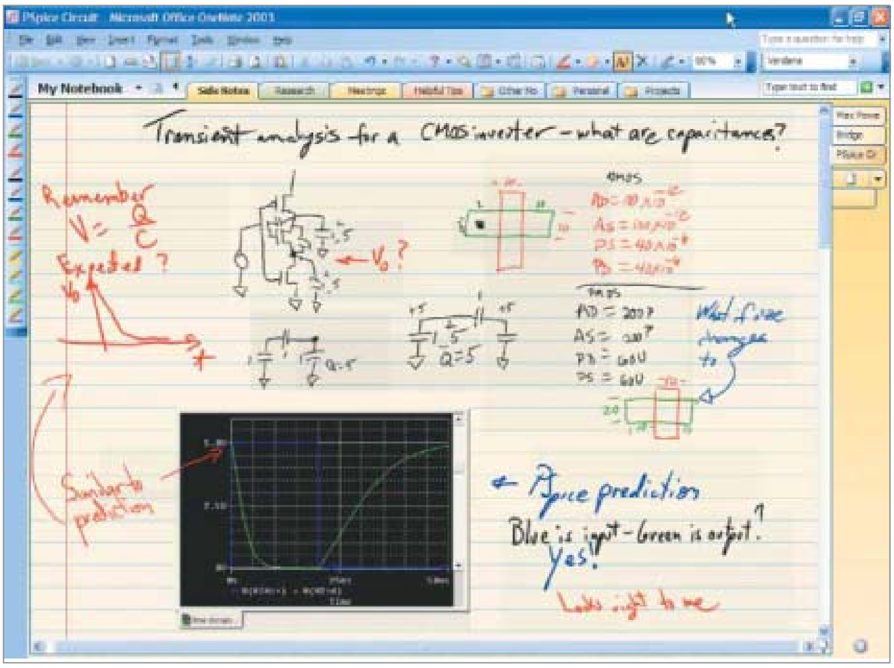


Figure 3. OneNote. Students can use the software tool to collaborate when solving a problem. Multiple participants in distributed locations can participate in the visual conversation.

FACULTY TRAINING AND SUPPORT

To take advantage of the new technology, faculty members are encouraged to participate in the Faculty Development Institute (FDI), a series of workshops where faculty are trained on pedagogical practices as well as details of the technology's operation. Early adopters of the new technology with knowledge of the difficulties and solutions teach the tablet workshops. Approximately 25 percent of the faculty receive training from the FDI each year.

Five or six times throughout the semester, faculty study groups meet to discuss progress and exchange tips and tricks for success. The faculty receives additional technical support at the beginning of the semester to help resolve in-class issues such as projector settings and network connectivity. Thus far, the faculty response has been positive, with participants enthusiastically working on developing and modifying materials as they participate in workshop sessions.

ASSESSMENT—PLANS AND PROGRESS

Assessing the tablet requirement program's effectiveness is paramount in ensuring that it produces the targeted pedagogical improvements. The assessment process is built around a core set of measures that are gauged each semester by having students respond to three surveys. A student-learning strategies instrument was adapted from the Motivated Strategies for Learning Questionnaire.[15]

The adapted MSLQ is used in a pre- and post-test design to measure changes in students' learning strategies using the Tablet PC during each semester. Adding a subset of questions from the national Educause Center for Applied Research study helps determine how students will self-report on their use of technology compared to other engineering students nationally. A second utilization questionnaire is distributed midsemester to measure the frequency and nature of technology use.

A survey administered longitudinally to a sample of the engineering faculty assesses faculty response to the use of the Tablet PC. Developed from extant teaching measures in the literature, the survey addresses not only the faculty's use of instructional technology but also their more general teaching practices and pedagogical beliefs. Responses will help determine if the faculty's philosophy and practice change over time and, in particular, if they become more attuned to active learning and the need to increase student engagement and collaboration in their teaching.

As the assessment program ramps up, school officials will use other measures to determine the Tablet PC's impact on how students organize and think about course materials, collaborate with other students, and participate in class.

Data collected and analyzed so far has focused on the impact on student note taking and whether using Tablet PCs encourages metacognitive strategies and critical thinking skills in individual studying and note taking. While confirming the Tablet PC's value in collegiate instruction, the results also have raised some technical and instructional issues related to its use.

Using the MSLQ data, the faculty painted a picture of an incoming freshman engineering student's learning strategies and technology use and assessed the changes after a semester in the program. In September 2006, 61 percent of freshman engineering students reported they did not have access to computers in their high school classes; by midsemester nearly all students reported using their PCs on a daily basis.

In the preliminary phases of data analysis, we see some significant changes in student learning habits as they apply to note taking and studying, and we will continue to measure and analyze changes as students progress though their degree program. As we collect and analyze data, we will report on results related to the measurement instruments' validity.

Using a multifaceted, collaborative approach, the Virginia Tech College of Engineering has developed an implementation process for using Tablet PCs that includes computer acquisition, faculty training, infrastructure modifications, and multiple assessments for program evaluation. Initial results of this groundbreaking program are positive, showing measurable improvements in pedagogical practices that are ultimately expected to lead to learning improvements. Various aspects of the program's processes are scalable and extensible to other institutions and to the science, technology, and mathematics disciplines. As we proceed, we imagine the types of Tablet PC-based dialogues Socrates and Plato might have had and how useful it would be to have searchable e-ink archives of those conversations. ■

Acknowledgments

The author acknowledges Glenda Scales, associate dean for engineering computing at Virginia Tech, for her leadership in this effort and Deborah Olsen and Kimberly Filer for their work on the program's assessment aspects. This endeavor's success is due mainly to the broad participation of our innovative engineering faculty and students. Thanks also to Microsoft Research and Fujitsu Computers, which, along with the Virginia Tech College of Engineering, have formed the Premier Alliance, which supports the efforts to effectively use Tablet PCs.

References

1. J.G. Tront, "A Personal Computer Requirement for Engineering Students," *Proc. ICEE 99 Conf.*, 1999, pp. 348-350; www.ineer.org/Events/ICEE1999/Proceedings/papers/348/348.htm.
2. R. Anderson, "UW Classroom Presenter," Univ. Washington Computer Science and Engineering; www.cs.washington.edu/education/dl/presenter.
3. M. Prince and R. Felder, "The Many Faces of Inductive Teaching and Learning," *J. College Science Teaching*, Mar./Apr. 2007, pp. 14-20.
4. M. Prince, "Does Active Learning Work? A Review of the Research," *J. Eng. Education*, July 2004, pp. 223-231.
5. R. Hake, "Interactive-Engagement vs. Traditional Methods: A Six-Thousand-Student Survey of Mechanics Test Data for Introductory Physics Courses," *Am. J. Physics*, vol. 66, no. 1, 1998, p. 64.
6. B. Smith and J. MacGregor, "What Is Collaborative Learning?" *Collaborative Learning: A Sourcebook for Higher Education*, A. Goodsell et al., eds., National Center on Postsecondary Teaching, Learning and Assessment, 1992, pp. 9-22.
7. D. Johnson, R. Johnson, and K. Smith, *Active Learning: Cooperation in the College Classroom*, Interaction Book Co., 1998.
8. D. Johnson, R. Johnson, and K. Smith, "Cooperative Learning Returns to College: What Evidence Is There that It Works?" *Change*, July/Aug. 1998, pp. 26-35.
9. L. Springer, M. Stanne, and S. Donovan, "Effects of Small-Group Learning on Undergraduates in Science, Mathematics, Engineering and Technology: A Meta-Analysis," *Rev. Educational Research*, vol. 69, no. 1, 1999, pp. 21-52.
10. M.P. Ryan, "Conceptual Models of Lecture Learning Guide Metaphors and Model-Appropriate Notetaking Practices," *Reading Psychology*, Oct. 2001, pp. 289-312.
11. N. Purdie and J. Hattie, "The Relationship between Study Skills and Learning Outcomes: A Meta-Analysis," *Australian J. Education*, Apr. 1999, pp. 72-86.
12. F.J. DiFesta and G.S. Gray, "Listening and Note Taking," *J. Educational Psychology*, Feb. 1972, pp. 8-14.
13. K. Kobayashi, "Combined Effects of Note-Taking/Reviewing on Learning and the Enhancement through Interventions: A Meta-Analytical Review," *Educational Psychology*, June 2006, pp. 459-477.
14. J.G. Tront and V. Eligeti, "WriteOn: A Tool for Classroom Presentations on Tablet PCs," *Proc. ITiCSE 06*, ACM Press, 2006, p. 352.
15. T.G. Duncan and W.J. McKeachie, "The Making of the Motivated Strategies for Learning Questionnaire," *Educational Psychologist*, vol. 40, no. 2, 2005, pp. 117-128.

***Joseph G. Tront** is a professor in the Bradley Department of Electrical and Computer Engineering at Virginia Tech. His research interests include the effective use of technology in education, embedded computers, security in mobile computing and communication devices, and microelectronics design. He received a PhD in electrical engineering from the State University of New York at Buffalo. He is a Senior Member of the IEEE. Contact him at jgtront@vt.edu.*

Computer Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

www.computer.org/election/

Click and Vote Online!

IEEE computer society

IEEE Computer Society Election

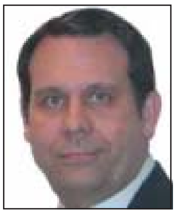
Nominees for IEEE Computer Society Offices and Board of Governors Positions in 2008

On the following pages are the position statements and biographies of the IEEE Computer Society's candidates for president-elect, first and second vice presidents, and Board of Governors. Within each category, candidates are listed in alphabetical order. Election of officers to one-year terms and of Board members to three-year terms, each beginning 1 January 2008, will be by vote of the membership as specified in the bylaws.

Ballots must be returned no later than 12:00 noon EDT on Tuesday, **2 October**. Members in all regions can vote via the Web at www.computer.org/election or by fax with Election Services Corp. at +1 516 248 4770. Return ballots by mail to the IEEE Computer Society, c/o Election Services Corp., PO Box 9209, Garden City, NY 11530-9009, USA. For replacement ballots, call +1 516 248 4200.

Results will be announced in the December issue of Computer. The opinions expressed in the statements are those of the individual candidates and do not necessarily reflect Computer Society positions or policies.

Nominees for president-elect



James D. Isaak

Position statement. Our Society must attract the next generation of technologists and phase out financial dependence on subscriptions. We have not fully leveraged our IEEE relationship, and now is the time to do so by delivering state-of-the-art services with lower fees.

While serving on the IEEE Board of Directors, as well as the Computer Society Board of Governors, I have been an innovative catalyst for change. In chairing the IT Strategy Committee, I have helped to establish a foundation for personalized services and related revenue opportunities and worked to procure the eCollaboration/community tools that are essential to attracting leading-edge professionals.

Challenge: Rapid innovation in our field. *Path forward*: Anticipate emerging areas and attract leaders; structure the IEEE and Computer Society for agile reconfiguration and response to opportunities.

Challenge: Declining membership. *Path forward*: Deliver a 21st century professional environment; champion individual participation; 2nd Life, Zude, what's next?

Challenge: Financial imbalance. *Path forward*: New business models and revenue streams.

Challenge: Competing with "free online." *Path forward*: Apply the Google approach to funding services.

Challenge: Attracting students into field. *Path forward*: Leverage IEEE and other precollege programs; develop a positive visibility program.

Challenge: Disruptive career environment. *Path forward*: Continuing education, social networking, employer-independent benefit programs.

Challenge: IEEE relationship. *Path forward*: Cross-pollinate IEEE and Computer Society leadership to build trust and common cause.

Please visit my Web site, www.JimIsaak.com, for details on how we can address these challenges.

My diversity of leadership roles within the IEEE provides the experience needed to address these challenges and the social network essential to make it happen. Many years in industry, a few in academia, and a lifetime of global collaboration provide me with an appreciation of our membership.

I am a proud recipient of the Hans Karlsson Award for "Outstanding leadership and achievement through cooperation," which reflects the approach I favor.

(Isaak continued on next page)

Susan K. (Kathy) Land

Position statement. This year, as the Computer Society's first vice president, I have been leading an effort to implement a revised business model for conference operations. This rollout has been challenging and follows a two-year review of the Society's conferences portfolio and subsequent analysis of the previous business model. These changes were prompted, in part, by a need to address the financial situation of the Computer Society, which is facing deficits in the foreseeable future and, in part, to provide a simplified process for conference organizers.

During my years as a volunteer with the Society, I have been privileged to serve in several different areas, including two years as vice president of standards activities and as a member of the Board of Governors. I feel my experience uniquely qualifies me to understand the challenges we now face and to continue working with our volunteer leaders and staff in pursuit of our plans for restructuring. We have made significant progress in our pursuit of organizational transformation by establishing deliberate policy and intentional directional changes to support the Society's position of advancing the theory, practice, and application of computer and information processing technology. We must recover our revenue streams to rebuild and better serve our membership. To do this will require concerted and continued effort by the senior leadership of the IEEE Computer Society.

As volunteers, we should all look for ways to help the Society provide its customers with total solution improvements to facilitate the retention of the Society position as the recognized authority and source for defining how software and systems are developed, tested, and maintained. If elected, I will strive to ensure that IEEE Computer Society products are relevant to the marketplace, are affordable, and provide a consistent view of the state of the practice. I will continue to support the definition of initiatives and directions that enable collaboration, support interoperability, strengthen our marketing, and sustain our plans for fiscal improvement.

For additional information in support of my experience and vision for the future of the Society, please visit www.susankathyland.com.

(Land continued on next page)

(Isaak continued)

If you are a volunteer, thank you. Participation fosters innovation and career growth—with positive global impact. Today's professional looks to Google for answers. We must provide the environment that delights tomorrow's technologists.

I appreciate your vote.

Biography. Jim Isaak has served in numerous leadership roles.

IEEE leadership: IEEE Board of Directors (2003-2005); Chair, IT Strategy Committee; Computer Science Accreditation Board; IEEE USA Committee on Communications and Information Policy; Standards Association Board of Governors; Technical Activities Board management; New Hampshire Computer Society chapter chair.

Computer Society leadership: Board of Governors (1997-2008); vice president, Standards Activities; vice president, Technical Activities; chair, PASC (Posix/Unix) Standards Committee; chair, Internet Best Practices Standards Committee.

Other roles: ISO/IEC JTC1/SC22/WG15 (Posix) convener; X/Open Board of Directors; Society for the Social Implications of Technology; Internet Society Advisory Council.

Education and employment: Isaak currently teaches IT at Southern New Hampshire University; MS-EE in computer engineering, Stanford University.

30 years in industry: DEC (director-information infrastructure standardization); Charles River Data Systems (strategic planning); Data General (product manager, systems engineering manager); Intel (test technology); IBM (network simulation).

Awards: Hans Karlsson; Outstanding Contribution; Meritorious Service; Computer Society Golden Core; IEEE Millennium Medal; US patent 6,622,247.

(Land continued)

Biography. Kathy Land is employed by MITRE, a not-for-profit organization chartered to operate in the public interest, which manages three federally funded research and development centers for the US government. She has more than 20 years of industry experience in the practical application of software engineering methodologies, the management of information systems, and leadership of software development teams.

Land has served on the Computer Society Board of Governors and in positions as first and second vice president. Land is the 2007 recipient of the IEEE Standards Association Standards Medallion. She is a current member of Computer Society bodies that include the Standards Activities Board, Software and Systems Engineering Standards Executive Committee, Professional Practices Committee, Membership Committee, Planning Committee, Computer Society International Design Competition Committee, Computer Society History Competition 61, and is currently chair of the Technical Achievement Award subcommittee.

Land is author of *Jumpstart CMM/CMMI Software Process Improvement: Using IEEE Software Engineering Standards* (John Wiley & Sons, 2005). She is coauthor of *Practical Support for CMMI-SW Software Project Documentation: Using IEEE Software Engineering Standards* (John Wiley & Sons, 2005), and *Practical Support for ISO 9001 Software Project Documentation: Using IEEE Software Engineering Standards* (John Wiley & Sons, 2006).

Nominees for first vice president

George V. Cybenko

Position statement. The next several years present great opportunities for the IEEE Computer Society. Distribution channels for technical publications are changing dramatically, so we have to rethink both our print and paper publications businesses to lead rather than follow trends. In addition, the demographics of the Society's membership will be changing as the global economy evolves, so we need to be constantly responsive to the changing needs of members.

Computer professionals still value technical content highly but are less willing to pay for it through subscriptions or society membership dues. This is especially true for developing economies and for younger professionals who have smaller discretionary budgets. Moreover, people are becoming more comfortable with Web searches and more free material is out there for them to find.

Professional societies have become large and hierarchical, leading many members to have only superficial, intermittent interaction with the societies. I believe there is much to do in redefining the IEEE Computer Society and its mechanisms for interacting with its members. This is the area I believe I can contribute to most as first vice president. In particular, it is important to better harness the human networking potential of the Society, both in hard-core technical issues and in individual professional development. We can start this by shifting our Web presence from the library model, where users check books out and do not talk, to the café model, where talking, thinking, and interacting are expected and rewarded. I welcome your ideas and support in realizing these visions.

Biography. George Cybenko has served two terms on the IEEE Computer Society Board of Governors (2002-2004 and 2005-2007). He has been involved in a variety of IEEE Computer Society publications activities, as founding editor in chief of *IEEE Computing in Science and Engineering* and as founding editor in chief of *IEEE Security and Privacy*. Cybenko has also served as member, author, editor, and conference organizer for several other IEEE bodies, including the Signal Processing, Information Theory, and Control Systems Societies. He currently holds the Dorothy and Walter Gramm Professor-

(Cybenko continued on next page)

Sorel Reisman

Position statement. This is my first opportunity to publicly thank you for electing me second vice president. I hope that the Electronic Products and Services Board's accomplishments have met your expectations and that you will continue to have faith in my leadership.

Last year, I advocated refocusing the Society's mission upon new electronic tools and services to meet the expectations and needs of our globally diverse practitioner and academic membership. My premise was, and continues to be, that it is essential for us to provide an electronic infrastructure to enable you to communicate better with one another, to share ideas, and to cooperate in inventing new technologies, systems, and applications, regardless of your country of residence. Over the past year, I have actively participated in planning committees on which I was an outspoken voice, promoting the transformational structures and processes necessary to make this happen. Consequently, we are now reinventing our Web presence with a completely redesigned, communities-based www.computer.org. This infrastructure, set for release in 2008, will be the foundation for our new, cost-economical, revenue-producing, Web 2.0-centric future.

While I am proud of this accomplishment, I am looking forward to next year, toward developing the kinds of new products, services, and policies that will capitalize on this new technology base and allow us to become a robust and thriving international membership enterprise.

To learn more about me, my board's accomplishments, my ideas, hopes, and plans, please visit www.sorelreisman.com. If you don't have time, then please just vote for me!

Biography. Sorel Reisman is managing director of the international consortium MERLOT, and professor of information systems and decision science at California State University, Fullerton. His 21-year academic career follows more than 15 years of senior management positions in systems development/marketing at IBM, Toshiba, and EMI. He is second vice president of the Electronic Products and Services Board and a 2006-2007 member of the Transformation, Planning, and Membership Committees. He has served

(Reisman continued on next page)

(Cybenko continued)

ship of Engineering at Dartmouth College. Prior to joining Dartmouth, Cybenko was a professor of electrical and computer engineering at the University of Illinois at Urbana-Champaign. He received a BS in mathematics from the University of Toronto and a PhD in electrical engineering and computer science from Princeton. Cybenko is a Fellow of the IEEE.

(Reisman continued)

as chair and member of the eLearning, Web Redesign, Digital Library, and Communities committees. Reisman served multiple terms as member-at-large of the Publications Board, chair of the Magazine Operations Committee, chair of six EIC search committees, editorial board member/columnist for *IEEE Software*, founding editorial board member of *IEEE Multimedia* and *IT Professional*, and author of the column, "The Ivory Tower." He is a member of the IEEE IT Strategy Committee and reviewer for *IEEE Transactions in Education*. Reisman has presented/published more than 50 articles and the books *Multimedia Computing: Preparing for the 21st Century* and *Electronic Learning Communities—Current Issues and Best Practices*. He serves as liaison to various international digital library consortia. Reisman received his electrical engineering degree, an MA, and a PhD in computer applications from the University of Toronto.

Nominees for second vice president

Antonio Dória

Position statement. Our Society is being challenged. We are running a structural deficit, membership is dropping, and market share is shrinking. We face hard times. Rather than hardship, I see a huge opportunity for improvement and for increasing efficiency.

If elected, I will work to ensure that we develop products, programs, and services that are relevant and consistent with the state of the practice. I will focus on attracting underserved communities such as practitioners, students, and young professionals, while strengthening our value to academics. I will explore new market opportunities by expanding our services to corporations.

Our Society should be inclusive. I will work to make our products, programs, and services affordable to all, especially to those members living in economically stressed areas like Latin America, Africa, Eastern Europe, and Asia. I intend to do this by diversifying our distribution model in order to meet our members' and customers' needs.

As second vice president of the Computer Society, I will work diligently to ensure the long-term sustainability of our Society by discontinuing or adapting existing products, programs, and services that are not adding value to our membership.

Change hurts, and can be disruptive, but it provides a good foundation and represents an opportunity to create a promising and healthy future. If we want to continue being the leading society in computing, we need to act.

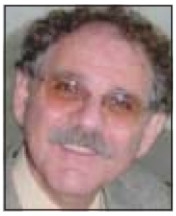
I ask for your support. It will be an honor to continue serving you. For more details about my vision for the Society, please visit my Web site, www.matakiterani.com/ieee-cs.

Biography. Antonio Dória has been an active Society volunteer since 2000. Currently, he is a member of the Board of Governors and is the Society's vice president and chair of the Chapters Activities Board. During his tenure, the Chapters Activities Board has focused on initiatives to bring the Society closer to members, especially in underserved regions such as Europe, Latin America, Africa, and Asia.

He serves on the Society's Executive Committee, History Committee, Intersociety Cooperation Committee, Membership Committee, Planning Committee, Outreach Committee, New Initiatives Committee, and Transformation Plan Committee. Dória served on the Executive Director Search Committee as Region 9 liaison to the Computer Society and as Ecuador Chapter chair.

Dória is chairperson and chief software architect at Matakiterani, a Portugal-based software house specializing in business intelligence solutions. Earlier, he was the technology vice president of Enterprise Software Solutions in Miami, where he led the development of an international ERP solution.

An IEEE senior member and the IEEE Region 9 student activities chair, Dória has earned several certificates of appreciation for his contributions to the IEEE and the IEEE Computer Society.



Michel Israel

Position statement. The Computer Society is at a crossroads; it is imperative that we renew our commitment to our essential purposes. Publishing magazines, journals, and proceedings are important aspects of our work, but not everything. We must facilitate the technical interchange among members through seminars, conferences, workshops, chapters, educational programs, and other means as well as publications. Computer engineering and science is practiced in an international environment. Hardware and important software systems are developed for worldwide use. As an international professional society, we can help our members understand the culture and approaches of those in other countries, enhancing their professional skill set in the era of globalization of science as well as business.

Creation and promotion of international standards, brokering cooperative research, certain certification programs, and international standards for undergraduate curricula are opportunities that need our continued and renewed support, especially when new programs for science and engineering will be launched through US agencies such as the National Science Foundation. The Society's Washington headquarters affords us a fantastic opportunity to organize international cooperation through the world's embassies also located there. Meanwhile, the Society faces significant financial problems, many deriving from the overly centralized procedures of our parent IEEE. Some improvements have been made, but we must carefully progress toward a better management. If elected, I will use my experience to help modernize our management, straighten out our relations with the IEEE, and extend our international presence and cooperation with national professional societies to make real our status as "The World's Computer Society."

Biography. Michel Israel, an IEEE senior member, has served the Society for 25 years as VP for Technical Activities, Chair of the Central & Eastern European Initiative Committee, Treasurer, DATC chair, Secretary, Ombudsman, Chair of the European Activities Committee. He served on the Nomination, Audit, Award, and Membership committees and Conference and Tutorial boards.

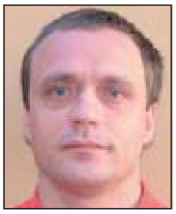
Israel has been the coordinator for the French Association on IT. He was a member of the Accreditation Board for CS of the French Ministry of Education.

An Outstanding Professor, Israel, now the Scientific Counsellor of the French Embassy in Washington after Tokyo, was Dean of the Faculty of Sciences and chair of the CNRS CS lab at the University of Evry. He was the EU chair of a FIPSE EU-US exchange program, a visiting professor at the University of Galatasaray, Turkey, at the University of Toronto and is developing an American-France doctoral network.

Israel, a PhD in computer sciences from Paris 6 University, received different awards: Distinguished Service award for VP TAB and CEEIC Chair; Meritorious Service as DATC chair; Outstanding Contribution for the establishment of the first computer chapter in the former USSR.

He is a CS Golden Core member and received the IEEE Millennium medal.

Board of Governors nominees (13 nominees; vote for seven)



Alfredo Benso

Position statement. "Do not go where the path may lead; go instead where there is no path and leave a trail." – Ralph Waldo Emerson

This is the spirit I try to follow, from teaching to sailing, and that I will maintain if elected to the Board. We must be innovative in the technology we support and in the services we provide.

If elected, it will be my goal to contribute in creating new paths to get back in touch with our members worldwide, helping them to share their knowledge and ideas by providing them with the services they need. I want them to remember the excitement of being a computer professional, and to keep considering the Computer Society as a supporter of the time, energy, and passion they dedicate to their profession.

Misquoting Spander: "The great thing about democracy is that it gives every voter a chance to do something smart."

Here's your chance…

Biography. Alfredo Benso was born in Italy in 1970. He studied at Politecnico di Torino, Italy, receiving a PhD in 1998. Since 2005, Benso has held a tenured associate professor position there in computer engineering. He currently teaches microprocessor systems and advanced programming techniques. Benso has coauthored more than 60 publications and has been involved in the organization of several Computer Society-sponsored conferences.

Benso became a Computer Society volunteer in 1999 as chair of the Test Technology Technical Council database task force. Currently, he is a member of the Electronic Products and Services, Conferences and Tutorials, and Technical Activities Boards. As a volunteer, Benso has been honored with three "Outstanding Contribution Awards" and a Golden Core membership. He is also a senior member of the IEEE.

Benso shares his time with his family and many other hobbies and interests, among which are sailing, surfing, and being a cofounder of Project Lisa (www.projectlisa.org), a nonprofit organization for reef restoration in the Mediterranean Sea.

Fernando H. Bouche

Position statement. The IEEE Computer Society strives to be the world's leading society for computer professionals. We must continue to reach out to our international communities and address the needs of IT professionals and students. To achieve our vision, the Society must listen to what our customers tell us and we must respond decisively and positively.

As a senior volunteer leader in Region 9 for many years, I worked hard to increase the Computer Society's membership and student chapter activities in Latin America. I also worked closely with volunteer leaders from other parts of the world to improve the value of our offerings and to ensure that we provide products that are relevant, affordable, and of technical value.

If elected, I will continue to work to increase our representation, membership, and participation worldwide and to address how we can best provide the services and products that the computer professionals of today and tomorrow need.

Biography. Fernando Bouche is the IT manager at the Smithsonian Tropical Research Institute in Panama and an adjunct professor at the Universidad Latina de Panamá. He received an MS in IT management from Universidad Latina de Panamá and a BS in systems engineering from Universidad Tecnológica de Panamá. As a student, he received the IEEE RAB Larry K. Wilson Regional Student Activities Award, among other recognitions from the IEEE. Bouche served as member of several IEEE bodies such as IEEE GOLD, Ethics, and Member Conduct Committees. He has served as a member of the Chapter Activities Board as Region 9 representative, Distinguished Visitor Program Latin America coordinator, and PACE representative. As a CAB member, Bouche has traveled to several IEEE and Computer Society international meetings as a representative of the Society.

Joseph R. Bumblis

Position statement. I'm a scientist by training and am very distressed to see that in many countries relatively little emphasis is placed on science and engineering by society in general and kids in particular. As a result, the next generation of scientists is growing up wanting to be lawyers and accountants. Computers are an integral part of the new generation's lifestyle—their brains are hardwired almost from birth to interact with the universe and each other through digital means. Because of this, I believe that the Computer Society holds a key responsibility to reach out to these kids. This is my personal passion, and it is why I work in the role I have at Microsoft.

If elected, I will direct my energy toward helping the Computer Society take a worldwide leadership role in bringing excitement back to science and engineering for kids, partnering with industry, government, teachers, schools, and parents.

Biography. Van Eden has a PhD in physics and worked for several years in research on surface microscopy and analysis at IBM Research and the University of Washington. His experiences led him to the realization that computing holds the key to the future of science, so he moved into software development for data acquisition and analysis.

After working for several scientific instrumentation companies, in 2000 he joined Microsoft Research in Cambridge, where he managed a team working with academics across Europe, the Middle East, and Africa, to help them learn about MSR and the contribution it makes to advancing technology. Eden created several new programs to improve knowledge transfer from MSR to academia—for example, "Microsoft Academic Days," a series of conferences in which Microsoft scientists present the science behind Microsoft technologies.

In 2004, following his passion for education, Eden transferred to Redmond, Wash., where he manages programs that support students and educators at all levels in the US. In primary and secondary education, the focus is on making science, engineering, and math more interesting to students, especially girls and minorities. In higher education, the focus is on research and how Microsoft technologies can be used to teach principles and concepts.

André Ivanov

Position statement. The IEEE and the Computer Society benefit from an unsurpassed worldwide reputation. However, with the current technological, business, social, and political trends, to not only maintain but also grow this reputation further, the Society needs to rapidly and effectively modernize its offerings and services. These have to be supported and deployed with viable business models that are adaptive and responsive to specific needs and capabilities of the different constituent communities. New activities and models of operation that revive existing membership and, more importantly, attract new membership and volunteers, especially in emerging technologies and emerging countries, are vital to the Society's success. I have been involved in many Society activities that have taught me about the needs and aspirations of professionals around the globe. I intend to bring this experience to the Board of Governors to enhance the Society's rejuvenation and its evolution toward serving all professionals with undisputed excellence.

Biography. André Ivanov, who holds a PhD from McGill University, currently chairs the Computer Society Test Technology Council and is a professor of electrical and computer engineering at the University of British Columbia. He has published widely and is a holder of several patents on test and reliability of integrated circuits and systems. Ivanov has served on the steering, program, or organizing committees of several international events sponsored by the Computer Society. He is currently serving on the steering committee of the International Test Conference.

Ivanov was technical program chair of the VLSI Test Symposium in 2001 and 2002 and general chair in 2003 and 2004. In 2004, he founded the first IEEE International GHz/Gbps Test Workshop. Ivanov now serves as associate editor for *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* and for *IEEE Design and Test of Computers*. He has served on the Society's Conference and Tutorials Board and its Technical Activities Board. In 2007, he was vice chair of the Computer Society Fellows Committee. He is a Golden Core member of the Society, a Fellow of the IEEE, and a registered professional engineer in British Columbia. In 2001, Ivanov cofounded Vector 12, a semiconductor IP company.

Hai Jin

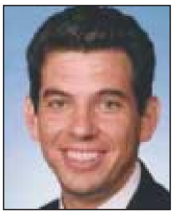
Position statement. I am interested in serving you as a member of the IEEE Computer Society Board of Governors, and I seek your vote in this regard. I have been involved with computer architecture, parallel and distributed processing, and network storage-related research and education over the past two decades as student, educator, researcher, and volunteer and have seen its amazing growth into today's technology. Being a candidate from the Asia-Pacific region, especially from a country containing one-quarter of the world's population, I will promote the IEEE Computer Society in this region, especially by expanding curricular support to vast numbers of students, attracting more young researchers and practitioners to be Society members, and fostering collaboration between researchers from this region and those in other regions of the world.

Please feel free to contact me at hjin@hust.edu.cn or visit my homepage, http://grid.hust.edu.cn/hjin, if you have any further questions about my candidacy.

Biography. Hai Jin is a professor of computer science and engineering at the Huazhong University of Science and Technology in China. He is now Dean of the School of Computer Science and Technology at HUST. Jin received a PhD in computer engineering from HUST in 1994. In 1996, he was awarded a German Academic Exchange Service fellowship to visit the Technical University of Chemnitz in Germany. Jin worked at the University of Hong Kong between 1998 and 2000 and as a visiting scholar at the University of Southern California between 1999 and 2000. He was awarded the Excellent Youth Award from the National Science Foundation of China in 2001. Jin is the chief scientist of ChinaGrid, the largest grid computing project in China.

Jin is a senior member of the IEEE and a member of the ACM. He has coauthored eight books and published more than 200 research papers. His research interests include computer architecture, cluster computing and grid computing, peer-to-peer computing, network storage, and network security.

Jin is a member of the steering committee of the IEEE/ACM International Symposium on Cluster Computing and the Grid, the IFIP International Conference on Network and Parallel Computing, and the International Conference on Grid and Cooperative Computing.



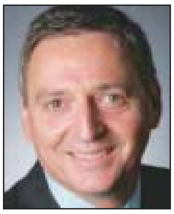
Phillip A. Laplante

Position statement. The Computer Society faces significant challenges from competition, rapid changes in industry, global geopolitical realities, and internal pressures from the IEEE. In order to meet these challenges we need to continue to aggressively align our programmatic offerings by ensuring that our publications and other products are more relevant to both academic and professional audiences. We also need to continue to find linkages between print products, conference activities, certifications, courseware, and other deliverables by expanding and mining our extensive database of authors, reviewers, conference and course attendees, and readers. In addition, the Computer Society can only move forward with the understanding and cooperation of the IEEE. I intend to leverage my experience and relationships with the IEEE and influential members in other societies to help further the cause of the Computer Society.

Biography. Phillip A. Laplante holds a PhD in computer science from Stevens Institute of Technology and is a professor of software engineering at Pennsylvania State University's Great Valley School of Graduate Professional Studies, where he conducts research in software engineering and teaches graduate courses to working IT and software engineering professionals. He is also the chief technology officer for the Eastern Technology Council, a regional advocacy organization, and is a licensed professional electrical engineer in Pennsylvania.

Laplante has served the Computer Society as a member of the Publications Board, chair of the Conference Publishing Operations Committee, member of the Press Operations Committee, and member of the editorial board for *IT Professional*. For the last 20 years, he has also served on many IEEE committees including the IEEE Press editorial board, Continuing Professional Education Committee, Periodicals Review Committee, the Magazines Committee (as chair), and most recently, as chair of the Periodicals Committee.

Laplante is a senior member of the IEEE and was elected a Fellow of the International Society for Optical Engineering for his contributions to real-time imaging research. He also received the IEEE Educational Activities Meritorious Service Award for his ongoing contributions to and innovations in continuing engineering education.



Gérard Medioni

Position statement. The environment, including the way we organize conferences, publish and access results, has drastically changed over the past few years, and we need to take a fresh look at our practices.

I have organized a number of conferences, both as program chair and as general chair, and served on the IEEE Conferences and Tutorials Board. In 2001, I introduced a major change to our annual meeting, transforming it from a conference into a week-long event, with workshops and tutorials before and after, all available through a passport registration. The response from the community has been enthusiastic. I hope to leverage this acquired experience as a member of the Board of Governors.

Biography. Gérard Medioni received the Diplôme d'Ingenieur from ENST, Paris in 1977 and an MS and PhD from the University of Southern California in 1980 and 1983, respectively. He has been at USC since then, and is currently professor of computer science and electrical engineering, chair of the Computer Science Department, codirector of the Institute for Robotics and Intelligent Systems, and codirector of the USC Games Institute.

Medioni has made significant contributions to the field of computer vision. He has published three books, more than 50 journal papers, and 150 conference articles, and is the recipient of eight international patents. Medioni is associate editor of the *Image and Vision Computing Journal*, the *Pattern Recognition and Image Analysis Journal*, and the *International Journal of Image and Video Processing*.

Medioni served as program cochair of both the 1991 IEEE CVPR Conference and the 1995 IEEE Symposium on Computer Vision. He also cochaired the 1997 IEEE CVPR Conference, the 1998 ICPR Conference, the 2001 IEEE CVPR Conference, and the 2007 IEEE CVPR Conference. He is a Fellow of IAPR, a Fellow of the IEEE, and a Fellow of the AAAI.

Itaru Mimura

Position statement. I am honored to be nominated to serve on the IEEE Computer Society Board of Governors. I believe that the IEEE Computer Society is the leading international organization of computer professionals. The IEEE Computer Society can use its body of professional experience and intellectual property to promote a huge, worldwide computer-related market, especially in Region 10 (China, Southeast Asia, and Japan).

In recent years, I feel that technical contributions from industry to the Computer Society have become less than sufficient. Based on my industrial carrier of more than 20 years, human networks, and experience on the Board of Governors, I will make an effort to consolidate more feedback from industry and will try to build a bridge between industry and the Society.

I have the time, the experience, and the enthusiasm to devote myself to realizing a brilliant era for the IEEE Computer Society. I ask for your support.

Biography. Itaru Mimura serves on the IEEE Computer Society Board of Governors (2005-2007). He was a member of the Audit Committee in 2006 and has contributed to several committees as a Region 10 representative since he was appointed to the Board.

He received a BS and an MS in image science from Chiba University, Japan, in 1982 and 1984, respectively. He joined the Central Research Laboratory of Hitachi in 1984, where he worked on digital video compression/transmission technology and communication network equipment research for more than 20 years.

Mimura was a research affiliate of the Media Laboratory at MIT from 1991 to 1992. Since 1998, he has led the next-generation IP router research team as a department manager. His research results have been adopted on Hitachi's router products, which are widely deployed in Japan's networks. His team earned notice in 2000 and 2001 for developing "One of the hundred most technologically significant new products of the year (R&D 100)," according to *R&D Magazine*.

In October 2004, he joined AlaxalA Networks, a joint venture of NEC and Hitachi. Mimura is currently responsible for developing technology breakthroughs for next-generation IP networks, which support a service-rich information communication society.

Raghavan Muralidharan

Position statement. The IEEE Computer Society continues to face challenges in multiple areas such as finance, value for membership, relevance to industry professionals, competition, and declining membership. Industry, too, faces challenges of convergence of communications and computers, probably resulting in the eventual disappearance of conventional computers and software. Products are giving way to services on the Web and building upon newer technologies. These new developments need standards, best practices, and continuing education support. Our Society and industry are riding a wave of challenges and changes that compel us to invest more time and effort to ensure a smooth sail ahead. It is at this critical juncture that I seek your support to strengthen and transform our Society and to prepare for the days ahead. We will have to work together toward this goal in many areas, some of which are certification programs, conferences, membership development, standards for new technologies, and publications.

Biography. Raghavan Muralidharan has been an IEEE volunteer for 27 years and a computer and communications industry professional for 28 years. Muralidharan earned an undergraduate degree in electronics and telecommunication engineering from the College of Engineering in Trivandrum, India in 1977. From 1977 to 1979, he conducted postgraduate studies in computer-focused electrical engineering at the Indian Institute of Technology in Bombay, India.

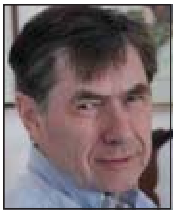
Muralidharan has played a major role in the development of the IEEE in India, especially over the past 12 years. His work with the IEEE Bombay Section has resulted in tremendous membership growth and the establishment of many student branches. As chair of the IEEE India Council, Muralidharan propelled Region 10 to become the largest region in the IEEE in 2004.

He works as technical vice president at Nayna Networks India in the areas of optical networking and fiber-to-the-x solutions. Previously, he was head of the computer and communications group at Tata Electric's research and development labs.

Muralidharan's IEEE activities include:

- India Council: chair (2003-2004); past chair (2005-2006)
- IEEE Bombay Section: chair (1999-2000); past chair (2001-2002)
- Computer Society Bombay Chapter: secretary (1983-1986); chair (1999-2001)
- ACE: general chair (2000, 2003)
- INDICON: general cochair (2004)

Computer Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page



Jon G. Rokne

Position statement. I believe that we are in a period of discontinuities. Technological advances are devaluing previous ways of conducting business and opening up new opportunities.

We should take advantage of the opportunities presented by the technological advances by making the Computer Society a truly international organization. This would require a concerted effort in attracting new members outside North America, as well as encouraging communications with content of interest to the international community. The use of fast electronic communications would enable the society to accomplish this.

One of the first action items that I would propose would be to establish a forum for exchanging innovative ideas. I would encourage experimentation based on these ideas with the expectation that some of the experiments could lead to transformations that would enable to the Society to move with the times.

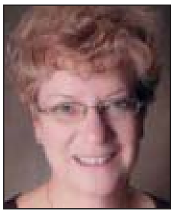
The board should also concentrate on strategic issues and leave the day-to-day details to the staff.

Biography. Jon Rokne, currently vice president of publications for the IEEE Computer Society, is a professor of computer science at the University of Calgary. He obtained a PhD in mathematics from the University of Calgary, where he chaired the Computer Science Department from 1989 to 1996. He has a working knowledge of Norwegian, German, French, and English.

In the Computer Society, Rokne has been a member at large of the Publications Board, chaired the Transactions Operations Committee, and chaired the ad hoc committee for ReadyNotes, in addition to serving on several other subcommittees. A Golden Core member, he is also currently a member of the Publications Board.

Rokne is a member of the IEEE Publication Services and Products Board and the Technical Activities Board Periodicals Committee. He also chairs the subcommittee on publications conduct, which deals with all IEEE plagiarism cases. He has been elected PSPB member at large for 2008-2010 and is a candidate for vice president of the PSPB.

Rokne has published extensively in mathematics, including three jointly authored books. In computer science, he has focused on computer graphics and physically and biologically based computer simulations. He published the jointly authored computer science book *Light Interaction with Plants*. In 2003, he organized the Pacific Graphics conference.



Christina M. Schober

Position statement. "The IEEE Computer Society's vision is to be the leading provider of technical information, community services, and personalized services to the world's computing professionals."

This can be accomplished by the continued involvement and enthusiasm of our student, affiliate, regular, senior, life, Gold, and Fellow members, as well as staff and volunteer leadership.

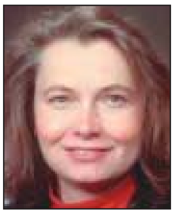
I have great respect for all the aspects of computing and its worldwide penetration into industry, schools, and homes. The diversity of our members and the wide breadth of computing technology is a challenge to providing value to each member. I will be dedicated as a Board of Governors member to having the enthusiasm to bring out new services and products, and a futuristic outlook on long-term planning and financial responsibility. I enthusiastically support continuing to offer Society products at their premier level of quality and working to increase our professional membership and prestige worldwide.

Biography. Christina M. Schober is an IEEE senior member and has been an active volunteer in the IEEE Computer Society since 1985. She currently serves as 2007 secretary and Executive Committee member on the Computer Society Board of Governors. Schober is vice president of finances for the IEEE Sensors Council.

Her IEEE volunteer work includes:

- IEEE Sensors Council: (2000-present, Executive Committee member 2004-present)
- IEEE Sensors Conference treasurer: (2002 to 2005, and 2007)
- IEEE Computer Society Board of Governors: (1998-2003, 2004-2007)
- IEEE Computer Society Conference & Tutorials Board: (1995-2005, vice president 2003-2004)
- IEEE Chapter Activities Board: (1986-2007, vice-president 2005- 2006)
- IEEE Twin Cities Section: chair and vice chair (1991, 1990)
- IEEE Computer Society Twin Cities Chapter: chair, vice chair, treasurer, and secretary between 1985 and 1988

Schober is an IEEE Computer Society Golden Core member and in 1990 was named the IEEE Twin Cities Section Young Engineer of the Year. She is a Six Sigma Black Belt, a Honeywell product team leader for the Tactical Ring Laser Gyro in the sensors group, and holds five ring laser gyroscope-related patents. Schober received a BME and an MME from the University of Minnesota. She is married and has two children currently attending college.



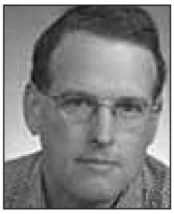
Ann E.K. Sobel

Position statement. As we reflect on the Computer Society's 60-year history, we should examine the composition of our membership throughout the years to identify the communities of members that we serve and to discover new communities that will attract large groups of potential members. Identifying these communities will direct us in how we might best serve each one, ultimately resulting in the support of all our current and future members. Such services should be assessed in light of the vast array of delivery options that current technologies provide. If we are unsuccessful in offering what any one community deems necessary or highly desirable, we could fail to sustain crucial subgroups within our membership. The Computer Society must therefore focus on identifying and providing the present and future needs of our communities so that we can meet our mission and maintain our vitality.

Biography. Ann Sobel has been involved in Society activities for many years, culminating with her service as a member of the IEEE-CS/ACM Curriculum Oversight Committee and as chair of the Certified Software Development Professional Training Committee. Sobel became a member of the Educational Activities Board, where she contributed to the creation of the CSDP examination. Her main contribution as an EAB member was her work on the *Computing Curricula: Software Engineering Volume*, serving as Body of Knowledge chair and later as the Steering Committee cochair. Sobel continues to serve on the EAB Executive, Professional Practices, and Awards Committees.

Sobel began her career as a research associate at the IBM T.J. Watson Research Center in Yorktown Heights, New York. She then became an associate professor in computer science at Ohio's Miami University. Her research interests include software engineering education and formal specification notations.

Sobel holds a PhD in computer science from the Ohio State University. She has received a Computer Society Outstanding Contribution Award and an ACM Certificate of Recognition for her work on the first IEEE Computer Society/ACM software engineering undergraduate curriculum guidelines. She has also received numerous awards from Miami University for leadership initiatives and breaking gender barriers.



Jeffrey M. Voas

Position statement. The Computer Society counts the most members and boasts the best-read publications within the IEEE, a proud accomplishment. However, because of how the IEEE is financially organized, smaller societies view us as the "800-lb. gorilla." The IEEE Technical Activities Board, which ultimately controls percentages of revenue returned to societies from publications, employs a distribution algorithm that negatively impacts the Computer Society. I, as a voting member of TAB from 2003 to 2005 witnessed this first hand. Consequently, we still face rapidly shrinking cash reserves. My goal is to help elevate Computer Society leaders, both volunteers and staff, to better positions within the current TAB hierarchy. A better-represented Computer Society will no longer be viewed as a "cash cow" target. Admittedly, the Society still requires immediate internal belt tightening. I believe that I grasp the overall issues and that the Computer Society can forge new win-win alignments with other societies.

Biography. Jeffrey Voas is director of systems assurance and a technical Fellow at Science Applications International Corporation. Before joining SAIC, Voas cofounded Cigital (www.cigital.com). Voas has been active in the software engineering research community for more than 15 years. He has served on numerous journal and magazine editorial boards. He was 2003-2005 president of the IEEE Reliability Society and has coauthored two books published by John Wiley & Sons. Voas is currently associate editor in chief of *IT Professional* magazine and serves on the advisory board of *IEEE Software* magazine, where he was an associate editor in chief from 1999-2003.

Voas received an undergraduate degree in computer engineering from Tulane University in 1985 and received an MS and PhD in computer science from the College of William and Mary in 1987 and 1990, respectively. Voas also performed two years of postdoctoral work for the National Research Council at the NASA Langley Research Center between 1990 and 1992.

IEEE Computer Society Election

Cast your vote quickly and easily on the Web at http://www.computer.org/election/ or via fax to +1 516 248 4770. Ballots must be received no later than 12:00 noon EDT on Tuesday, 2 October 2007.
To vote by mail, use the return-mail envelope provided and send your ballot to the address provided below.

Return ballots by mail to:
IEEE Computer Society
c/o Election Services Corp., PO Box 9209
Garden City, NY 11530-9009, USA

Computer | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | qMags

IEEE President-Elect Candidates Address Computer Society Concerns

The IEEE Computer Society has established a reputation for excellence within the computing field. As a component of the IEEE, the Computer Society's activities parallel those of 44 other IEEE societies and councils serving the computing and engineering disciplines. Representing by far the largest IEEE society contingent, the Computer Society has 76,885 members, approximately 60 percent of whom are full IEEE members.

Recognizing the influence of the IEEE leadership over the Computer Society and in turn the power of Computer Society members' votes to influence the selection of the IEEE leadership, we posed questions to this year's IEEE president-elect candidates. Because this election determines who will serve as president-elect in 2008, president in 2009, and past president in 2010—vital positions within the IEEE's governing body—our members must cast informed votes.

Our volunteer leaders have identified the following questions as essential to the Computer Society, the IEEE, and the Computer Society's relationship with the IEEE. The first response to each question states the Computer Society's position. These positions synthesize the views of our most senior leadership: the Society's current, past, and incoming presidents.

We present these questions and answers (limited to 150 words each) to help you make your decision in the IEEE election. Only ballots received by 1 November 2007 will be counted.

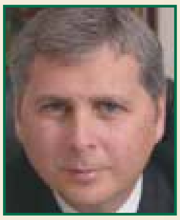
We also remind and encourage you to cast your vote for Computer Society leaders by 2 October 2007 in our Society election.

—Rangachar Kasturi,
IEEE Computer Society President-Elect

MARC APTER

Marc Apter received a BSEE from Pennsylvania State University (1964). He served in the US Navy for five years. He then worked for the US Navy as a civilian with the responsibility for installation of various electronic and other systems, developing electronic maintenance policy and auditing maintenance planning for new systems, managing part of the Navy's Metrology and Calibration Program, managing the Navy's Maintenance and Configuration Management Programs, and finally as the Command Information Systems Security Manager. He is currently a Senior Information Assurance Specialist for EG&G Technical Services, developing security documentation for government computer systems and applications. Apter has held many offices in the IEEE and has served on numerous IEEE committees and boards.

For more information about his candidacy, see www.marcapter.com.



PEDRO RAY

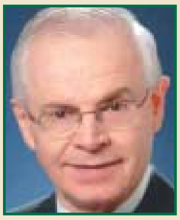
Pedro Ray successfully completed the Harvard Business School OPM program, class of 2001. He received a BSEE and MSEE from Georgia Tech. He is a licensed professional engineer.

He is president of Ray Architects Engineers, one of the largest design firms in Puerto Rico, with a subsidiary in Martinique. He owns or presides over various commercial and residential development corporations.

Ray was the chief examiner in charge of revision to the Puerto Rico Electricity Pricing Structure (2000) and was named Puerto Rico's Electrical Engineer of the Year 2000. He received the 2005 IEEE PES Chapter Outstanding Engineer Award.

Ray is a member of the YPO, a group of 10,000 young leaders from around the world.

For more information about his candidacy, see www.pedroray.org.



JOHN VIG

John Vig received a BS from CCNY in 1964 and a PhD in physics from Rutgers-The State University in 1969. He joined the Electronic Components Laboratory at Fort Monmouth, N.J., in 1969. Throughout his professional career, working as an electronics engineer and program manager, he performed and led research aimed at developing precision oscillators and sensors. In 1988, Vig was elected a Fellow of the IEEE.

He has 55 patents and has published more than 100 papers and nine book chapters. He serves as a volunteer in his home town as an environmental commissioner. In his spare time, he and his wife of 44 years enjoy ballroom dancing.

For more information about his candidacy, see www.johnvig.org.

Question 1: Financial Management

The IEEE is enjoying a positive financial position, and spending at the IEEE corporate level is increasing. Increased expenses translate into lower revenue returns and higher infrastructure costs for Societies.

How would you ensure that the IEEE follows prudent financial management practices?

How would you make sure that the Infrastructure Oversight Committee prevents excessive growth of the infrastructure?

How would you address the concerns of Societies that are running a deficit?

Society Position

The Computer Society encourages the IEEE president to lead the IEEE Board in limiting corporate spending. Increases in IEEE corporate spending decrease IEEE societies' capacity to offer services tailored to the needs of members and nonmember customers.

The IEEE president should work with the Infrastructure Oversight Committee to ensure that spending proposals have prudent and realistic goals and are thoroughly reviewed for financial soundness. Each proposal should have a periodic evaluation and a final assessment report reviewed by the IEEE Board of Directors.

The IEEE president should support efforts to have transparency in corporate financial statements so that volunteers and staff can better understand IEEE revenue and expense structures and be able to establish appropriate financial controls. The IEEE president should encourage IEEE volunteers and staff to work collaboratively and creatively to aid societies that are actively working to reduce their deficits.

Marc Apter

I would move to develop plans to implement the recommendations of the consultants that the Computer Society paid for over four years ago; I would have a team of volunteers, staff, and consultants look at our current financial formulas that have parts of the organization bleeding money when at the same time the reserves are growing in value by the many millions of dollars; and I would support the continuation of the efforts of the IEEE Fincom to move more costs from indirect to direct infrastructure charges. Since staff salaries are one of the largest growth items in the budget, and since the size of the staff has been growing, I would freeze the total staff size and require the Executive Director to manage within that total count for at least two years, while a joint volunteer/staff team identify possible staff positions that can be reutilized.

Pedro Ray

I guided the IEEE finances as treasurer with a set of principles that proved successful: realistic budgeting processes with each OU having a balanced budget (no deficit budget) and tight control over expenses. Recently, the IEEE has become more lenient in allowing more expenses to creep in. I have the experience to bring the IEEE online with good financial practices.

The Infrastructure Oversight Committee was recently overhauled. The members got longer terms so there could be continuity. The size and composition of the committee were changed to give it more power. As president, my role will be guiding and making sure the leadership of this committee is aligned with the financial goals of my presidency.

Some Societies are having difficult times, due mainly to the internal allocation (algorithm) of funds and infrastructure by the Technical Activities Board. I have the financial background and the will to solve this issue.

John Vig

Although it is true that "spending at the IEEE corporate level is increasing," revenues has been increasing faster, overhead rates have actually been declining. I will continue to support prudent budgeting and financial controls and program sunsetting, and I will make the Infrastructure Oversight Committee (IOC) more effective.

Currently, the IOC "… shall consist of… the IEEE Past Past President, who shall be Chair; the IEEE Past President, who shall be Vice Chair…" One way I would improve the IOC is by eliminating such entitlements. Those who are best qualified and are willing to make the necessary time commitment should be the chair and vice-chair.

Societies' problems are often rooted in Technical Activities Board (TAB) policies, especially in revenue distribution algorithms. Ultimately, however, it is up to the Societies to increase their revenues, or reduce their expenses, or both. I will address Societies' concerns in cooperation with the VP, TAB.

Question 2: Membership Models

The IEEE and its Societies face competitive challenges as other professional associations have created more creative outreach programs. Among other strategies, they have dramatically lowered prices and allowed for new models of affiliation outside traditional membership.

There have been recent efforts within the IEEE to improve membership value and expand the traditional IEEE member base. At the June Board meeting, the IEEE established a working group to evaluate alternate membership models.

What new membership models would you propose?

What is your position on lowering the affiliate fee?

Would you support the trial of a "low-barrier" offering to meet competitive challenges?

Society Position

The IEEE Computer Society endorses efforts to evaluate new membership models to increase the IEEE's membership worldwide and improve the value of membership. The IEEE's traditional member benefit of quality technical information faces increasing competition from commercial organizations and from the Internet. If the IEEE does not offer more flexibility in its member options and pricing structure, enabling members to select benefits tailored to their needs, and make fees affordable in certain markets, the IEEE will find its share of membership continuing to decline. A new membership model should support Societies' efforts by allowing more flexibility in membership requirements, particularly to attract IT practitioners who aren't engineers.

The Computer Society strongly supports lowering the affiliate fee, which is currently set at 50 percent of IEEE member dues. More than 90 percent of IEEE affiliates belong to the Computer Society. Affiliates receive no direct benefits from the IEEE, and the costs of servicing affiliate members are already covered by indirect infrastructure charges. Affiliate membership allows those who are uninterested in IEEE services or unable to afford full IEEE member dues to join. Affiliate membership enables the Computer Society to compete with other professional associations that can offer lower membership fees.

Marc Apter

I would do exactly what the question says—look at new models of affiliation outside traditional membership. With the consideration of these new models would come a reevaluation of what is currently known as the "affiliate fee." I would also include consideration for a trial of a "low-barrier" offering, based on the various membership-related studies of the past two years, to meet competitive challenges in countries where our potential members make salaries that in dollars are inadequate to afford our current dues structure.

Pedro Ray

We need to change our current membership model to create a different value interpretation. We need to change the value perception of what members believe they are getting for their money.

We need to repackage the higher grade membership with training/programs by experts, Xplore downloads, and other similar benefits.

At the lower end of the spectrum, we need to create an electronic membership tier for developing countries where now it would be impossible to join the IEEE even at our reduced prices.

As for an affiliate fee, I propose that we should revisit the subject and see what the competition is doing—find the price point, sensitivities, and make a determination.

I could support a trial of "low barrier," but we must not compete on price alone. We must not commoditize our membership because then we will all lose forever.

John Vig

We have repeatedly conducted membership surveys and discussed a variety of membership models. It is now time to experiment. Let us try a variety of models, for example, each in a small number of geographic areas. Let us try lowering the dues by making some offerings optional or, as some call it, "pay-by-the-drink."

I would definitely "support the trial of a 'low-barrier' offering" as this would be especially attractive for Societies facing lower-cost competition (for example, the Computer Society) and in countries where income levels make our dues unaffordable. Basic IEEE dues revenues represent <10 percent of total revenues. We have >US$200 million in reserves. We can afford to experiment.

I support lowering the affiliate fee. Moreover, I support allowing Societies to set their own fee. Affiliates can be a mechanism for growing the membership, attracting members whose primary professional society is other than the IEEE, and addressing price-of-membership concerns.

Question 3: Globalization

Aside from the issues of price and membership models, globalization poses other challenges to the IEEE. Examples include the protection of intellectual property, the need to coordinate standards activities, and the existence of and potential creation of regional groups formed to influence local public policy. Sometimes the positions taken on these issues by the IEEE or its units may appear to conflict with one another or with the IEEE's global vision.

Please comment on how you would work to foster a global approach to such issues.

Society Position

The most effective approach to manage opportunities and address challenges should be a collaborative, coordinated effort. The IEEE president should take the lead in coordinating the activities of entities within the IEEE. Careful coordination will avoid duplication of efforts and avoid creating confusion. Individuals and companies may not be able to distinguish overall IEEE policies from the policies of regional units focused on the needs of its members. This can lead to confusion and, worse, mistrust when the IEEE's global approach conflicts with an entity's localized position. Such conflicts discourage member and corporate support for IEEE and society activities.

Marc Apter

I would try to establish a policy that would require any IEEE subunits that intend to try and influence public policy to establish separate organizations controlled by the IEEE, but without having IEEE in their name. They must also be limited in area to only one country, except for the multiple countries that are part of the EU.

I would lead the effort required to have the IEEE really work with all our volunteers throughout the world to have them be directly part of the efforts as we address the protection of intellectual property and the need to coordinate standards activities.

Pedro Ray

The best approach is constant communication and coordination between the units (Societies and others) and the IEEE.

On the one hand, the IEEE needs to be nimble and be "out there," making decisions without interference from headquarters, like a federate kind of governmental system. On the other hand, there is the need in such a system of constant consideration that the global vision is followed and that the rights of the one stop where the rights of the others begin.

Also, we serve the whole IEEE membership, not just one country or a small segment of the IEEE. Conflicting views and positions will be vented and clarified internally before we make any public policy or statement. I will propose that the Executive Committee serve as the clearinghouse for such policy statements and as the resolution center for conflicts.

John Vig

I have proposed the creation of an IEEE Professional Activities Board. This Board would oversee and coordinate the professional activities of regional IEEE groups. The region directors, among others, would serve on this Board. Currently, IEEE-USA is the major "regional group" whose mission includes influencing public policy. However, 42 percent of IEEE membership is now outside the US, and this percentage has been growing faster than membership in the US. If the trend continues, the majority of members will be outside the US in about 10 years. It is likely that other IEEE-USA-like regional groups will be formed in the future, especially in areas where no national society exists.

The Professional Activities Board's mission would include the orderly formation of regional professional activities groups and the screening of positions taken by these groups to ensure that the positions do not conflict with each other and with the IEEE's policies.

Question 4: Emerging Technologies

An increasing number of emerging technologies are interdisciplinary, and this fact conflicts with the IEEE's internal organization. Groups that represent emerging technologies are often perceived as internal competitors to long-established groups. In attempting to present an interdisciplinary face to the world, we often foster internal fragmentation that may threaten our effectiveness.

How would you integrate emerging technologies into the current IEEE structure?

How would you encourage new groups to cooperate with or be adopted by existing units?

How can the IEEE encourage and support the inclusion of new technologies without diverting support from existing entities?

Society Position

It is important for the IEEE to identify emerging technologies to keep its members informed of the latest technical developments. However, the identification of new technologies should not automatically lead to the creation of new separate entities. Truly new technologies that are not already well-covered by existing entities would justify creation of new entities. But creating new entities by carving out strong activities in existing entities is unproductive and should be discouraged. Further, expanding the number of independent groups within TAB leads to an unworkable governance structure and increased overhead charges for existing societies.

The IEEE president should lead the IEEE Executive Committee in developing effective mechanisms to determine the viability of new groups and then help viable groups integrate into existing entities. The IEEE should provide financial incentives for new groups to establish themselves as part of an existing society rather than creating a stand-alone entity. The IEEE could address new technologies by creating virtual journals in focused areas of interest that offer compilations of articles published in different journals.

Marc Apter

My many years as an IEEE volunteer have shown me that to start anything new requires a temporary organizational structure initially. After something is viable, it can either be made part of an appropriate existing organization, or it occasionally requires the formal establishment of a new organization to operate successfully. These new efforts addressing emerging technologies will require funding above our current budget levels, at least until such time as they become self-sustaining.

Pedro Ray

Identifying and adding emerging technology is key to the future of the IEEE. Our members want it and the industry wants it. How we deliver these technologies is our challenge. It is important that the IEEE be nimble in identifying new technologies, but there should be a process that must be followed before a community could be formed. Consideration must be given to existing communities as well as to financial capabilities to support these new groups before a decision is made to form a new group.

Once the decision is made to create a new emerging technology community, then we must assign the resources to make it a viable operation. Emerging technologies will expand the revenue base of the IEEE and shall share in the results.

John Vig

When multiple societies have activities in an emerging technical area, creating a council, led by a highly committed, entrepreneurial person, is often the best course of action.

For example, I proposed, and was eventually elected to be the founding president of, the IEEE Sensors Council. At the time, the IEEE had no journal and no major conference devoted to sensors. Today, the Council has >20 member societies (the Computer Society is one of them), a successful journal, and a successful conference. The *IEEE Sensors Journal* published 400 pages in 2001, its first year. In 2007, it is publishing 2,000 pages. The IEEE SENSORS conference typically attracts 600 participants.

Dealing with emerging technologies is the Technical Activities Board's responsibility. I would encourage new groups to cooperate with existing units by encouraging TAB to foster such cooperation through the formation of councils, joint publications, and joint conferences.

Question 5: Tab Structure

The IEEE Computer Society, a unit within the IEEE Technical Activities Board (TAB), is often perceived as an outlier among the entities of TAB due to its size, member profile, external competition, the constituency it serves, and other factors. The IEEE TAB structure is seen by some as a potential barrier for the IEEE Computer Society to meet its challenges to innovate, seize opportunities, manage competitive threats, and ensure long-term financial stability.

How would you address these concerns?

Would you support the creation of an alternate structure to facilitate the IEEE Computer Society in reaching its full potential?

Society Position

TAB is a large body and its decision-making process is slow and cumbersome. Because of its size, membership composition, and number of competitors, other societies with very different profiles might not understand the Computer Society's needs. This hinders the Computer Society's efforts to react quickly to leverage market opportunities and to offer innovative products. TAB's revenue and expense distribution algorithms are complicated, and the impact of formula changes is often difficult to determine. In recent years, the Computer Society has absorbed increases in direct allocations and decreases in revenue distributions. Complex and frequently changing algorithms, combined with the voting patterns of TAB, make it difficult to plan the Computer Society's finances.

The Computer Society must deal with significant opportunities and threats, and it needs to allocate staff, volunteers, and resources in a strategic way to mitigate threats and provide for its financial future. It needs a stable environment in which to retain its ability to innovate and contribute to the IEEE. We believe that this can be achieved by having a more direct relationship with the IEEE Board. To reduce complexity and increase opportunities, the IEEE Board should consider having entities that meet certain predefined criteria report directly to the Board as Major Operating Units.

Marc Apter

There is a problem with the current structure of TAB, and I don't think further patches will solve it. It is time for TAB to be replaced with a completely new structure, and the best and the brightest within TAB should meet and propose the new structure. Until this relook at TAB is complete, any change to the Computer Society's position within the IEEE structure should not be considered.

Pedro Ray

I am a strong supporter of the Computer Society becoming a Major Operating Unit (MOU). I think this has great benefit for the IEEE. As the question mentions, this will allow the largest society to compete and to reach its full potential (this is good for the IEEE). By having a seat at the IEEE Finance Committee, the operations of the Computer Society and that of the IEEE will be better integrated. For example, we could better use the CS Publication Group as a resource for the rest of the Institute (this is also good for the IEEE). Again, by having a seat at Fincom, decisions of importance could be made quicker (good for the IEEE). Eliminating the need to respond through TAB but instead responding directly to the IEEE Board of Directors makes the Computer Society more nimble (which is good for the IEEE).

John Vig

I agree that the Computer Society is an outlier in TAB. I shall be happy to work with the Computer Society leadership to address the Computer Society's concerns. I shall support reducing the affiliate dues, as I stated above. I shall support proposals aimed at making sure that the Computer Society is not "taxed" unfairly. I shall be open to Computer Society proposals aimed at reducing the Computer Society's expenses through reducing duplications of effort between it and the IEEE and through other means.

Beyond that, whether or not I would support the creation of an alternate structure will depend on the structure proposed. The successes of the Computer Society and the IEEE are intimately linked. The continued growth of IEEE must not occur at the expense of the Computer Society. I will welcome proposals that will help the Computer Society reach its full potential.

Question 6: Presidential Priorities

What are your three most important priorities for your presidential year, and how do you intend to accomplish these priorities?

Society Position

No IEEE Computer Society position.

Marc Apter

Start addressing financial issues by moving in February 2008 to freeze the total staff headcount. I would ask the 2008 president to appoint me to lead a team both to develop actions to implement the recommendations of the consultants from over four years ago and to look at our current financial formulas that have parts of the organization bleeding money when at the same time the reserves are growing in value by the many millions of dollars.

I want to ensure that our members paying full dues get benefits they can see. I want to look at membership models that will help us meet the challenges in countries where our current and potential members make salaries that are inadequate to afford our current dues structure.

I want to develop a recognition and awards program for working professionals that is similar to the current one for academics and researchers.

Pedro Ray

Membership. We need to create a membership model that addresses the needs and cost for high-grade members by repackaging our offering and changing our value proposition. For developing countries, we need an electronic membership with a low barrier to entry and a low cost of service.

Finances. For the past decade we have depended heavily on the revenue that we get from IEL, and we hope that it continues. But we are depending too much on that source of revenue. I think we need to convene a strategic meeting to find alternate sources of revenue for the IEEE.

Organization. Not all Societies have done well during the past several years, and this is due to some structural problem in our organization. I will dedicate the time required to help reorganize the IEEE to become an efficient and effective organization.

John Vig

My three highest priorities are to:

Foster experimentation and innovation. The IEEE has great volunteers, staff, products, and services, but innovation within the organization has not been among IEEE's greatest strengths. The Computer Society has been more innovative—for example, it was first in IEEE to create a digital library. I want to bring about more experimentation and innovation throughout the IEEE.

Invest more in our future. In 2005 and 2006, we added 10 times more to our reserves than we invested. Today, our reserves are >US$200 million (>US$500 per member)! So, we can definitely afford to invest more in the IEEE's future!

Improve the cost-benefit ratio of membership for everyone.

I intend to accomplish my priorities through practicing good leadership. I have demonstrated my leadership abilities via my accomplishments as conferences chair, society president, standards committee chair, VP of Technical Activities, and my service on the Board of Directors.

2008 MEMBERSHIP APPLICATION

IEEE Computer Society





FIND THE RIGHT SOLUTION!

Solve problems, learn new skills, and grow your career with the cutting edge resources of the IEEE Computer Society.

www.computer.org/join

IEEE

IEEE computer society

2008 RATES for IEEE COMPUTER SOCIETY

Membership Dues and Subscriptions

Membership and periodical subscriptions are annualized to and expire on 31 December 2008.
Pay full or half-year rate depending upon the date of receipt by the IEEE Computer Society as noted below.

Membership Options*

All prices are quoted in U.S. dollars.

	FULL YEAR Applications received 17 Aug 07 – 29 Feb 08	HALF YEAR Applications received 1 Mar 08– 15 Aug 08
I do not belong to the IEEE and I want to join only the Computer Society:	☐ $113.00	☐ $57.00
I want to join **both** the Computer Society and the IEEE:		
I reside in the USA	☐ $220.00	☐ $110.00
I reside in Canada	☐ $195.00	☐ $98.00
I reside in Africa/Europe/Middle East	☐ $187.00	☐ $94.00
I reside in Latin America	☐ $180.00	☐ $90.00
I reside in Asia/Pacific	☐ $181.00	☐ $91.00
I already belong to the IEEE, and I want to join the Computer Society:	☐ $50.00	☐ $25.00

Are you now or were you ever a member of the IEEE?
☐ Yes ☐ No If yes, please provide member # if known: ____________________

Add Periodicals**

	ISSUES PER YEAR	FULL YEAR Applications received 16 Aug 07 – 29 Feb 08 PRINT + ONLINE	HALF YEAR Applications received 1 Mar 08 – 15 Aug 08 PRINT + ONLINE
BEST VALUE!			
IEEE Computer Society Digital Library (online only)	n/a	☐ $121	☐ $61
ARTIFICIAL INTELLIGENCE			
IEEE Intelligent Systems	6	☐ $43	☐ $22
IEEE Transactions on Learning Technologies†	4	☐ $39	☐ $18
IEEE Transactions on Pattern Analysis and Machine Intelligence	12	☐ $52	☐ $26
BIOTECHNOLOGY			
IEEE/ACM Transactions on Computational Biology and Bioinformatics	4	☐ $36	☐ $18
COMPUTATION			
Computing in Science & Engineering	6	☐ $45	☐ $23
COMPUTER HARDWARE			
IEEE Computer Architecture Letters	4	☐ $29	☐ $15
IEEE Micro	6	☐ $41	☐ $21
IEEE Design & Test of Computers	6	☐ $40	☐ $20
IEEE Transactions on Computers	12	☐ $47	☐ $24
GRAPHICS & MULTIMEDIA			
IEEE Computer Graphics and Applications	6	☐ $43	☐ $22
IEEE MultiMedia	4	☐ $38	☐ $19
IEEE Transactions on Haptics†	2	☐ $31	☐ $16
IEEE Transactions on Visualization and Computer Graphics	6	☐ $43	☐ $22
HISTORY OF COMPUTING			
IEEE Annals of the History of Computing	4	☐ $34	☐ $17
INTERNET & DATA TECHNOLOGIES			
IEEE Internet Computing	6	☐ $43	☐ $22
IEEE Transactions on Knowledge and Data Engineering	12	☐ $49	☐ $25
IEEE Transactions on Services Computing†	4	☐ $39	☐ $18
IT & SECURITY			
IT Professional	6	☐ $42	☐ $21
IEEE Security & Privacy	6	☐ $24	☐ $12
IEEE Transactions on Dependable and Secure Computing	4	☐ $33	☐ $17
MOBILE COMPUTING			
IEEE Pervasive Computing	4	☐ $43	☐ $22
IEEE Transactions on Mobile Computing	12	☐ $43	☐ $22
NETWORKING			
IEEE Transactions on Parallel and Distributed Systems	12	☐ $47	☐ $24
SOFTWARE			
IEEE Software	6	☐ $49	☐ $25
IEEE Transactions on Software Engineering	12	☐ $38	☐ $19

* Member dues include $25 for a 12-month subscription to *Computer* magazine.
** Periodicals purchased at member pirces are for the member's personal use only.
† Online issues only.

Payment Information

Payment required with application

Membership fee
$ ____________

Periodicals total
$ ____________

Applicable sales tax***
$ ____________

TOTAL:
$ ____________

Enclosed:
☐ Check/Money Order****

Charge my:
☐ MasterCard
☐ VISA
☐ American Express
☐ Diner's Club

____
Card Number

____
Exp Date (month/year)

____
Signature

USA Only include
5-digit billing zip code
☐☐☐☐☐

* Member dues include $25 for a 12-month subscription to *Computer.*

** Periodicals purchased at member prices are for the member's personal use only.

*** Canadian residents add 14% HST or 6% GST to total. AL, AZ, CO, DC, GA, IN, KY, MD, MO, NM, and WV add sales tax to periodical subscriptions. European Union residents add VAT tax to IEEE Computer Society Digital Library subscription.

**** Payable to the IEEE in U.S. dollars drawn on a U.S. bank account. Please include member name and number (if known) on your check.

Allow up to 8 weeks for application processing. Allow a minimum of 6 to 10 weeks for delivery of print periodicals.

Please complete both sides of this form.

For fastest service, apply online at www.computer.org/join

Personal Information

Enter your name as you want it to appear on correspondence. As a key identifier in our database, circle your last/surname.

☐ Male ☐ Female

Date of birth (Day/Month/Year)

Title | First name | Middle

Last/Surname

Home address

City | State/Province

Postal code | Country

Home telephone

Home facsimile

Preferred e-mail

Send mail to: ☐ Home address ☐ Business address

Educational Information

First professional degree completed | Month/Year degree received

Program major/course of study

College/University | State/Province | Country

Highest technical degree received | Program/Course of study

Month/Year received

College/University | State/Province | Country

Business/Professional Information

Title/Position

Years in current position | Years of practice since graduation

Employer name

Department/Division

Street address

City | State/Province

Postal code | Country

Office phone

Office facsimile

I hereby make application for Computer Society and/or IEEE membership and agree to be governed by IEEE's Constitution, Bylaws, Statements of Policies and Procedures, and Code of Ethics. I authorize release of information related to this application to determine my qualifications for membership.

Signature | Date

IF8I

NOTE: In order for us to process your application, you must complete and return BOTH sides of this form to the office nearest you:

Asia/Pacific Office
IEEE Computer Society
Watanabe Bldg.
1-4-2 Minami-Aoyama
Minato-ku, Tokyo 107-0062 Japan
Phone: +81 3 3408 3118
Fax: +81 3 3408 3553
E-mail: tokyo.ofc@computer.org

Publications Office
IEEE Computer Society
10662 Los Vaqueros Circle
P.O. Box 3014
Los Alamitos, CA 90720-1314 USA
Phone: +1 800 272 6657 (USA and Canada)
Phone: +1 714 821 8380 (worldwide)
Fax: +1 714 821 4641
E-mail: help@computer.org

BPA Information

This information is used by society magazines to verify their annual circulation. Please refer to the audit codes and indicate your selections in the box provided.

A. Primary line of business

1. Computers
2. Computer peripheral equipment
3. Software
4. Office and business machines
5. Test, measurement, and instrumentation equipment
6. Communications systems and equipment
7. Navigation and guidance systems and equipment
8. Consumer electronics/appliances
9. Industrial equipment, controls, and systems
10. ICs and microprocessors
11. Semiconductors, components, sub-assemblies, materials, and supplies
12. Aircraft, missiles, space, and ground support equipment
13. Oceanography and support equipment
14. Medical electronic equipment
15. OEM incorporating electronics in their end product (not elsewhere classified)
16. Independent and university research, test and design laboratories, and consultants (not connected with a manufacturing company)
17. Government agencies and armed forces
18. Companies using and/or incorporating any electronic products in their manufacturing, processing, research, or development activities
19. Telecommunications services, and telephone (including cellular)
20. Broadcast services (TV, cable, radio)
21. Transportation services (airlines, railroads, etc.)
22. Computer and communications and data processing services
23. Power production, generation, transmission, and distribution
24. Other commercial users of electrical, electronic equipment, and services (not elsewhere classified)
25. Distributor (reseller, wholesaler, retailer)
26. University, college/other education institutions, libraries
27. Retired
28. Others (allied to this field) ______________________

B. Principal job function

1. General and corporate management
2. Engineering management
3. Project engineering management
4. Research and development management
5. Design engineering management — analog
6. Design engineering management — digital
7. Research and development engineering
8. Design/development engineering — analog
9. Design/development engineering — digital
10. Hardware engineering
11. Software design/development
12. Computer science
13. Science/physics/mathematics
14. Engineering (not elsewhere classified)
15. Marketing/sales/purchasing
16. Consulting
17. Education/teaching
18. Retired
19. Other ______________________

C. Principal responsibility

1. Engineering or scientific management
2. Management other than engineering
3. Engineering design
4. Engineering
5. Software: science/management/engineering
6. Education/teaching
7. Consulting
8. Retired
9. Other ______________________

D. Title

1. Chairman of the Board/President/CEO
2. Owner/Partner
3. General Manager
4. V.P. Operations
5. V.P. Engineering/Director Engineering
6. Chief Engineer/Chief Scientist
7. Engineering Manager
8. Scientific Manager
9. Member of Technical Staff
10. Design Engineering Manager
11. Design Engineer
12. Hardware Engineer
13. Software Engineer
14. Computer Scientist
15. Dean/Professor/Instructor
16. Consultant
17. Retired
18. Other Professional/Technical ______________________

CALLS FOR ARTICLES FOR *COMPUTER*

Computer seeks articles for an April 2008 special issue on data-intensive computing. Guest editors are Ian Gorton of the Pacific Northwest National Laboratory, Paul Greenfield of Australia's Commonwealth Scientific and Industrial Research Organization, Alex Szalay of Johns Hopkins University, and Roy Williams of the California Institute of Technology.

The need to capture, store, and provide timely processing on ever-increasing data volumes exists in a diverse range of application domains. These include scientific computing, Internet-based social computing, Internet search, bioinformatics, enterprise applications, video surveillance, and cybersecurity.

Many existing technologies fail when data volumes reach multiterabyte and petabyte ranges, requiring new approaches to solving such data-intensive computing problems. Innovations in data-intensive computing comprise both processor hardware and system and application software that can scale to handle ballooning data volumes and simultaneously reduce the time needed to provide useful data analysis.

Computer's special issue on data-intensive computing will bring together contributions from leading scientists that address current and future problems and describe promising solutions in data-intensive computing. The articles will represent the state of the art in research and development.

Computer invites papers that provide a perspective on all facets of data-intensive computing. Examples of suitable topics include scalable algorithms to search and process massive data sets; advances in high-performance computing platforms; high-performance, high-reliability, petascale distributed file systems and databases; and techniques for processing massive data volumes from sensor networks.

Also welcome are case studies of solutions to data-intensive computing problems as well as surveys of current and future problems in data-intensive application domains such as astronomy, social computing, biology, environmental sciences, and the intelligence community.

The deadline for papers is **15 October**. Detailed author instructions are available at www.computer.org/portal/pages/computer/mc/author.html. Send inquiries to the guest editors at ian.gorton@pnl.gov, paul.greenfield@csiro.au, szalay@jhu.edu, or roy@cacr.caltech.edu.

Computer seeks articles for a July 2008 special issue on high-assurance service-oriented architectures. The guest editors are Jing Dong from the University of Texas at Dallas, Raymond Paul from the US Department of Defense, and Liang-Jie Zhang from IBM's T.J. Watson Research Center.

Recent advances in services computing technology allow users to register, request, discover, and supply software services online. Such loosely coupled software services form a service-oriented architecture with the support of resources on a network. Service-oriented architectures have been applied in many mission-critical environments, including medical and healthcare systems, traffic-control applications, and defense systems. These systems are required to be highly reliable, secure, available, timely, fault-tolerant, and dependable. Recently, architects have faced new challenges in developing service-oriented systems with high-assurance requirements.

Computer invites papers that describe techniques, tools, or experiences related to the design, development, or assessment of practical high-assurance systems. The issue editors are particularly interested in submissions that address applications of service-oriented techniques. Examples of suitable topics include high-assurance service compositions; service specifications for security, reliability, dependability, availability, and QoS properties; service discoveries with high-assurance system requirements; and service security, trust, and privacy.

The deadline for papers is **14 December**. Detailed author instructions are available at www.computer.org/portal/pages/computer/mc/author.html. Send inquiries to the guest editors at jdong@utdallas.edu, raymond.paul@osd.mil, or zhanglj@us.ibm.com.

CALLS FOR PAPERS

WICSA 2008, Working IEEE/IFIP Conf. on Software Architecture, 18-21 Feb., Vancouver, Canada; Submissions due **17 Sept.** www.wicsa.net

EDCC 2008, 7th European Dependable Computing Conf., 7-9 May, Kaunas, Lithuania; Submissions due **20 Sept.** http://edcc.dependability.org/call-for-contributions/call-for-papers

IPDPS 2008, IEEE Int'l Parallel & Distributed Processing Symp., 14-18 April, Miami; Submissions due **8 Oct.** www.ipdps.org/ipdps2008/2008_cfp.html

Submission Instructions

The Call and Calendar section lists conferences, symposia, and workshops that the IEEE Computer Society sponsors or cooperates in presenting.

Visit www.computer.org/conferences for instructions on how to submit conference or call listings as well as a more complete listing of upcoming computer-related conferences.

CALENDAR

OCTOBER 2007

2-5 Oct: ICSM 2007, 23rd IEEE Int'l Conf. on Software Maintenance, Paris; http://icsm07.ai.univ-paris8.fr

3-5 Oct: CRITIS 2007, 2nd Int'l Workshop on Critical Information Infrastructures Security, Malaga, Spain; http://critis07.lcc.uma.es

8-10 Oct: WiMob 2007, 3rd IEEE Int'l Conf. on Wireless & Mobile Computing, Networking, & Comm., White Plains, N.Y.; www.gel.usherbrooke.ca/WiMob2007

9-11 Oct: ATS 2007, 16th Asian Test Symp., Beijing; http://ats07.ict.ac.cn

10-12 Oct: AIPR 2007, Applied Imagery Pattern Recognition, Washington, D.C.; www.aipr-workshop.org

10-12 Oct: Tabletop 2007, 2nd IEEE Int'l Workshop on Horizontal Interactive Human-Computer Systems, Newport, R.I.; www.ieeetabletop2007.org

10-13 Oct: FIE 2007, Frontiers in Education Conf., Milwaukee, Wis.; www.fie-conference.org/fie07

11-13 Oct: IPC 2007, Int'l Conf. on Intelligent Pervasive Computing, Jeju, Korea; www.sersc.org/IPC2007

11-13 Oct: ISWC 2007, Int'l Symp. on Wearable Computers, Boston; http://iswc.net

12-13 Oct: WRTLT 2007, 8th Workshop on RTL & High-Level Testing (with ATS), Beijing; http://wrtlt07.ict.ac.cn

14-17 Oct: Tapia 2007, Richard Tapia Celebration of Diversity in Computing Conference, Orlando, Florida; www.richardtapia.org/2007

14-21 Oct: ICCV 2007, 11th IEEE Int'l Conf. on Computer Vision, Rio de Janeiro; www.research.rutgers.edu/~iccv2007

15 Oct: MWS 2007, Middleware for Web Services Workshop (with EDOC), Annapolis, Md.; www.green-pea.net/mws

15-17 Oct: BIBE 2007, IEEE 7th Symp. on Bioinformatics & Bioengineering, Boston; www.cs.gsu.edu/BIBE07

15-18 Oct: LCN 2007, 32nd IEEE Conf. on Local Computer Networks, Dublin, Ireland; www.ieeelcn.org

15-18 Oct: LCN ON-MOVE 2007, IEEE Workshop On User Mobility and Vehicular Networks (with LCN), Dublin, Ireland; www.ieee-lcn-onmove.org

15-19 Oct: EDOC 2007, 11th IEEE Enterprise Distributed Object Computing Conf., Annapolis, Md.; http://edoc.mitre.org

15-19 Oct: RE 2007, 15th IEEE Int'l Requirements Eng. Conf., Delhi, India; www.re07.org

16-19 Oct: CIT 2007, IEEE 7th Int'l Conf. on Computer and Information Technology, Aizu, Japan; www.u-aizu.ac.jp/conference/cit07

Events in 2007

OCTOBER 2007

- 2-5 . . . . ICSM 2007
- 3-5 . . . . CRITIS 2007
- 8-10 . . . . WiMob 2007
- 9-11 . . . . ATS 2007
- 10-12 . . . . AIPR 2007
- 10-12 . . . . Tabletop 2007
- 10-13 . . . . FIE 2007
- 11-13 . . . . IPC 2007
- 11-13 . . . . ISWC 2007
- 12-13 . . . . WRTLT 2007
- 14-17 . . . . Tapia 2007
- 14-21 . . . . ICCV 2007
- 15 . . . . MWS 2007
- 15-17 . . . . BIBE 2007
- 15-18 . . . . LCN 2007
- 15-18 . . . . LCN ON-MOVE 2007
- 15-19 . . . . EDOC 2007
- 15-19 . . . . RE 2007
- 16-19 . . . . CIT 2007
- 20-23 . . . . FOCS 2007
- 24-27 . . . . SBAC-PAD 2007
- 27 Oct-1 Nov . . . . Vis 2007
- 28-31 . . . . ICDM 2007
- 28-31 . . . . WCRE 2007
- 29-31 . . . . ICTAI 2007
- 29-31 . . . . SKG 2007

NOVEMBER 2007

- 2-5 . . . . WI-IAT 2007
- 7-9 . . . . HLDVT 2007

DECEMBER 2007

- 5-7 . . . . APSEC 2007
- 10-12 . . . . ISM 2007
- 10-13 . . . . eScience 2007
- 11-14 . . . . APSCC 2007
- 17-20 . . . . ICIT 2007

20-23 Oct: FOCS 2007, IEEE 48th Ann. Symp. on Foundations of Computer Science, Providence, R.I.; www.focs2007.org

24-27 Oct: SBAC-PAD 2007, 19th Int'l Symp. on Computer Architecture and High-Performance Computing, Gramado, Brazil; www.sbc.org.br/sbac/2007

27 Oct-1 Nov: Vis 2007, IEEE Visualization, Sacramento, Calif.; http://vis.computer.org/vis2007

28-31 Oct: ICDM 2007, 7th IEEE Int'l Conf. on Data Mining, Omaha, Neb.; www.cs.uvm.edu/~icdm

28-31 Oct: WCRE 2007, 14th Working Conf. on Reverse Engineering, Vancouver, Canada; www.rcost.unisannio.it/ wcre2007

29-31 Oct: ICTAI 2007, 7th IEEE Int'l Conf. on Tools with Artificial Intelligence, Patras, Greece; http://ictai07.ceid.upatras.gr

29-31 Oct: SKG 2007, 3rd Int'l Conf. on Semantics, Knowledge, and Grid, Xi'an, China; www.culturegrid.net/SKG2007

NOVEMBER 2007

2-5 Nov: WI-IAT 2007, IEEE/WIC/ACM Int'l Conf. on Web Intelligence & IEEE/WIC/ACM Int'l Conf. on Intelligent Agent Technology, Fremont, Calif.; www.cs.sjsu.edu/wi07/wi

7-9 Nov: HLDVT 2007, IEEE Int'l High-Level Design Validation and Test Workshop, Irvine, Calif.; http://hldvt.com/07

DECEMBER 2007

5-7 Dec: APSEC 2007, 14th Asia-Pacific Software Engineering Conf., Nagoya, Japan; http://apsec2007.fuka.info.waseda.ac.jp

10-12 Dec: ISM 2007, IEEE Int'l Symp. on Multimedia, Taichung, Taiwan; http://ism2007.ncu.edu.tw

10-13 Dec: eScience 2007, 3rd IEEE Int'l Conf. on e-Science and Grid Computing, Bangalore, India; www.escience2007.org

11-14 Dec: APSCC 2007, IEEE Asia-Pacific Services Computing Conf., Tsukuba, Japan; http://conferences.computer.org/apscc/2007

17-20 Dec: ICIT 2007, 10th Int'l Conf. on Information Technology, Rourkela, India; http://icit2007.home.comcast.net

JANUARY 2008

7-10 Jan: HICSS 2008, Hawai'i Int'l Conf. on System Sciences, Waikoloa, Hawai'i; www.hicss.hawaii.edu/hicss_41/apahome41.html

FEBRUARY 2008

18-21 Feb: WICSA 2008, Working IEEE/IFIP Conf. on Software Architecture, Vancouver, Canada; www.wicsa.net

MARCH 2008

18-21 Feb: SimuTools 2008, 1st Int'l Conf. on Simulation Tools and Techniques for Communications, Networks, and Systems, Vancouver, Canada; www.simutools.org

APRIL 2008

7-12 Apr: ICDE 2008, 24th IEEE Int'l Conf. on Data Engineering, Cancun, Mexico; www.icde2008.org

14-15 Apr: RAW 2008, 15th Reconfigurable Architectures Workshop (with IPDPS), Miami; www.ece.lsu.edu/vaidy/raw

14-18 Apr: IPDPS 2008, 22nd IEEE Int'l Parallel and Distributed Processing Symp., Miami; www.ipdps.org

2007 IEEE International Symposium on Multimedia

ISM 2007 is an international forum where researchers come together to share recent advances in the state of the art and practice of multimedia computing.

Organizers have solicited submissions that identify emerging research topics and define the future of multimedia computing. Examples of possible topics include pervasive and interactive multimedia systems, visualization and virtual reality, and software development using multimedia techniques.

The technical program at ISM 2007 will consist of invited talks, paper presentations, and panel discussions.

The conference will take place in Taichung, Taiwan, **10-12 December**. Visit http://ism2007.ncu.edu.tw for complete details on ISM 2007, including registration information as it becomes available.

TU WIEN

Announcement of an open position at the Faculty of Informatics, Vienna University of Technology, Austria

Full Professor (tenured) in Design and Assessment of Technology

The faculty is looking for candidates that bring expertise in new areas of Human Computer Interaction (HCI), in particular mobile and ubiquitous interaction, as well as novel interfaces and interaction design (e. g. ambient displays, interactive sonification, multi-touch interfaces), with a focus on innovative concepts and on design and evaluation methods. Application areas of interest are, for example, culture, creativity, learning, communities, and sustainability.

Applicants are expected to practice a multi-disciplinary approach grounded in participatory design, combining competence in informatics with expertise in the social sciences. Applicants should have an outstanding academic record (habilitation or a comparable scientific accomplishment), experience in project acquisition and management at the national and international level, and experience with supervising thesis work.

A more detailed announcement and information on how to apply can be found at **http://www.informatik.tuwien.ac.at/DAT.pdf**

Application deadline: **November 15, 2007**

Philips has the following job opportunities available (various levels/types):

PHILIPS ELECTRONICS NORTH AMERICA

ATLANTA, GA
- Applications Developer Manager (ADM-PENAC-GA)

BROOKLYN, NY
- Programmer Analyst (PA-PENAC-NY)

LATHAM, NY
- Mechanical Engineer (ME-PENAC-NY)

GAINESVILLE, FL
- Design Engineer (DE-PENAC-FL)

PHILIPS NUCLEAR MEDICINE

MILPITAS, CA
- Software Engineer (SWE-PNM-CA)

PHILIPS MEDICAL SYSTEMS CLEVELAND

AURORA, IL
- Design Engineer (DE-PMSC-IL)

PHILIPS ULTRASOUND

ANDOVER, MA
- Product Manager (PM-PU-MA)

Some positions may require travel. Submit resume by mail to PO box 4104, Santa Clara, CA 95056-4104, ATTN: HR Coordinator. **Must reference job title and job code (i.e. SWE-PENAC-CA) in order to be considered.** EOE.

PROGRAMMER/ANALYST (Pepper Pike, OH) Dsgn for data retrieval & mgmt systems to meet user needs. Support applic s/ware incl but not ltd to Jenzabar EX & JICS. Must have BS in Comp Sci. No exp reqd. 40hr/wk 8a-5p, M-F. Send resumes (no calls) to HR Dept, Ursuline College, 2550 Lander Rd, Cleveland, OH 44124.

UNIVERSITY OF ALABAMA IN HUNTSVILLE. The Computer Science Department of the University of Alabama in Huntsville (UAH) invites applicants for two tenure-track faculty positions at the Assistant Professor level. A Ph.D. in Computer Science or closely related area is required. Though Bioinformatics is our primary area of interest, exceptional applicants specialized in Computer Networks, Distributed Processing, and related areas are encouraged to apply. We have a strong commitment to excellence in teaching, research and service. Applicants should have the ability to perform research and have good communication and teaching ability. UAH is strategically located along the Tennessee River in scenic North Alabama in a rapidly expanding high technology area. Huntsville has one of the largest Research Parks in the nation and is home to NASA's Marshall Space Flight Center, the Army's Redstone Arsenal, and many high-tech industries that present many opportunities to pursue collaborative research. The surrounding population of approximately 365,000 well-educated and highly technically skilled people has access to excellent public schools and inexpensive housing. The University has an enrollment of approximately 7700 students with research centers in the areas of Information Technology, Robotics, Optics, Microgravity, Space Science, and Materials Science that provide further opportunities for collaborative projects. The Computer Science Department has 14 full-time faculty and offers B.S., M.S., and Ph.D. degrees in Computer Science and M.S. degree in Software Engineering. There are approximately 300 undergraduate CS majors, 170 MS, and 53 PhD. candidates enrolled in our program. Current faculty research interests include Software Engineering, Visualization, Neural Networks, Pattern Recognition, Image Processing, Distributed Systems, Graph Theory, High Performance Computing, Data Mining, Information Technology, Artificial Intelligence, Modeling and Simulation, and Networking. According to recent NSF figures, the CS department ranks 30th in the nation in federal research funding. Please submit a detailed resume with references to Chair, Search Committee, Computer Science Department, UAH, Huntsville, AL, 35899. Qualified women and minority candidates are encouraged to apply. Initial review of applicants will begin in November 2007 and will continue until a suitable candidate is found. The University of Alabama in Huntsville is an equal Opportunity/Affirmative Action Institution.

TECH. CONSULT., Ascendant Technology, Austin, TX. Req.: BS or for. equiv. in Comp. Sci. or rel. + 2 yrs. exp. w/ WebSphere, Java, J2EE. Resume only attn.: C. Jones (File #061034) 10215 161st Pl. NE Redmond, WA 98052.

HEWLETT - PACKARD COMPANY is accepting resumes for the position of Technical Analyst in Austin, TX. (Reference # AUSBAS). Please send resumes with reference number to: Hewlett-Packard Company, 19483 Pruneridge Avenue, Mail Stop 4206, Cupertino, California 95014. No phone calls please. Must be legally authorized to work in the U.S. without sponsorship. EOE.

SUBMISSION DETAILS: Rates are $299.00 per column inch ($320 minimum). Eight lines per column inch and average five typeset words per line. Send copy at least one month prior to publication date to: Marian Anderson, Classified Advertising, *Computer* Magazine, 10662 Los Vaqueros Circle, PO Box 3014, Los Alamitos, CA 90720-1314; (714) 821-8380; fax (714) 821-4010. *Email: manderson@computer.org.*

In order to conform to the Age Discrimination in Employment Act and to discourage age discrimination, *Computer* may reject any advertisement containing any of these phrases or similar ones: "…recent college grads…," "…1-4 years maximum experience…," "…up to 5 years experience," or "…10 years maximum experience." *Computer* reserves the right to append to any advertisement without specific notice to the advertiser. Experience ranges are suggested minimum requirements, not maximums. *Computer* assumes that since advertisers have been notified of this policy in advance, they agree that any experience requirements, whether stated as ranges or otherwise, will be construed by the reader as minimum requirements only. *Computer* encourages employers to offer salaries that are competitive, but occasionally a salary may be offered that is significantly below currently acceptable levels. In such cases the reader may wish to inquire of the employer whether extenuating circumstances apply.

香 港 大 學
THE UNIVERSITY OF HONG KONG

Centenary Recruitment Plan

Founded in 1911, The University of Hong Kong is committed to the highest international standards of excellence in teaching and research, and has been at the international forefront of academic scholarship for many years. Of a number of recent indicators of the University's performance, one is its ranking at 33 among the top 200 universities in the world by the UK's Times Higher Education Supplement. The University has a comprehensive range of study programmes and research disciplines, with 20,000 undergraduate and postgraduate students from 50 countries, and a complement of 1,200 academic members of staff, many of whom are internationally renowned.

As the University approaches its 100th anniversary, a major human resource expansion plan has been launched to provide 200 new academic positions. The purpose of this Centenary Recruitment Plan is to enhance our research competitiveness and to facilitate the introduction and delivery of a new four-year undergraduate curriculum from 2012.

Building on Hong Kong's international status and its mission to serve China, the University offers an intellectually-stimulating and culturally-rich academic environment, with attractive remuneration packages.

Department of Computer Science

Applications are invited for appointments as **(1) Chair/Professor (Ref.: RF-2007/2008-20)** and **(2) Associate Professor/Assistant Professor (Ref.: RF-2007/2008-21)** in the Department of Computer Science, tenable from September 1, 2008. The appointments will initially be made on a three-year fixed-term basis, with consideration for tenure after satisfactory completion of a second fixed-term contract.

The Department offers programmes at both undergraduate and postgraduate levels, and has excellent computing resources, well-equipped teaching and research facilities and support. Information about the Department can be obtained at http://www.cs.hku.hk/.

Applicants should have a Ph.D. degree in Computer Science, Computer Engineering, or related fields, and a strong interest in research and teaching. A solid track record in research is essential.

For post (1), applicants should have an exceptional record of research that aligns with the University's initiatives in information technology and bioinformatics. The Chair Professor/Professor will be expected to maintain innovative research activities that will attract outstanding students and faculty; to provide leadership for Department's research activities; and to acquire external funding.

For post (2), applicants should be active and committed to research in bioinformatics, theoretical computer science, data engineering, computer graphics and vision, and systems or computer security. Applicants in other research areas will also be considered exceptionally.

Applicants should indicate clearly which post, field and level (preferably with reference number) they wish to be considered for. Those who have responded to the previous exercises (Ref.: RF-2006/2007-293-294) need not re-apply, as they will be reconsidered together with the new applicants.

Annual salaries will be in the following ranges (subject to review from time to time at the entire discretion of the University):

Chair	:	Professorial range, which is not less than HK$1.1M
Professor	:	HK$803,700 - 1,125,720
Associate Professor	:	HK$593,100 - 917,220
Assistant Professor	:	HK$451,980 - 698,520

(approximately US$1 = HK$7.8)

The appointments will attract a contract-end gratuity and University contribution to a retirement benefits scheme, totalling up to 15% of basic salary, as well as leave, and medical/dental benefits. Housing benefits will also be provided. At current rates, salaries tax does not exceed 16% of gross income.

Further particulars and application forms (272/302 amended) can be obtained at https://www.hku.hk/apptunit/; or from the Appointments Unit (Senior), Registry, The University of Hong Kong, Hong Kong (Fax: (852) 2540 6735 or 2559 2058; E-mail: senrappt@hkucc.hku.hk). **Applications will be accepted until the positions are filled.** Candidates who are not contacted within 4 months may consider their applications unsuccessful.

The University is an equal opportunity employer

The Institute for Defense Analyses Center for Computing Sciences

Is looking for outstanding Ph.D. level scientists, mathematicians and engineers to address problems in high-performance computing, cryptography and network security. IDA/CCS is an independent research center sponsored by the National Security Agency. IDA/CCS scientists and engineers work on difficult scientific problems, problems vital to the nation's security. Stable funding provides for a vibrant research environment and an atmosphere of intellectual inquiry free of administrative burdens.

Research at IDA/CCS emphasizes computer science, computer architecture, electrical engineering, information theory and all branches of mathematics. Because CCS research staff work on complex topics often engaging multidisciplinary teams, candidates should demonstrate depth in a particular field as well as a broad understanding of computational science and technology.

Developing imaginative computational solutions employing novel digital technology is one of several long-term themes of work at CCS. The Center is equipped with a very large variety of hardware and software. The latest developments in high-end computing are heavily used and projects routinely challenge the capability of the most advanced architectures.

IDA/CCS offers a competitive salary, an excellent benefits package and a superior professional working environment. IDA/CCS is located in a modern research park in the Maryland suburbs of Washington, DC.

U.S. citizenship and a DoD TS//SI clearance are required. CCS will sponsor this clearance for those selected.

The Institute for Defense Analyses is proud to be an equal opportunity employer.

Please send responses or inquiries to:
Dr. Francis Sullivan
Director
IDA Center for Computing Sciences
17100 Science Drive
Bowie, MD 20715-4300
Phone: (301) 805-7534 Fax: (301) 805-7602
fran@super.org

IDA

THE GEORGE WASHINGTON UNIVERSITY
WASHINGTON DC

The Department of Electrical and Computer Engineering at The George Washington University invites applications for tenure-track, tenured and contractual non-tenure-accruing faculty positions at all ranks, in the area of Computer Engineering. Two positions will be for tenure-track/tenured faculty, and the third position will be a one-year renewable non-tenure-accruing contractual position at the Assistant/Associate Professor rank, and successful candidates may start as early as Fall 2007. Faculty with research in High-Performance Computing and Reconfigurable Computing are particularly encouraged to apply, however, all areas of Computer Engineering will be considered. Additional information and details on position qualifications and the application procedure are available on http://www.ece.gwu.edu/. Review of applications will continue until the positions are filled. The George Washington University is an Equal Opportunity/Affirmative Action Employer.

HEWLETT - PACKARD COMPANY is accepting resumes for the position of Software Designer in San Francisco, CA. (Reference # SFAKU). Please send resumes with reference number to: Hewlett-Packard Company, 19483 Pruneridge Avenue, Mail Stop 4206, Cupertino, California 95014. No phone calls please. Must be legally authorized to work in the U.S. without sponsorship. EOE.

SONOMA STATE UNIVERSITY, Assistant Professor of Computer Science to begin in August, 2008. Candidates must have a PhD in Computer Science or equivalent field by the hiring date. Sonoma State is part of the California State University system and is located just north of San Francisco. A detailed description of the position can be found at http://www.sonoma.edu/FacAffairs. Deadline is October 12, 2007.

HEWLETT - PACKARD COMPANY is accepting resumes for the following positions in Cupertino, CA. Software Engineer (Reference #CUPVKO). Product Marketing Manager (Reference #CUPGGA). Information Technology Technical Analyst (Reference # CUPYLI). Please send resumes with reference number to: Hewlett-Packard Company, 19483 Pruneridge Avenue, Mail Stop 4206, Cupertino, California 95014. No phone calls please. Must be legally authorized to work in the U.S. without sponsorship. EOE.

ARCHITECT, Ascendant Technology, Austin, TX: Design/Devel. J2EE-based apps. Req.: MS or for. equiv. in Comp. Sci. or rel. Resume only attn.: C. Jones (File #07-1133) 10215 161st Pl. NE Redmond, WA 98052.

HEWLETT - PACKARD COMPANY is accepting resumes for the position of ITO Consultant in Boise, ID. Job duties to include architectural design or engineering, integration, and deployment of SANs and BC Solutions for Core Infrastructure Services (Reference # BOIUAM). Please send resumes with reference number to Hewlett-Packard Company, 19483 Pruneridge Avenue, Mail Stop 4206, Cupertino, California 95014. No phone calls please. Must be legally authorized to work in the U.S. without sponsorship. EOE.

HEWLETT - PACKARD COMPANY is accepting resumes for the position of Technical Analyst (IT) in Roseville, CA. (Ref # ROSBSU). Please send resumes with reference number to Hewlett-Packard Company, 19483 Pruneridge Avenue, Mail Stop 4206, Cupertino, California 95014. No phone calls please. Must be legally authorized to work in the U.S. without sponsorship. EOE.

HEWLETT - PACKARD COMPANY is accepting resumes for the following positions in San Diego, California. Engineering Project Manager (Reference # SDVYE). Hardware Development Engineer (Reference # SDNTO). Please send resumes with reference number to: Hewlett-Packard Company, 19483 Pruneridge Avenue, Mail Stop 4206, Cupertino, California 95014. No phone calls please. Must be legally authorized to work in the U.S. without sponsorship. EOE.

HEWLETT - PACKARD COMPANY is accepting resumes for the position of Business Intelligence/Market Research Manager in Vancouver, WA. Job duties to include understanding business and competitive insight for competitors operating in the printing market in North America. (Reference # VANPGI). Please send resumes with reference number to: Hewlett-Packard Company, 19483 Pruneridge Avenue, Mail Stop 4206, Cupertino, California 95014. No phone calls please. Must be legally authorized to work in the U.S. without sponsorship. EOE.

PENN STATE UNIVERSITY. Penn State Greater Allegheny. Computer Science (Assistant Professor, tenure-track). Start August 2008. Teach beginning computer science, mathematics, and related courses. Publish in refereed journals. Participate in campus, university, and community service activities. Ph.D. in Computer Science or related discipline required. To learn more about the campus, visit http://www.psu.edu/ur/cmpcoll.html. To learn more about the position and how to apply, visit http://www.psu.jobs/Opportunities/Opportunities.html and follow the "Faculty" link. AA/EOE.

EMC CORPORATION has the following job opportunities available (various levels/types) in Hopkinton, MA (1), Cambridge, MA (2), New York, NY (3), Santa Clara, CA (4), Pleasanton, CA (5), Research Triangle Park, NC (6), Nashua, NH (7), Alexandria, VA (8), and White Plains, NY (9): **Architects** (A1, A2, A3, A4, A5, A6, A7, A8, A9), **Engineering Managers** (EM1, EM2, EM3, EM4, EM5, EM6, EM7, EM8, EM9), **Project Managers** (PM1, PM2, PM3, PM4, PM5, PM6, PM7, PM8, PM9), **Product Managers** (PDM1, PDM2, PDM3, PDM4, PDM5, PDM6, PDM 7, PDM8, PDM9), **QA Engineers** (QAE1, QAE2, QAE3, QAE4, QAE5, QAE6, QAE7, QAE8, QAE9), **Sales Engineers** (SE1, SE2, SE3, SE4, SE5, SE6, SE7, SE8, SE9), **Software Engineers** (SWE1, SWE2, SWE3, SWE4, SWE5, SWE6, SWE7, SWE8, SWE9), **Software Engineers (Consultant)** (SEC1, SEC2, SEC3, SEC4, SEC5, SEC6, SEC7, SEC8, SEC9)*, **Support Engineers** (DSE1, DSE2, DSE3, DSE4, DSE5, DSE6, DSE7, DSE8, DSE9), **Systems Engineers** (SSE1, SSE2, SSE3, SSE4, SSE5, SSE6, SSE7, SSE8, SSE9), *Some positions may require travel. Send resume to: 4400 Computer Drive, Westboro, MA 01580. Must reference job title and job code (i.e. A1) to be considered. EOE.
www.emc.com

APPLICATION SUPPORT ENGINEER. Support Fiorano / Servicemix ESB processes, HireRight apps, DB, & HR systems; F5 load balancer admin/config. Qualifications: Must know HireRight system and sw design process, apps from all user perspectives and integration with our partners and vendors. Knowledge of Linux, J2EE (App servers, JMS, JNDI), in-depth SQL & Oracle PL/SQL, Nagios, RRD, XML, SOAP, WSDL, UDDI.Knowledge of Russian and Estonian languages is a plus. Requirements: Be able to provide 24/7 support coverage. Strong teamwork skills, ability to interface with engineering, business managers, our of shore Dev teams in Estonia/Russia after hours and emergencies. Knowledge of Software QA and Change Control processes and procedures. HireRight, Inc. 5151 California Ave, Irvine, CA 92617. Please apply at http://www.hireright.com. EOE/FIMIDIV

PURCHASING AGENT, TECHNOLOGY. Provide general IT & telecom eqpmt procurement & inventory tracking. Reqs Bachelors in Comp Sci. Send resumes to American International Investors Administration, LLC, 100 SE 2nd St, 47th Flr, Miami, FL 33131.

HEWLETT - PACKARD COMPANY is accepting resumes for the following positions in Houston, TX. Technical Analyst (Reference # HOUPMO). Lead Technical Analyst (Reference # HOUSGO). Please send resumes with reference number to: Hewlett-Packard Company, 19483 Pruneridge Avenue, Mail Stop 4206, Cupertino, California 95014. No phone calls please. Must be legally authorized to work in the U.S. without sponsorship. EOE.

PROGRAMMER ANALYST: Vignette Subject Matter Expert who can analyze, dsgn, dvlp, test & implmt content mgmt applics, web based applics & s/ware using Vignette 7, Java, Oracle 9i, MS SQL server 2000/2005, XML, WebSphere, Korn Shell Scripts, HTML, DHTML, JSP, JDBC, Unix & Windows 98/00/NT. Provide tech assistance in connection w/systm maintenance. Prep tech documentation for user ref.; conduct training sessions for end users. Reqs MS in Comp Sci, Info Systms.

Mail resumes to Proteam LLC 10000 N 31st Ave, Ste C309, Phoenix, AZ 85051

HEWLETT - PACKARD COMPANY is accepting resumes for the following positions headquartered in Palo Alto, CA: Business Consultant (Ref# KBBC); Customer Project/Program Manager (Ref# KBPM); Information Systems Architect (Ref# KBISA); Technology Consulting (Ref# KBTC). Please send resumes with Ref# to: Hewlett-Packard Company, Attn: D. Slusher, 19483 Pruneridge Avenue, Mail Stop 4206, Cupertino, California 95014. No phone calls please. Must be legally authorized to work in the U.S. without sponsorship. EOE.

SR. PROG. ANALYST, $59K/yr. Analyze Design, develop, test & implement software applications using Java, J2EE, Struts, UML, XML, weblogic,C, C++ & Oracle. MS in Eng/Sci w/2 yrs relevant exp or Bachelor degree in Eng/Si w/5 yrs relevant exp. email: donna@ifuturistics.com.

SYSTEM ADMINISTRATOR sought by Home Healthcare Co. (Warren, MI) Requires Bachelor's in Computer Information Technology. Min 1 yr exp. Responsibilities include administering and maintaining the network and internet systems. Resume to HRM, Mobility Plus Home Health Care, 5701 Chicago Road, Suite #D, Warren, MI 48092.

DB, $59K: Dev.,design, tune & test, ETL clnt/sever apps, Data Warehouse, Unix. MS in Eng/Sci w/2 yrs relavent exp or Bachelor degree in Eng/Si w/5 yrs relavent exp.email: donna@ifuturistics.com.

STANFORD UNIVERSITY, Department of Computer Science, Faculty Opening. The Department of Computer Science at Stanford University invites applications for a tenure-track faculty position at the junior level (Assistant or untenured Associate Professor). We give high priority to the overall originality and promise of the candidate's work rather than the candidate's sub-area of specialization within Computer Science. We are seeking applicants from all areas of Computer Science, including Foundations, Systems, Artificial Intelligence, Graphics, Computer Vision and Perception, Databases, and Human-Computer Interaction. We are also interested in applicants doing research at the frontiers of computer science with other disciplines, such as biology, neuroscience, and economics, and arts, with potential connections Stanford's main multidisciplinary initiatives: Human Health, Environment and Sustainability, the Arts and Creativity, and the International Initiative. An earned Ph.D., evidence of the ability to pursue a program of research, and a strong commitment to graduate and undergraduate teaching are required. A successful candidate will be expected to teach courses at the graduate and undergraduate levels and to build and lead a team of graduate students in Ph.D. research. Further information about the Computer Science Department can be found at http://cs.stanford.edu/. The School of Engineering website may be found at http://soe.stanford.edu/. Applications should include a curriculum vita, brief statements of research and teaching interests, and the names of at least four references. Candidates are requested to ask references to send their letters directly to the search committee. Applications and letters should be sent to: Search Committee Chair, c/o Laura Kenny-Carlson, Stanford University, Gates Hall 278, Stanford, CA 94305-9025; or via electronic mail to search@cs.stanford.edu. The review of applications will begin on January 9, 2008, and applicants are strongly encouraged to submit applications by that date; however, applications will continue to be accepted until the position is filled. Stanford University is an affirmative action, equal opportunity employer.

Technische Universität Berlin

Technical University of Berlin

Tenured Faculty Position (Full professorship W 3)

in "Computer Architecture"

Reference number: IV-680

The Faculty of Electrical Engineering and Computer Science invites applications of outstanding scientists for a full professorship in "Computer Architecture" with the following research focus: Design, development and realization of computer and network architectures. Applicants with a strong industrial background are welcome. Applicants are expected to have extended knowledge and experience in several of the following areas: safe and secure and/or real-time oriented computers; distributed, parallel, or multi-core architectures; storage technologies and architectures; architecture and realization of embedded processors with focus on system-on-chip; network-on-chip; dynamic reconfiguration; self repair; self organization.

The holder of the position is expected to conduct research and teaching in the above fields, both in the Bachelor program and in the Master program of the Faculty. The successful candidate is expected to teach courses in German after two years.

Requirements: Compliance with Berlin appointment laws in § 100 BerlHG. Further information can be provided on request.

To ensure equal opportunity for men and women, applications from female candidates with the advertised qualifications are explicitly solicited. Severely disabled applicants with equivalent qualifications will be given preferential treatment.

Applications should be sent within 4 weeks after publication to: **Dekan der Fakultaet IV, Sekr. FR 5-1, Franklinstrasse 28/29, D 10587 Berlin, Germany**

VirginiaTech
Invent the Future

The Department of Computer Science at Virginia Tech (www.cs.vt.edu) invites applications for the position of department head. The head should strive to continue the accelerating rise of the department toward greater national prominence – and is expected to have a strong commitment to advancing the research and teaching missions of the department, to nurturing interdisciplinary collaborations, and to working closely with the university in advancing departmental strategic goals.

CS@VT is expanding with regard to people, research, and resources. In the last 5 years, 18 faculty members have been hired to make a total of 45. The number of PhDs awarded places the department among the top 30 in the United States. Research expenditures have more than tripled in the last 7 years. Broadening from long-standing research strengths in HCI, HPC, CS education, and digital libraries, there are growing numbers of projects in interdisciplinary areas such as computational biology and bioinformatics, CyberArts, digital government, and problem solving environments. A new building opened in 2006, leading to a near doubling in space.

CS@VT offers BS, MS, and Ph.D. programs in the Blacksburg campus and a graduate program at its satellite campus in the Washington, D.C. area (which has about 100 students). CS@VT is part of the College of Engineering (www.eng.vt.edu) – the premier engineering school in the Commonwealth of Virginia. The college has a history of innovation, e.g., inexpensive supercomputing (System X). Blacksburg is consistently ranked among the country's best places to live (http://www.liveinblacksburg.com/).

Salary for suitably qualified applicants is competitive and commensurate with experience. Virginia Tech is an Equal Opportunity/Affirmative Action Institution.

Applications must be submitted online to https://jobs.vt.edu for posting 070739. Inquiries should be directed to Edward A. Fox, fox@vt.edu.

ADVERTISER / PRODUCT INDEX

SEPTEMBER 2007

Boldface denotes advertisements in this issue.

Computer

IEEE Computer Society
10662 Los Vaqueros Circle
Los Alamitos, California 90720-1314
USA
Phone: +1 714 821 8380
Fax: +1 714 821 4010
http://www.computer.org
advertising@computer.org

Advertising Sales Representatives

Mid Atlantic (product/recruitment)
Dawn Becker
Phone: +1 732 772 0160
Fax: +1 732 772 0164
Email: db.ieeemedia@ieee.org

New England (product)
Jody Estabrook
Phone: +1 978 244 0192
Fax: +1 978 244 0103
Email: je.ieeemedia@ieee.org

New England (recruitment)
John Restchack
Phone: +1 212 419 7578
Fax: +1 212 419 7589
Email: j.restchack@ieee.org

Northwest (product)
Lori Kehoe
Phone: +1 650 458 3051
Fax: +1 650 458 3052
Email: l.kehoe@ieee.org

Southeast (recruitment)
Thomas M. Flynn
Phone: +1 770 645 2944
Fax: +1 770 993 4423
Email: flynntom@mindspring.com

Midwest (product)
Dave Jones
Phone: +1 708 442 5633
Fax: +1 708 442 7620
Email: dj.ieeemedia@ieee.org

Will Hamilton
Phone: +1 269 381 2156
Fax: +1 269 381 2556
Email: wh.ieeemedia@ieee.org

Joe DiNardo
Phone: +1 440 248 2456
Fax: +1 440 248 2594
Email: jd.ieeemedia@ieee.org

Midwest/Southwest (recruitment)
Darcy Giovingo
Phone: +1 847 498 4520
Fax: +1 847 498 5911
Email: dg.ieeemedia@ieee.org

Southwest (product)
Steve Loerch
Phone: +1 847 498 4520
Fax: +1 847 498 5911
Email:
steve@didierandbroderick.com

Connecticut (product)
Stan Greenfield
Phone: +1 203 938 2418
Fax: +1 203 938 3211
Email: greenco@optonline.net

Southern CA (product)
Marshall Rubin
Phone: +1 818 888 2407
Fax: +1 818 888 4907
Email: mr.ieeemedia@ieee.org

Northwest/Southern CA (recruitment)
Tim Matteson
Phone: +1 310 836 4064
Fax: +1 310 836 4067
Email: tm.ieeemedia@ieee.org

Southeast (product)
Bill Holland
Phone: +1 770 435 6549
Fax: +1 770 435 0243
Email: hollandwfh@yahoo.com

Japan (recruitment)
Tim Matteson
Phone: +1 310 836 4064
Fax: +1 310 836 4067
Email: tm.ieeemedia@ieee.org

Europe (product/recruitment)
Hillary Turnbull
Phone: +44 (0) 1875 825700
Fax: +44 (0) 1875 825701
Email: impress@impressmedia.com

Advertising Personnel

Marion Delaney
IEEE Media, Advertising Director
Phone: +1 415 863 4717
Email: md.ieeemedia@ieee.org

Marian Anderson
Advertising Coordinator
Phone: +1 714 821 8380
Fax: +1 714 821 4010
Email: manderson@computer.org

Sandy Brown
IEEE Computer Society,
Business Development Manager
Phone: +1 714 821 8380
Fax: +1 714 821 4010
Email: sb.ieeemedia@ieee.org

Forensic Computing: A Practitioner's Guide, 2nd ed., Tony Sammes and Brian Jenkinson. Forensic computing is becoming of primary importance as computers increasingly provide sources of evidence in criminal investigations. However, for such evidence to be legally useful it must be collected and processed according to rigorous principles.

The authors show how information held in computer systems can be recovered when it has been hidden or subverted by criminals, and they give readers the means to ensure that it is accepted as admissible evidence in court. Updated to fall in line with ACPO 2003 guidelines, the book is illustrated with many case studies and worked examples and can help practitioners and students gain a clear understanding of the principles involved in password protection and data encryption, the evaluation procedures used in circumventing a system's internal security safeguards, and full search and seizure protocols for experts and police officers.

Springer; www.springer.com; 1-84628-397-3; 470 pp.

Competing for the Future: How Digital Innovations Are Changing the World, Henry Kressel with Thomas V. Lento. Digital technology has revolutionized our economy and lifestyles. But how many of us really understand the drivers behind the technology—the significance of going digital, the miniaturization of circuit boards, the role of venture capital in financing the revolution, the importance of research and development? How many of us understand what it takes to make money from innovative technologies? Should we worry about manufacturing going offshore? What role do India and China play in the digital economy?

Drawing on a lifetime's experience in the industry—as an engineer, senior manager, and a partner in a venture capital firm—the author offers answers to all these questions. He explains how the technology works, why it matters, how it is financed, and what the key lessons are for public policy.

Cambridge University Press; www.cambridge.org; 978-0-521-86290-5; 422 pp.

Outside-in Software Development: A Practical Approach to Building Successful Stakeholder-Based Products, Carl Kessler and John Sweitzer. Imagine the ideal development project, one that will deliver exactly what clients need. It will achieve broad, rapid, enthusiastic adoption and a productive, high-morale team of expert software professionals will design and build it.

This book's outside-in approach to software development can make such a project possible. The authors show readers how to identify the stakeholders who will determine a project's real value, shape decisions around their real needs, and deliver software that achieves broad, rapid, enthusiastic adoption. They present an end-to-end framework and practical implementation techniques that development teams can quickly benefit from, regardless of project type or scope. Using this proven approach can help developers improve the effectiveness of every client conversation, define priorities with greater visibility and clarity, and make sure all code delivers maximum business value.

IBM Press/Prentice Hall PTR; www.phptr.com/ibmpress; 0-13-157551-1; 256 pp.

Ethical and Social Issues in the Information Age, Joseph Migga Kizza, 3rd ed. Ethical dilemmas have risen in number and intensity with the increasing dependence of contemporary society on computers and computer networks. Despite the proliferation of expert remedies, viable solutions to computer security issues remain too elusive, and society continues to suffer at the hands of cybercriminals, vandals, and hackers.

This comprehensive third edition takes off where the second ended, examining ethical, social, and policy challenges stemming from the emergence of cyberspace, the convergence of telecommunication and computing technologies, and the miniaturization of computing, telecommunication, and information-enabling devices.

This accessible volume broadly surveys thought-provoking questions about the impact of these new technologies, with particular emphasis on the rapid growth of a multitude of computer networks, including the Internet. It assumes only a modest familiarity with basic computer literacy.

Springer; www.springer.com; 1-84628-658-2; 439 pp.

Aligning Modern Business Processes and Legacy Systems: A Component-Based Perspective, Willem-Jan van den Heuvel. Distributed business component computing—the assembling of business components into electronic business processes, which interact via the Internet—caters to a new breed of enterprise systems that are flexible, relatively easy to maintain and upgrade to accommodate new business processes, and relatively simple to integrate with other enterprise systems. Companies with unwieldy legacy systems find it difficult to align their old systems with novel business processes. These systems, tightly intertwined with existing business processes and procedures, also have a brittle architecture after years of ad hoc fixes.

This book provides a methodological framework based on reverse engineering, which lets legacy systems be componentized; forward engineering, which derives a set of business components from requirements of the new business processes; and alignment of new business processes and componentized legacy systems.

MIT Press; mitpress.mit.edu; 0-262-22079-8; 230 pp.

Send book announcements to newbooks@ computer.org.

The Current State of Business Intelligence

Hugh J. Watson, University of Georgia
Barbara H. Wixom, University of Virginia

BI is emerging as a key enabler for increasing value and performance.

In the early 1970s, decision-support systems were the first applications designed to support decision making. They were a contrast to transaction-processing or operational applications, such as order entry, inventory control, and payroll systems.

Over the years, various decision-support applications—executive information, online analytical processing (OLAP), and predictive analytics—have emerged and expanded the decision-support domain. In the early 1990s, Howard Dressner, then an analyst at the Gartner Group, coined the term *business intelligence*. BI is now widely used, especially in the world of practice, to describe analytic applications.

BI is currently the top-most priority of many chief information officers. In a survey of 1,400 CIOs, Gartner found that BI projects were the number one technology priority for 2007. BI "has become a strategic initiative and is now recognized by CIOs and business leaders as instrumental in driving business effectiveness and innovation," according to the company's research vice president, Andreas Bitterer (www.gartner.com/it/page.jsp?id=500680).

BI FRAMEWORK

As Figure 1 shows, BI is a process that includes two primary activities: *getting data in* and *getting data out*.

Data warehousing

Getting data in, traditionally referred to as data warehousing, involves moving data from a set of source systems into an integrated data warehouse. The source systems typically represent heterogeneous technical platforms and data structures. Sources can reside within the organization, be supplied by an external data provider, or come from a business partner.

A data warehouse team extracts data from the source systems and transforms it so that it is meaningful for decision support. For example, records from several systems can be matched and consolidated based on a customer identification number, or all currency fields can be converted into dollars. Sometimes the warehouse team creates new fields during data transformation, such as time period totals or customer value scores.

Getting data in is the most challenging aspect of BI, requiring about 80 percent of the time and effort and generating more than 50 percent of the unexpected project costs. The challenge stems from multiple causes, such as poor data quality in the source systems, politics around data ownership, and legacy technology.

The data warehouse team places the transformed data into a data store that is subject-oriented, integrated, time-variant, and nonvolatile. These are the classic characteristics of warehouse data that Bill Inmon described in 1991 in his seminal *Building the Data Warehouse* (QED/Wiley).

Depending on the architecture, the data warehouse may feed dependent data marts, which have a more narrow scope than a warehouse; marts focus on a particular functional area, geographic region, application, or organization division. Maintaining decision support data in a warehouse or marts that source their data from the warehouse ensures "a single version of the truth."

Metadata plays a key role in data warehousing because of the complexity of the data migration process. Data warehouse teams and business users must understand myriad characteristics of the data to manipulate and use it effectively. Metadata is technical and business in nature; it describes field values, sizes, ranges, field definitions, data owners, latency, and transformation processes. Metadata provides transparency as data moves from sources to the warehouse to end users.

Business intelligence

Getting data in delivers limited value to an enterprise; only when users and applications access the data and use it to make decisions does the organization realize the full value from its data warehouse. Thus, getting data out receives most attention from organizations. This second activity, which is commonly referred to as BI, consists of business users and

Computer Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page qMags

applications accessing data from the data warehouse to perform enterprise reporting, OLAP, querying, and predictive analytics.

USE AND VALUE OF BI

The cost of hardware, software, and staff to run a distributed decision support platform is significant.

Benefits

Initially, BI reduces IT infrastructure costs by eliminating redundant data extraction processes and duplicate data housed in independent data marts across the enterprise. For example, 3M justified its multimillion-dollar data warehouse platform based on the savings from data mart consolidation (H.J. Watson, B.H. Wixom, and D.L. Goodhue, "Data Warehousing: The 3M Experience," *Organizational Data Mining: Leveraging Enterprise Data Resources for Optimal Performance,* H.R. Nemati and C.D. Barko, eds., Idea Group Publishing, 2004, pp. 202-216).

BI also saves time for data suppliers and users because of more efficient data delivery. End users ask questions like "What has happened?" as they analyze the significance of historical data. This kind of analysis generates tangible benefits like headcount reduction that are easy to measure; however, these benefits typically have local impact.

Over time, organizations evolve to questions like "Why has this happened?" and even "What will happen?" As business users mature to performing analysis and prediction, the level of benefits become more global in scope and difficult to quantify. For example, the most mature uses of BI might facilitate a strategic decision to enter a new market, change a company's orientation from product-centric to customer-centric, or help launch a new product line.

Figure 2 illustrates the spectrum of BI benefits.

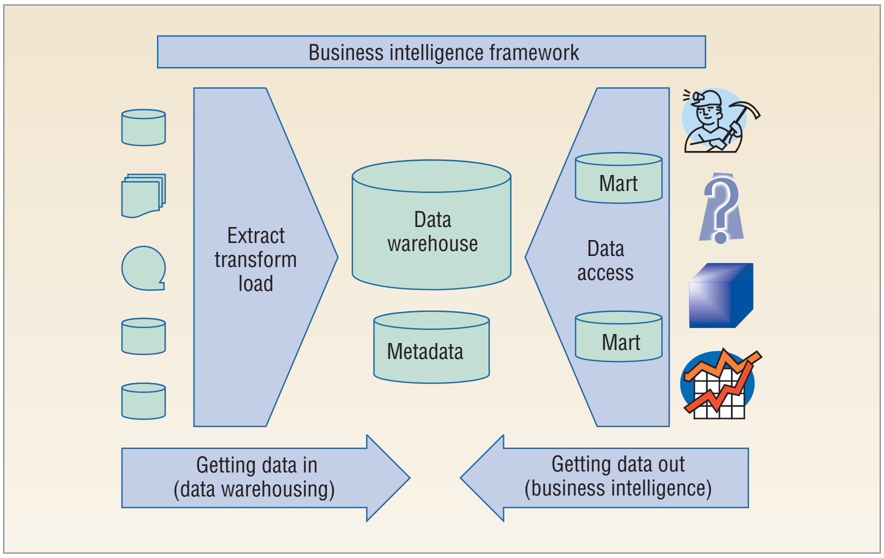


Figure 1. Business intelligence framework. The BI framework includes two primary activities, getting data in and getting data out.

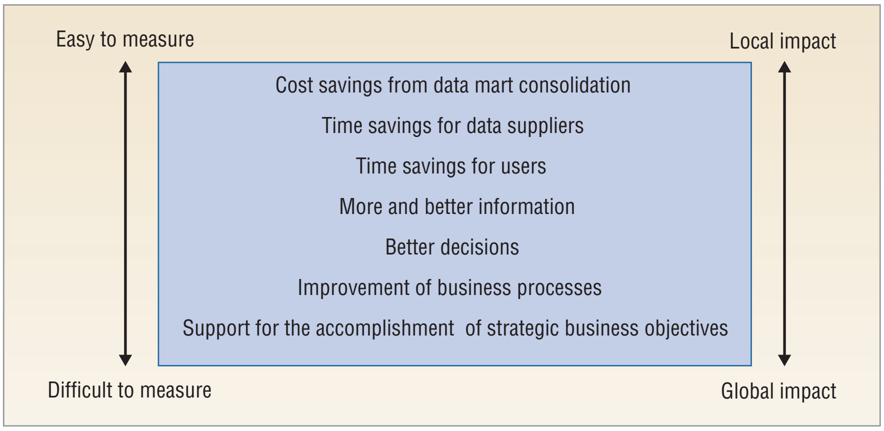


Figure 2. Spectrum of BI benefits. As business users mature to performing analysis and prediction, the level of benefits become more global in scope and difficult to quantify.

A BI-enabled business strategy

Through an innovative BI-enabled business strategy, the Las Vegas-based gaming corporation Harrah's Entertainment transformed how it competes in the marketplace (H.J. Watson and L. Volonino, "Customer Relationship Management at Harrah's Entertainment," *Decision Making Support Systems: Achievements, Trends and Challenges for the Decade,* M. Mora, G.A. Forgionne, and J.N.D. Gupta, eds., Idea Group Publishing, 2002, pp. 157-172).

In the wake of legislation in the late 1980s and early 1990s allowing gambling on riverboats and Indian reservations, senior management saw significant opportunities for growth by acquiring and building new properties. Management also decided to implement a brand strategy, understand its customers exceptionally well, and promote cross-casino play through Total Rewards, an innovative loyalty card program. This was a dramatic departure from then-current practices in which individual casino managers ran their properties as independent fiefdoms, and marketing was done on a property-by-property basis.

At the heart of the initiative was a customer-centric data warehouse that stored data on gaming (such as slot machine usage), hotels, and special events (such as wine-tasting weekends). Analysis of this data made it possible to understand customer profitability, lifetime value, and preferences, as well as the popularity of various games and

Computer | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

what promotional offers were most likely to be accepted within various market segments. Whereas in the past marketing campaigns and other decisions were based on intuition-based beliefs, or "Harrahisms," a more scientific, evidence-based approach to decision making emerged.

Harrah's investment in BI has been highly successful, resulting in a high return on investment and significant increases in revenue and its stock price. Other gaming companies have emulated some of Harrah's more discernable methods.

KEYS TO BI SUCCESS

Success with BI isn't automatic. Companies are more likely to be successful when certain facilitating conditions exist. Many of these conditions were critical to Harrah's success.

Senior management believes in and drives the use of BI. For BI to be useful on an enterprise basis, it must be driven from the top. Senior management should have a vision for BI, provide the necessary resources, and insist on the use of information-based decision making.

The use of information and analytics is part of the organization's culture. This contrasts with decision making based on intuition or "gut feelings." Not everyone can make this change easily, and new people might need to be put in place.

There is alignment between the business and BI strategies. When there is alignment, BI can be a powerful enabler of business strategy, including new business models that bring about organizational transformation.

There is effective BI governance. People, committees, and processes must be in place to manage and support BI. Governance addresses many important issues, including alignment, funding, project prioritization, project management, and data quality.

There is a strong decision support data infrastructure. Data warehouses help solve perhaps the most common cause of BI failure—a lack of high-quality data. Users will not accept or rely on data they do not trust.

Users have the necessary tools, training, and support to be successful. Users should be provided with data access tools that are appropriate for their needs, trained in how to use the tools and the data that is available, and given access to people who can help them use BI.

CURRENT BI DEVELOPMENTS

BI is continuing to evolve, and several recent developments are generating widespread interest, including real-time BI, business performance management (BPM), and pervasive BI.

> **Web-based systems provide access to data from anyplace that has an Internet connection.**

Real-time BI

Experienced BI professionals know that there has always been a demand for fresher data. Today, enterprise information integration (EII), enterprise application integration (EAI), and real-time data warehousing technologies are making it possible to deliver decision support data that is literally only minutes old. This development is profoundly changing the face of decision-support by making it possible to affect current decision making, operational business processes, and customer-facing applications.

Continental Airlines provides an excellent example of the value of real-time BI in dealing with late flights. From the flight manifest, customer profitability data, customer reservation information, real-time flight data from the plane, and current gate and departure time data, all stored in a real-time data warehouse, Continental can identify the high-value passengers who are at risk of missing a connection and make special arrangements to get those passengers and their luggage to their connecting flights on time (R. Anderson-Lehman et al., "Continental Airlines Flies High with Real-Time Business Intelligence," *MIS Quarterly Executive,* Dec. 2004, pp. 163-176).

Business performance management

Many companies are implementing scorecards and dashboards as key components of BPM initiatives. These tools visually summarize large amounts of data related to organizational performance. In a single or a few screens, users can quickly see how actual performance compares to goals, benchmarks, and previous performance. The adage "that which gets watched, gets done" exemplifies the value of scorecards and dashboards.

The world's largest supplier of contact lenses, 1-800-CONTACTS, has implemented performance dashboards in its call center to monitor and provide incentives for its agents. The dashboards use gauges and charts that show the agents' performance on metrics such as closing ratio, average order size, productivity, and quality.

Monthly incentive bonuses are tied to these metrics. The system updates the data every 15 minutes and shows not only the agents' current and month-to-date performance but how they are doing with respect to the call center averages, to other members of their team, and to a goal they set for themselves. The dashboards have significantly improved the center agents' performance.

Pervasive BI

BI is becoming more pervasive as it spreads to a larger use base, providing users the information they need to perform their jobs better. The availability of real-time data and easy-to-understand dashboards are important contributors to information democracy, but there are others as well.

Web-based systems provide access to data from anyplace that has an Internet connection. Event-based triggers are used to initiate alerts, such as messages to the sales force that an item is out of stock and that they should suggest an alternative item.

BI is also being embedded in business processes such as call center operations and applications like campaign management to improve organizational performance, often in ways that make the analytics transparent to users. Microsoft is stressing BI and increasing its products in the BI marketplace, including integrating new BI capabilities in Microsoft Office 2007.

BI RESOURCES

Resources on BI are available for both practitioners and academics.

The Data Warehousing Institute

TDWI (www.tdwi.org) is the premier educational organization for business intelligence and data warehousing managers and professionals. TDWI provides BI training, research, and networking opportunities to its members around the world. Founded in 1995, the organization offers vendor-neutral educational opportunities at quarterly conferences, on-site classes, and regional events and through its Web site. Each year, TDWI sponsors a BI best practices competition that recognizes organizations that have achieved significant success in BI.

Teradata University Network

For those who teach BI, the Teradata University Network (www.teradatauniversitynetwork.com) offers various free BI-related resources: course syllabi used by other faculty, book chapters, articles, research reports, cases, projects, assignments, PowerPoint presentations, software, large data sets, Web seminars, Web-based courses, discussion forums, podcasts, and certification materials.

The software—such as the Teradata database, MicroStrategy—is available through an application service provider arrangement so that schools do not have to install and maintain BI software.

A companion site, the Teradata Student Network (www.teradatastudentnetwork.com), provides students access to resources assigned by their instructor.

The future of business intelligence is bright. BI is changing how companies are managed, decisions are made, and people perform their jobs. Major vendors such as Microsoft, Oracle, and SAP are increasing their commitment and investment in BI. And there are other developments on the horizon—the integration of structured and unstructured data, rules engines, guided analytics, and enhanced data visualization—that promise to keep BI fresh and exciting. ■

Hugh J. Watson *holds a C. Herman and Mary Virginia Terry Chair of Business Administration in the Terry College of Business at the University of Georgia. Contact him at hwatson@terry.uga.edu.*

Barbara H. Wixom *is an associate professor at the McIntire School of Commerce at the University of Virginia. Contact her at bwixom@mindspring.com.*

Editor: Richard G. Mathieu, Dept. of Computer Information Systems and Management Science, College of Business, James Madison Univ., Harrisonburg, VA; mathierg@jmu.edu

Join the IEEE Computer Society online at

www.computer.org/join/

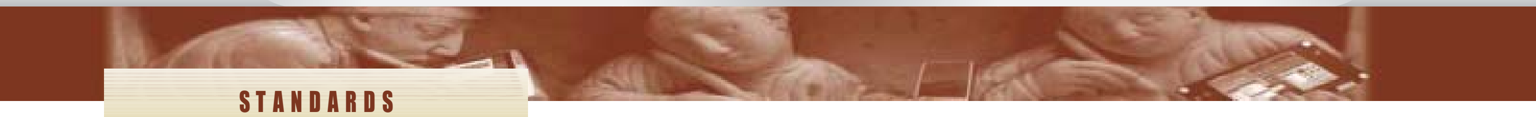
IEEE computer society

Complete the online application and get

- immediate online access to ***Computer***
- a free e-mail alias — **you@computer.org**
- free access to 100 online books on technology topics
- free access to more than 100 distance learning course titles
- access to the IEEE Computer Society Digital Library for only $118

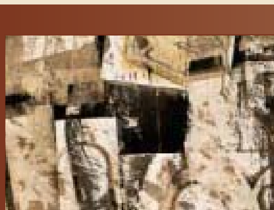
Read about all the benefits of joining the Society at

www.computer.org/join/benefits.htm



Standards, Agility, and Engineering

François Coallier, École de technologie supérieure



Engineering involves choosing the right tool, which implies an understanding of both the tools and the problem. Such behavior is agile.

Parts of the software and IT engineering community harbor many misconceptions associated with standards, especially process standards. The most common are that standards are rigid, obsolete, and plain boring. Standards are also perceived as being the antithesis of agility.

Yet standards create markets by enabling interoperability and economies of scale through component and interface standardization. In the IT service industry, standards also enable interoperability between organizations through process and competency benchmarking and standardization.

These benefits derive from interoperability between people and organizations that share a common vocabulary, compatible work processes, and the ability to benchmark individuals and organizations. Thus, we see a growing need for standards in areas such as processes, methods and good practices, bodies of knowledge, and formalisms.

STANDARDS DEFINED

Using the International Organization for Standardization (ISO) definition as a baseline, standards are "guideline documentation that reflects agreements on products, practices, or operations by nationally or internationally recognized industrial, professional, trade associations or governmental bodies."

Standards are referred to as guideline documents because they are not compulsory unless mandated so by an individual, an organization, or the market. They are agreements because they often reflect a specific level of consensus.

Many types of standards exist and can be elaborated inside an organization; a professional society such as the IEEE, the International Council on Systems Engineering (INCOSE), or the IT Service Management Forum (itSMF); an industrial consortium such as the Object Management Group (OMG), the Organization for the Advancement of Structured Information Standards (OASIS), and others; formal national bodies such as the Americal National Standards Institute (ANSI), the Standards Council of Canada (SCC), and others; or international organizations such as ISO, the International Electrotechnical Commission (IEC), and the International Telecommunication Union (ITU).

A given standard might be developed in one environment—market, professional, industry, national—and then migrate into a formal international standard. Market, professional, and industry standards might also represent an international consensus or de facto state that differs from formal international standards in the degree of breadth and formality that this consensus embodies.

STANDARDS LIMITATIONS

Standards are usually elaborated through a consensus process. This means that a group of experts creates the document through a series of iterations. Consensus is built among these experts from successful compromises reached through multiple negotiation sessions. Afterward, the document undergoes review by a larger body of experts and organizational representatives, with the goal of reaching a greater level of consensus to meet the final publishing criteria.

All of this underscores the fact that standards, like any other human creations, are far from perfect. Also, because they represent the consensus of a collection of individuals and organizations, they do not necessarily represent the state of the art or the most optimal solution in a given area. Rather, standards are conventions or baselines that a collection of people or organizations reach at a given point. This applies to all types of IT standards.

AGILITY

Either as a philosophy or, more explicitly, as a collection of concepts embedded into methodologies, *agility* has been a popular topic in the software and IT engineering world for many years. Succinctly, an agile approach to development is essentially a result-focused method that iteratively manages changes and risks. An agile method also actively involves customers, making them in effect part of the development team.

Agility is often contrasted to process-oriented approaches in which a rigid, documentation-heavy methodology is followed with very detailed specifications, and the relationships between developers and the customer are formal and sometimes adversarial.

As with many such concepts, the agility characteristic is not binary, as Figure 1 shows (modified from P. Kroll and P. Kruchten's *The Rational Unified Process Made Easy—A Practitioner's Guide to the RUP*, Addison-Wesley, 2003).

In Figure 1, the *x*-axis represents the degree of documentation and formality associated with a development process. "High ceremony" means many formal, documented meetings and heavy, process-driven documentation, while "low ceremony" means less formal meetings whose sole documentation might be in the form of updated process artifacts and more, lighter, outcome-driven documentation.

In Quadrant II, the process-based approaches have much in common with industrial engineering practices. In Quadrant III, the focus shifts to people and their skills. At the upper-hand corner of Quadrant II, a fixed-cost contract is in place, and lawyers can be involved when the developers and customer communicate about the project. At the lower corner of Quadrant III, "pay as you go projects" reside. In these, the customer forms part of the team. While agile methods are positioned in Quadrant III, extreme programming (XP) is usually positioned in the lower corner of this quadrant.

Agility is thus both a philosophy and a nonbinary attribute of a development process.

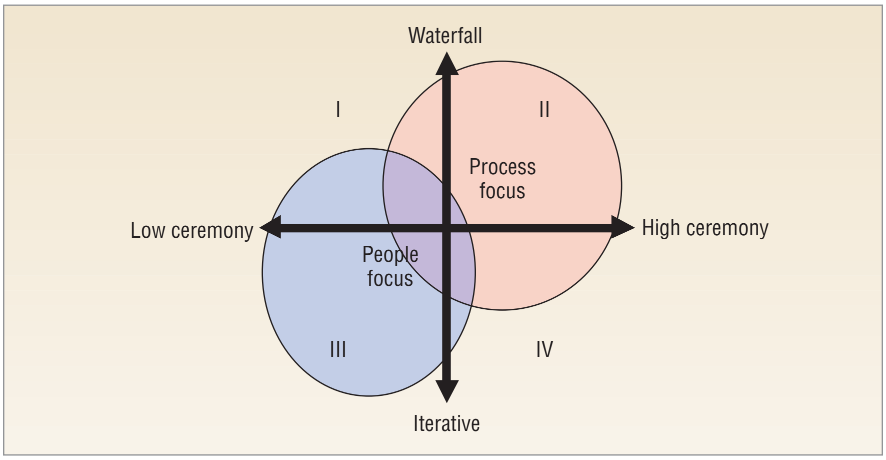


Figure 1. The spectrum of development approaches. The x-axis represents the degree of documentation and formality associated with a development process: High ceremony involves many formal, documented meetings; low ceremony involves less formal meetings with less documentation.

ENGINEERING

While the objective of science is to understand nature, engineering's objective is to build useful things using science. Put another way, scientists do reverse engineering while engineers do forward engineering. Engineers must not only build things, they must also do so within budgetary, schedule, resource, regulatory, and operational constraints.

When developing IT systems, engineers must not only deliver functionalities but also all those other "ities" commonly referred to as either "nonfunctional requirements" or "quality attributes." The quality attributes include performance, reliability, availability, security, usability, scalability, and maintainability. These attributes are not only important for embedded software systems, but also for modern information systems. Delivering those quality attributes can, in many cases, be much more challenging than delivering the functionality. All of this points out that engineering IT systems is much more involved than doing classical programming.

Many IT systems that engineers build can be safety-critical or, in the case of financials and electronic commerce systems, financially critical. In many cases, the best way to minimize risks when developing such systems, and any other type of software systems is through reuse or incrementally improving proven solutions. This means that engineering is by nature a profession where reuse—be it of proven standardized components, interface specifications and protocols, development methods, or architectural patterns—is a way of life.

Practicing reuse does not mean that engineering is boring technical work without any challenges. On the contrary, it requires a deep understanding of the problem to be solved and its applicable solution paths. Both analytic and holistic systems thinking are necessary. Reuse does not mean the absence of innovation—it means that innovation and risks must be balanced within the project's context.

ENGINEERING AND STANDARDS

The relationship between engineering and standards should now be obvious. Because standards facilitate reuse and, usually, document proven and generally acceptable practices, they are an intimate part of every engineer's professional practice. Standards also constitute a significant part of an engineer's teaching and training curriculum.

In software engineering, the relationship with standards goes deeper. While the foundations of software engineering include fields such as computer science, systems engineering, and project management, standards have played a significant role in the development and codification of this rather young engineering discipline.

The US National Bureau of Standards published the first software engineering standard barely eight years after the term "software engineering" was coined in 1968. The same year this standard was published, the IEEE created its software engineering standards committee. In 1977, the joint ISO and IEC standard subcommittee (ISO/IEC JTC 1/SC7, or SC7) was then created.

The IEEE Computer Society and SC7 have been working together to harmonize their respective collection of software and systems engineering standards and to continue joint development. Noteworthy has been the

IEEE's publication of the *Guide to the Software Engineering Body of Knowledge* (SWEBOK Guide) in 2004, followed closely by its adoption as a technical report by ISO and the IEC as TR 19759:2005. As of August 2007, there are 101 published SC7 standards, with approximately 50 from the IEEE CS, focusing on the area of software and systems engineering.

These standards are still evolving, with new ones being elaborated that cover new areas. Six new proposals for software and systems engineering as well as IT service management are currently under consideration at SC7.

AGILITY AND STANDARDS

Some developers perceive standards as the antithesis of agility. This view is difficult to understand because all agile methodologies, including XP, are documented processes. The difference between an agile method and a nonagile one is a matter of content, but it is also a function of the way the methodology's components are applied in a given project.

A project where safety-critical software is developed can be agile even if many of the project artifacts must be produced because of regulatory constraints. The agility in such a case can be in the collaborative nature of the customer relationship, a people-focused work environment, or other factors.

Many process standards also have built-in provisions that, implicitly, enable their use in an agile context. These take the form of a tailoring clause. While machine-to-machine interface standards provide rather straightforward engineering specifications and constraints, the process standards used as a baseline to define a methodology are very different.

ENGINEERING AND AGILITY

Engineers are, in principle, pragmatic professionals strongly focused on their work's outcome. This implies agility. The multitude of patterns, regardless of their nature, that engineers learn through formal training and their professional career are all part of a toolbox. This is true also for process patterns and methods. Competent engineers who understand the contents of their toolbox and who also understand the nature of the problem that must be solved should use the proper tools. Such behavior is agile.

The people dimension of agility involves being able to choose the right tool, which implies an understanding of both the tools and the problem. This means having the right knowledge, skills, and experience. The horror stories that circulate about wasteful work generated by standards are generally more a result of poor engineering and management skills than a problem with the standard.

Not only are standards and agility mutually compatible, both concepts are also key components of any engineering discipline.

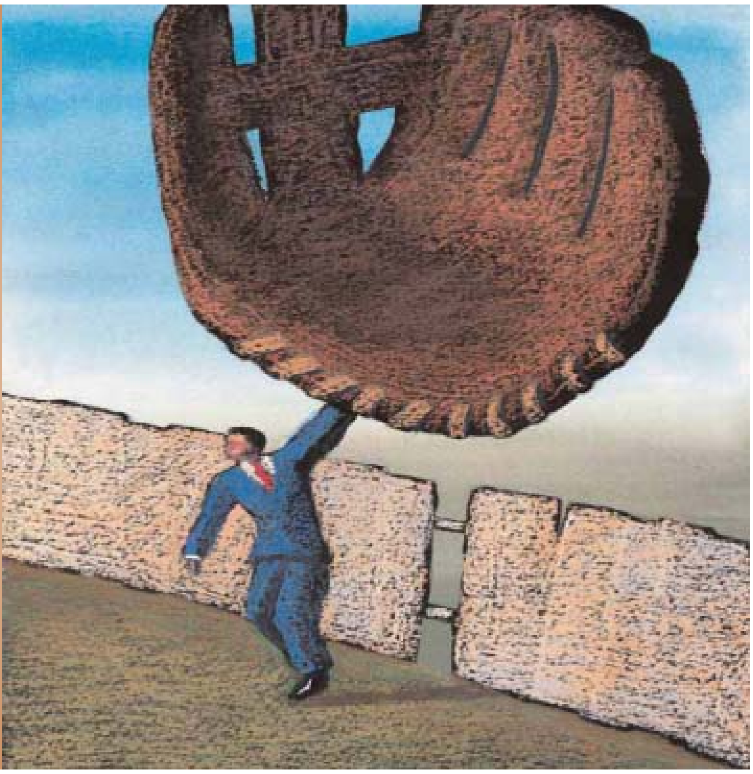
François Coallier is a professor and chair of the Department of Software and IT Engineering, École de technologie supérieure, Montréal. He is also the chairman of the ISO/IEC JTC 1/SC7, Software and Systems Engineering subcommittee. Contact him at francois.coallier@etsmtl.ca.

Editor: John Harauz, Jonic Systems Engineering Inc., Willowdale, Ont., Canada; j.harauz@computer.org

Giving You the Edge

IT Professional magazine gives builders and managers of enterprise systems the "how to" and "what for" articles at your fingertips, so you can delve into and fully understand issues surrounding:
- Enterprise architecture and standards
- Information systems
- Network management
- Programming languages
- Project management
- Training and education
- Web systems
- Wireless applications
- And much, much more …

IT Professional
www.computer.org/itpro



Online Experiments: Lessons Learned

Ron Kohavi and Roger Longbotham
Microsoft

Web experiments generate insights and promote innovation.

While at Amazon.com from 1997 to 2002, Greg Linden created a prototype system that made personalized recommendations to customers when they placed items in their shopping cart (http://glinden.blogspot.com/2006/04/early-amazon-shopping-cart.html). The prototype looked promising, but "a marketing senior vice-president was dead set against it," claiming it would distract people from checking out, and Linden was forbidden to work on it further.

Nonetheless, Linden ran an online controlled experiment, and the "feature won by such a wide margin that not having it live was costing Amazon a noticeable chunk of change. With new urgency, shopping cart recommendations launched." Since then, multiple commercial Web sites have imitated Amazon.com's feature.

The Web provides an unprecedented opportunity to evaluate proposed changes or new features quickly using controlled experiments. The simplest experiments randomly assign live users to one of two variants: the *control,* which is commonly the existing version, and the *treatment,* which is usually a new version being evaluated.

Experimenters first collect metrics of interest, from runtime performance to implicit and explicit user behaviors and survey data. They then conduct statistical tests on this data to determine whether a significant difference exists between the two variants, which in turn leads them to retain or reject the null hypothesis that there's no difference between the versions.

Online experiments offer substantial benefits, but many pitfalls can trip up practitioners. Our work at Microsoft and other companies, including Amazon.com, Blue Martini Software, Capital One, and Xerox has led to some important lessons in properly conducting such experiments. A practical guide is available at http://exp-platform.com/hippo.aspx.

OVERALL EVALUATION CRITERION

A common pitfall in Web experiments is the use of multiple metrics. For an organization that seeks to run many experiments in a given domain, it's strongly desirable to select a single quantitative measure, or *overall evaluation criterion* (OEC), to help determine whether a particular treatment is successful or not.

Consider an experiment with 25 different metrics in which three organizational teams have a stake in the outcome. How do the teams decide whether to launch the treatment if some of the metrics are positive and some are negative? Worse, if the metrics are favorable for one team and negative for another, deciding what to do could be contentious.

Also, with 25 metrics, even if the experiment has no effect, we should expect one or more metrics to appear statistically significantly different when using the common 95 percent confidence intervals. Having a single metric for decision making simplifies the process, clarifies the interpretations, and aligns the organization behind a clear objective.

The OEC can be a simple metric that summarizes important business goals or a weighted combination of metrics, as is often used in credit scores. It should reflect long-term objectives such as higher revenue, more users, or greater user engagement. In many cases, the OEC is an estimate of users' lifetime value.

Factors that might positively impact the OEC include higher visit frequency, longer sessions, and disclosure of personal information such as zip codes and hobbies that can be used to improve the user experience. For retail sites, adding to a cart or wish list increases the probability of a future purchase, even if the transaction didn't occur in the current session. Users who have a large social network, sometimes called "connectors" or "sneezers," can have a much larger influence and thus a higher value in the OEC.

RANDOMIZATION

Good experimental design calls for blocking or randomizing over *nuisance factors* that impact the OEC but aren't of interest, such as the time of day, the day of the week, and the server that handles the request. Because time is a critical factor, running the control and treatment concurrently is essential for online experiments and far superior to interrupted time series.

Server fleets

Using different server fleets for the control and treatment can skew experimental results. Suppose, for example, server fleet F1 runs the control, and a newer server fleet, F2, which handles requests faster, runs the new treatment. This discrepancy introduces an unintentional bias. One way to identify such biases and other problems with user assignment is to run an A/A, or null, test in which the control and treatment are the same.

Representative environment

The experimental environment should represent the eventual environment as closely as possible. For example, if a Web site runs an experiment for three weekdays but has a different set of users during weekends, the results might not generalize. The experiment should run for multiple weeks to determine whether significant differences exist between weekdays and weekends. Partial days shouldn't be included for the same reason: Morning users might react differently to the treatment than afternoon, evening, or overnight users. Likewise, running an experiment during the Christmas season might not generalize to the rest of the year.

Hashing function

Randomization is too important to be left to chance. A common way to maintain user experience consistency is to employ a hashing function on a user ID stored in a cookie. Picking a good hashing function is critical—some aren't random enough. Cryptographic hashes such as MD5 are generally the best.

Failure to randomize properly can confound results when running multiple tests simultaneously. For example, computing separate hashes of both the experiment name and user ID, and then executing the final XOR at assignment time, produces severe correlations between experiments.

Assuming two experiments with two variants each running at 50/50, if the most significant bit of the hashes of the experiment names matched, users would always get the same assignment across both experiments; if they didn't match, users would get exactly the opposite assignment. Automated checks can ensure that user assignment is random and matches the design proportions.

Opt in/opt out

Letting users opt in or out of an experiment invalidates the randomness. Opt in especially is usually a bad idea. Clickthroughs for users that opt in are significantly higher, but these often don't materialize when all users switch to the new format. One reason for this is that people who opt in are likely to be heavy users whose clickthrough rate is already higher.

A significant portion of traffic to most Web sites consists of one-click sessions—users who enter the site and never click again. Some of these users are nonhuman bots and crawlers, and some are people who go to the site by mistake. Whatever the reason, this "bad" traffic never opts in, thus opt-in treatments have a selection bias.

Failure to randomize properly can confound results when running multiple tests simultaneously.

MINIMUM DURATION

An important but often overlooked step in a controlled experiment is planning for sufficient sample size, that is, statistical power. For most online experiments, this translates to how long to run the experiment.

Power calculation

It's not uncommon to run a test on a small population—say, one percent of users—and discover four weeks later that the test must continue for 10 more weeks to detect the expected size change. Power calculations help plan how long an experiment should run and what percentage of the population to expose.

To accomplish this, first determine the *sensitivity*, or how large a change in the OEC you want to be able to detect. For example, if your OEC is revenue per user and the mean is $7 on your site, you might want a sensitivity of 1 percent, or the ability to detect a change to the mean of $0.07. Then, estimate your OEC's standard deviation from historical data or an A/A test.

Assuming that you will do t-tests against the control, apply the formula $n = (16 \times \sigma^2)/\Delta^2$, where n is the number of users in each variant (assumed to be of equal size), σ^2 is your OEC's variance, and Δ is the sensitivity. The coefficient of 16 provides 80 percent power—the sample size provided by the formula gives an 80 percent probability of rejecting the null hypothesis that there's no difference between the treatment and control if the true mean differs from the true control by Δ. Replacing the 16 with 21 will increase the power to 90 percent.

Once you have your sample size, calculate how long you'll need to run the test based on normal traffic to the Web site and preferably round up to whole weeks.

Overlapping experiments

Novice experimenters overly concerned about interactions tend to run experiments sequentially or on small disjointed subsets of users. However, unless there's a good reason to believe that experiments interact strongly, second-order effects are usually weaker than main effects.

Determining whether significant interactions occurred post hoc through, say, pairwise testing, is relatively easy. It's more important to run experiments on a large percentage of users to have sufficient power to detect small effects. Four independently randomized experiments that are concurrent and overlapping, each splitting users equally into control and treatment groups, are generally preferable to splitting the population into a 20 percent control group and four 20 percent treatment groups.

Concurrent univariate experiments

Commercial marketing literature suggests that univariate experiments that test one factor at a time are inferior to multivariate experiments, which vary many variables simultaneously according to a specialized design such as fractional factorials. However, this is akin to substituting polynomial for linear models: It sounds good in theory, but the complexity leads to linear models being used more often in practice.

Running multiple univariate experiments concurrently offers several advantages: Univariate analysis is much easier for end users to understand, experiments can be turned on and off when bugs are found without shutting down other experiments, and any interactions among the factors can still be estimated as if using a full-factorial design.

ANALYSIS

Because they contain new code for the treatments, online experiments have a higher failure rate and more bugs. On the other hand, unlike offline experiments, analysis of online experiments should begin as soon as possible, that is, in near real time.

Bugs

Suppose you're looking for a 1 percent improvement in the treatment, but it has some bugs and key metrics are degrading by 10 percent. Because the power formula is quadratic in the effect size (Δ), you can detect a 10 percent degradation 100 times faster than a 1 percent change—thus, an experiment planned to run for two weeks to detect a 1 percent delta will have enough power to detect a 10 percent drop in less than 4 hours.

Consider defining bounding boxes for some metrics. For example, if the time to generate a page increases above a preset threshold too often, it's likely there's a bug, and you should abort the experiment.

Ramping up the percentage assigned to treatment over time is also recommended. Start with a small percentage, check that the metrics are within reasonable bounds, and only then increase the percentage assigned to the treatment. Ramping up over a day or two makes it possible to catch egregious errors while the experiment is exposed to a small population. More sophisticated analysis must be done if you combine time periods in which the treatment percentage changes.

Primacy and newness effects

Users accustomed to a particular feature might reject a new one, even if it's better. Conversely, a flashy new widget might initially attract users. In these cases, it's important to run the experiment longer and evaluate users' behavior after multiple exposures to the feature. Analysis of trends in the OEC is also helpful.

Secondary metrics

Experimenters often ignore secondary metrics that impact the user experience such as JavaScript errors, customer-service calls, and Web-page loading time. Experiments at Amazon.com showed that every 100-ms increase in the page load time decreased sales by 1 percent, while similar work at Google revealed that a 500-ms increase in the search-results display time reduced revenue by 20 percent.

> **Analysis of online experiments should begin as soon as possible, that is, in near real time.**

Data cleansing

Bots and crawlers can account for 5 to 40 percent of a Web site's page views. Because many robots are from search engines, you want them to crawl the site. In most cases, robots don't send cookies, and they tend to distribute across the control and treatments, simply adding noise. However, some do send cookies and can significantly impact statistics and the OEC.

Looking for outliers and investigating them is important to detect and remove possible instances of fraud. For example, sellers on online retail sites might buy their own products to make a top-sellers list. The orders, usually of large quantities, are either cancelled later, returned, or, in cases where the seller is also the shipper, not actually shipped.

Finally, in our experience, data from Web sites is fragile, especially that from the online logs. Always check for metrics that get contaminated in some way. Automatic data validation checks are very helpful.

Falling for features

Two other common mistakes in online experiments are launching a feature that is statistically significantly different but has little business value and launching a feature because it doesn't negatively impact users.

Consider, for example, a feature that improves on some key metric and the difference is statistically significant, but the delta is 0.01 percent. The business impact is negligible, but the cost of maintaining the feature is high, outweighing any benefit of the feature.

As another example, suppose the CEO's favorite new feature isn't significantly different but there's a strong push to deploy it because the project development team worked hard. Not only are there maintenance costs involved, but the experiment could be underpowered and might not detect that the feature has a lower OEC.

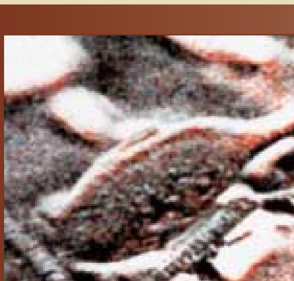
Online experiments, whether they fail or succeed, generate insights that can bring a quick return on investment and promote innovation. We reserve the most important lesson for the end, and it's called Twyman's law: Any statistic that appears interesting is almost certainly a mistake. Make sure to double-check your data and computations.

__Ron Kohavi__ is the general manager of Microsoft's Experimentation Platform, and __Roger Longbotham__ is the team's lead statistician. Contact them at http://exp-platform.com.

Editor: Simon S.Y. Shim, Director of Development, SAP Labs, Palo Alto, CA; simon.shim@sap.com.

It's Time to Stop Calling Circuits "Hardware"

Frank Vahid, University of California, Riverside



Expanding the software concept to spatial models like circuits will facilitate programming next-generation embedded systems.

In computer technology, names often change. Even the word "computer" previously referred to a human whose job was to crunch numbers, but today refers to a machine. The advent of field-programmable gate arrays requires that we stop calling circuits "hardware" and, more generally, that we broaden our concept of what constitutes "software." Doing so will help establish a cadre of engineers capable of maximally exploiting modern computing platforms that involve multiprocessors and FPGAs.

When the term software was coined in the late 1950s to raise awareness that writing programs was becoming as important as the design and maintenance of the hardware that then dominated computing efforts, the meanings of software and hardware were clear. Hardware referred to physical components like boards, fans, tubes, transistors, and connections—and could be touched. Software referred to the bits representing a programmable processor's program. During the decades that followed, building programs (software design) evolved to require dramatically different skills than connecting physical components into circuits (hardware design).

IMPLEMENTING CIRCUITS

Today, embedded-system designers frequently supplement microprocessors with custom digital circuits, often called coprocessors or accelerators, to meet performance demands. A circuit is simply a connection of components, perhaps low-level components like logic gates, or higher-level components like controllers, arithmetic logic units, encoding engines, or even processors.

FPGAs as software

Increasingly, designers implement those circuits on an FPGA, a prefabricated chip that they can configure to implement a particular circuit merely by downloading a particular sequence of bits. Therefore, a circuit implemented on an FPGA is literally software.

The key to an FPGA's ability to implement a circuit as software is that an N-address-input memory can implement any N-input combinational circuit. For example, given the function $A'B'C' + ABC$, it's possible to simply connect A, B, and C to the address inputs of a three-address-input (eight-word) memory, store a 1 into the first and last memory words, and store 0s into the other six words. The memory will then output 1 when $ABC = 000$ ($A'B'C'$) or when $ABC = 111$ (ABC), and will output 0 for any other values of ABC.

An FPGA has thousands of small memories each having perhaps four inputs and two outputs, and each known as a lookup table. A circuit with more than four inputs can be partitioned into four-input subcircuits, each mapped to a LUT. To support sequential circuits, LUT outputs feed into flip-flops, collectively known as a configurable logic block. A multiplexor, along with a programmable register that controls the multiplexor's select lines, chooses whether the CLB output comes directly from the LUT or from the flip-flop.

Finally, CLBs are preconnected through miniature crossbar-like switches, in which programmable registers also determine the actual paths, to ultimately achieve a particular circuit's connections. A developer thus can implement a desired circuit on an FPGA by writing the proper bits to each LUT and programmable register, as determined by design automation tools.

The fact that developers can implement circuits as software on FPGAs is a very big deal because circuits can implement some computations hundreds or even thousands of times faster than microprocessors. The speedups come from a circuit's concurrency at the bit level, arithmetic level, or even process level, because every circuit component executes simultaneously. For example, reversing the bits of a 32-bit number, performing 50 independent multiplications, or executing 20 processes each requires a long sequence of instructions on a microprocessor. But on an FPGA, the bit reversal is just crossed connections, the multiplications could

Computer | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | qMags

execute concurrently on 50 multipliers, and the processes could execute in parallel as 20 different circuits.

Underutilizing FPGAs

Computationally intensive applications involving video or audio processing, compression, encryption, real-time control, or myriad other tasks found in domains like consumer electronics, communications, networking, industrial automation, automotive electronics, medical devices, office equipment, and much more could thus execute hundreds or even thousands of times faster as circuits on FPGAs than as instructions on microprocessors—without the rigidity, high cost, and long turnaround times associated with building new chips. Such speedups, achieved using the changeable medium of software (on FPGAs), enable new applications not otherwise possible.

Yet, embedded-system application designers working with software often overlook FPGAs, even when FPGA performance gains would tremendously benefit an application while satisfying cost and power constraints. And designers never even conceive many applications that FPGA performance would enable because they don't consider that software can meet those applications' performance demands. However, because an embedded-system designer faces the challenge of making design tradeoffs under multiple excruciating constraints on design metrics like performance, power, cost, size, and design time, the effective designer can't be strong with just microprocessors or FPGAs; these designers must be capable of finding just the right balance of both.

Part of the problem is that computing-oriented application designers tend to be unfamiliar with FPGAs, which in turn occurs largely because the engineering design and education communities continue to treat circuits as hardware. Until the advent of FPGAs, treating circuits as hardware made sense. Designers implemented circuits either by connecting physical components or by creating a new physical chip—things they could touch.

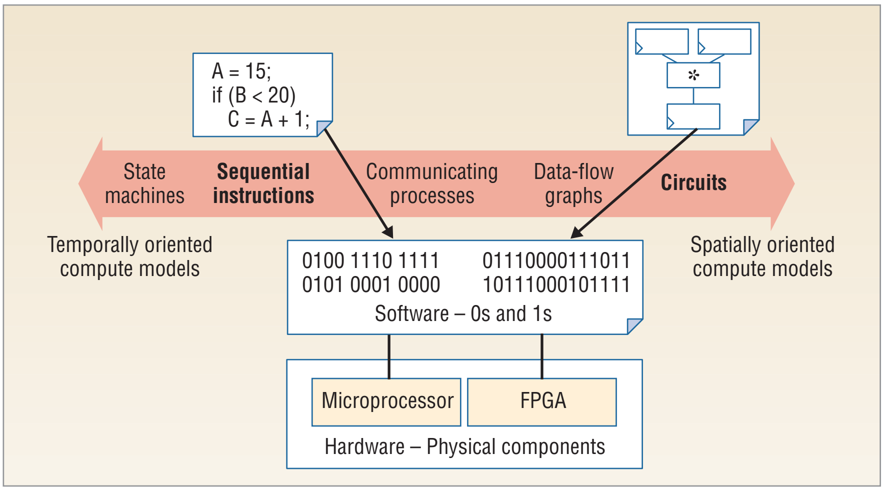


Figure 1. Circuits as software. Temporally and spatially oriented computing models can both be compiled to software that is downloaded into hardware components.

Yet today, even with circuits increasingly implemented as software on an FPGA, circuit designers are still known within companies as hardware people, circuit-design languages are still called hardware description languages, algorithms ported to circuits are still known as hardware algorithms, and coprocessing circuits are still known as the hardware in hardware/software systems. But with the FPGA market generating billions in annual revenue from the sale of hundreds of millions of devices to thousands of customers, the categorizing of circuits as hardware is grossly inaccurate.

EFFECT ON COMPUTING FIELD

That circuits on FPGAs are actually software rather than hardware isn't just an insignificant technicality, like a tomato actually being a fruit rather than a vegetable. On the contrary, treating circuits as hardware hurts the computing field, especially embedded systems. The reason is that most computing-oriented engineers—software designers—aren't interested in learning about hardware design. Hardware to them is a different beast. Thinking that circuits are hardware, computing-oriented engineers aren't motivated to learn to design for FPGAs, and perhaps even more significantly, computing-oriented college students often avoid courses that involve circuits and FPGAs, due to not being "hardware types."

Temporal versus spatial modeling

A typical computer science department will have computing-oriented students with programming specializations in graphics, the Web, or real-time system development, for example. But rarely do computer science students specialize in FPGA development; those students are typically in electrical engineering departments, where "hardware" is the focus.

The issue actually extends well beyond circuits and hardware and into the realm of broadening the concept of computation models to consider both temporally and spatially oriented models, a concept illustrated in Figure 1. Developing modern embedded software capable of executing on multiprocessor and FPGA platforms requires expertise not just in temporally oriented modeling (*W* comes after *X*) like writing sequential code but also in spatially oriented modeling (*Y* connects with *Z*) like creating circuits.

The former involves ordering tasks in time, while in the latter, tasks execute concurrently in a parallel or

Tried any new gadgets lately?

Any products your peers should know about? Write a review for *IEEE Pervasive Computing*, and tell us why you were impressed. Our New Products department features reviews of the latest components, devices, tools, and other ubiquitous computing gadgets on the market.

Send your reviews and recommendations to pvcproducts@computer.org today!

www.computer.org/pervasive

pipelined fashion. Sequential coding languages like C excel at capturing temporally oriented computations. Designers traditionally capture spatially oriented computations in hardware description languages, but they also capture them using extensions to sequential languages, such as Posix threads or even SystemC's explicit circuit extensions to C++. In fact, sequential code coupled with knowledge of how a synthesis tool will convert that code to a circuit can capture many spatially oriented computations. Circuit designers do this all the time when they write loops that they know will be unrolled, for example.

Software developers tend to excel at the temporally oriented task of creating sequential code models, like defining algorithms, subroutines, and so on. But they're typically weaker at creating models that also involve some amount of spatial orientation, like parallel processes, data-flow graphs, or circuits, largely because computing education in universities tends to emphasize the former with little attention given to the latter. Yet with embedded systems continuing to grow in importance, such imbalance can't persist much longer.

Introducing students to circuits

What can be done? For starters, computing departments in universities might reconsider how they introduce circuits to their students. Most computing curricula today introduce circuits as the foundation of how microprocessors work, and perhaps as glue logic for interfacing physical components. But the modern computing curriculum should also introduce circuits early on as a model for describing computations, representing a highly parallel alternative to a sequential code model.

For example, a temporally oriented approach to sorting *N* data items makes several sequential passes over the data, swapping certain data items along the way. A spatially oriented approach might use *N*/2 components to order pairs of data, followed by *N*/4 components to order quadruples of data, and so on in a pipelined manner. The ways of thinking of those two computation approaches differ dramatically. With the advent of multiprocessors and FPGAs, who's to say that the spatially oriented way is any less important?

Developing students' spatial thinking via computing-oriented circuit design, when coupled with the existing developing of their temporal thinking via traditional sequential programming, could lead to students more easily learning parallel-programming concepts, something many writers in recent years insist must be strengthened. Parallel-programming concepts are really a hybrid of temporal and spatial thinking. Building both temporal and spatial skills before embarking on parallel programming can be thought of as building both the left and right wings of a plane before attempting to fly.

Meanwhile, along with changing university curricula, we all (whether in universities or companies) can stop calling circuits hardware, and we also can broaden our use of the word software beyond just microprocessors. In this way, circuits and other modeling approaches with a spatial emphasis can assume their proper role in the increasingly concurrent computation world of software development. ■

Frank Vahid *is a professor in the Department of Computer Science and Engineering at University of California, Riverside. Contact him at vahid@cs.ucr.edu.*

Editor: Wayne Wolf, School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta; wayne.wolf@ece.gatech.edu

IEEE

Innovative Technology for Computer Professionals

Computer

Welcomes Your Contribution

***Computer* magazine looks ahead to future technologies**

- ***Computer***, the flagship publication of the IEEE Computer Society, publishes peer-reviewed technical content that covers all aspects of computer science, computer engineering, technology, and applications.
- Articles selected for publication in ***Computer*** are edited to enhance readability for the nearly 100,000 computing professionals who receive this monthly magazine.
- Readers depend on ***Computer*** to provide current, unbiased, thoroughly researched information on the newest directions in computing technology.

To submit a manuscript for peer-review, see *Computer's* author guidelines:

www.computer.org/computer/author.htm

IEEE computer society

Continued from page 112

components of the domain names that create the Web's upper-level structure.

From this, it can be seen that the Internet could well provide a separate DNS for each writing system without compromising, and maybe even helping, its end-to-end model and unique binary naming. This would effect a World Wide Web for each writing system, and I strongly believe this should be done. Indeed, it will probably happen anyway in the long run, but in a drawn out, unmanaged, and costly way.

If each writing system has its own Web, then each will have its own Web software such as browsers and searchers. These will be simpler because the needs will be simpler, and it will be possible to give better and more specific support to each system's typographical and compositional aspects. Webs can be practically supported and gradually developed for minor writing systems, systems that wouldn't have a chance were there to be a Universal Web. Anyone wishing to work in two writing systems would need only two sets of relatively simple Web software rather than one set of grotesquely complex software riddled with feature bloat.

Software needing to mix writing systems, for educational or scholastic use, could use markup for system switching and formatting, but this need not involve URLs and domain names. In any case, users could mix URLs for different Webs if needed because each DNS would translate the domain names for the different Webs into the underlying Internet addresses.

CULTURAL SOFTWARE

Many arguments favor providing separate support for different writing systems. Neglecting these arguments is tragic, but in the case of ICANN, it only continues the computing world's impoverishing of writing cultures, which has persisted since the early years of electronic computing.

In *Coded Character Sets, History and Development* (Addison-Wesley, 1980), Charles Mackenzie describes well the beginnings of the Latin writing system's sad story when brought under computation. Early printers could only use capital letters and a few special characters, all but one of which (the lozenge) were needed for commercial use in names and addresses, product names, and the like. Scientists and engineers using programming languages like Fortran were thus not only restricted to capital letters but also had only the hyphen as a basic mathematical symbol. This led to the replacement of the traditional arithmetic symbols by commercial ones: multiplication's saltire (×) by the asterisk (*), division's obelus (÷) by the virgule (/), and even addition's plus (+) by the ampersand (&), although users could pay extra money to get printer features that replaced ampersands with plusses.

Many arguments favor providing separate support for different writing systems.

The banditry of the computing industry and its profession became more pronounced when its developers introduced upper- and lower-case alphabets. While they still ignored the multiplication and division symbols, developers introduced a quite impoverished set of special characters in both ASCII and EBCDIC, mainly because the impact line printers of the time could deal only with a limited numbers of fixed-size characters.

When personal computers came along, developers adopted the limited ASCII character set, even though the PCs were accompanied by dot matrix printers that were not inherently limited like line printers and typewriter terminals. Many PC users put far more text onto their display screens than out to their printers anyway, and the screens also used dot matrices. The profession therefore imposed the PC's typographical poverty, not the hardware, particularly when cheap ink-jet and laser printers became common.

This was the theme: improve the graphics, colors, and images, but leave the character set in the dark ages of early digital computing. Capabilities available to old-fashioned letterpress printers were never passed on to ordinary text users.

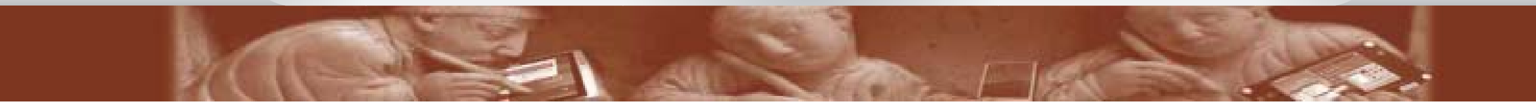
Thus, I cringe when I see H2O and CO2 almost everywhere nowadays, online and in print. Reading technical texts, I shudder at a^2, 45uF, and >=. Even when I can read good online text with proper opening and closing quotes and em dashes, pasting those marks into my vi editor brings it to a sudden stop. When I get e-mail from Europe, the names are often difficult to interpret when they include special alphabetic characters. Although I can, with difficulty, include × and ÷ in the Word version of this essay, they aren't easy to get to, and getting them safely across to the final copy presents a challenge to the editorial staff.

Indeed, the neglect by computing professionals of the Latin writing system's culture brings about a multitude of problems. The Latin writing system was, until taken over by digital techniques, a rich graphical cultural artifact. With digital technology it could have been made even richer, but it has instead been made poorer through, in a word, theft.

The poverty is not just a property of the coding system. We have also been completely barbaric to culture with our keyboards. As a tiny example, there are three peculiarly chosen old-fashioned accents on my keyboard— ~ ` ^ —but I can't use them to key señor or caffè or Côte d'Azur in here, and certainly not café. It's ridiculous. And the Accenture people have it even worse.

With digital technology, we could have a single simple keyboard and accompanying software that would allow simple and complete support of all languages and disciplines that use the Latin alphabet (eprints.utas.edu.au/1564). It's just a matter of using our writing system's rich graphical culture and tradition. What we have now disgraces our profession.

Computer Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page



OTHER CULTURES

Writing systems other than the one used here have been, if anything, even more culturally impoverished by the computing profession than the Latin alphabetic system.

The Chinese writing system provides perhaps the greatest contrast (wikipedia.org/wiki/Chinese_character). It's a very old system and has the wonderful advantage of being independent enough of the spoken languages that use it to let people who speak mutually unintelligible languages write intelligibly to each other. It's also faster to read and more economical of space than linear alphabetic writing systems.

In the computer age, however, users are crippled with qwerty keyboards, for which there are two main methods of keying in the Chinese (hànzì) characters: the pīnyīn alphabetic method and the wŭbĭzìxíng root method.

Under the pīnyīn method, the user keys in the Romanization of the official language, Mandarin, and software converts the syllables to characters. This is easy to learn, provided you know Mandarin. However, it's rather slow, especially if there's much need for disambiguation, which is only too likely as there are typically several characters for any of the relatively few pīnyīn syllables.

The wŭbĭzìxíng method takes advantage of the Chinese characters' graphical nature. Many characters are written with two or more separate components (roots) in a formal sequence, and dictionary sequence traditionally relies on this structure. Wŭbĭzìxíng keying also uses the structure and so is much faster than pīnyīn and usable by speakers of languages other than Mandarin. However, it too is superimposed on the qwerty keyboard and, because there are several hundred distinct roots, is so hard to learn that few take the trouble.

Clearly, a hànzì keyboard with, say, 64 root keys and three shift keys for each hand would make wŭbĭzìxíng style keying yet faster and much easier to learn.

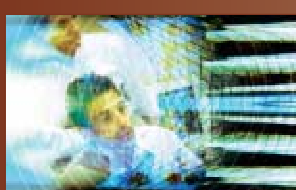
But the qwerty keyboard is not the only cultural theft of the Chinese computing profession. All the coding schemes in use represent characters, not roots. Not only does this make dictionary sequencing difficult, it also makes the introduction of new characters impractical. Root encoding would free up both the written and spoken language for change, and it could even eliminate the need for ugly insinuation of alphabetic words into hànzì text.

The computing industry has savagely attacked written languages. The computing profession is gravely at fault in allowing this to happen. However, it's not too late to make amends. A good place to start would be to kill the Universal Web. ■

Neville Holmes *is an honorary research associate at the University of Tasmania's School of Computing. Contact him at neville.holmes@utas.edu.au. Details of citations in this essay, and links to further material, are at www.comp.utas.edu.au/users/nholmes/prfsn.*

The Profession as a Culture Killer

Neville Holmes, University of Tasmania



Having a World Wide Web for each writing system would allow much better software than would a Universal Web.

Literate cultures are based on writing systems and print. In the early years of digital computing, machinery had a limited ability to print, but it didn't matter culturally as that printing almost entirely served to create commercial documents such as invoices and sales summaries. There were problems, though, in specialized areas like scientific programming.

Problems loomed larger in the mid-1960s when character sets with upper- and lower-case alphabets came into use on computers and in telegraphy. This enabled the processing of running text, but the typography was extremely crude. The computing profession and industry, however, saw this crudeness as simplicity and promulgated separate sets of similar crudity to cater to other cultural needs.

When personal computers became widespread, the computing industry—with the unthinking acquiescence of the computing profession—simply moved one of the main collections of crudities, the ASCII family, onto those machines. When the pressure for cultural extension grew impossible to ignore, a monster crudity, Unicode, subsumed the various small crudities. My essay entitled "Toward Decent Text Encoding" (*Computer*, Aug. 1998, pp. 108-109) advocated a quite different approach, one that could have been adopted as one of Unicode's subsets, but approaches like that were, to my knowledge, not seriously considered.

To my dismay, I recently read in a local newspaper that ICANN, the Internet Corporation for Assigned Names and Numbers, would begin allowing non-Latin characters in domain names. Getting it done will be immensely complicated, and the result will likely be chaotic. Worse, it shows a complete disregard for mankind's second greatest digital technology: writing.

I have a professional responsibility to protest against these developments, although I don't expect my protest to affect anything.

THE INTERNET

It turns out that ICANN, had started "the internationalization of the Internet's domain names" with its resolution of 25 September 2000 (icann.org/topics/idn) that it "must be … fully compatible with the Internet's existing end-to-end model and … preserve globally unique naming in a universally resolvable public name space."

This resolution seems rather strange. For example, the domain names are already international, and not only in having two-letter top-level domain names (TLDNs) for each country, from which Tuvalu profits greatly. What ICANN seems actually to have meant is to enable mixing of writing systems Unicode-style in domain names.

The delay in implementing this change is not surprising. Even confined to the Latin writing system, the domain name system seems to be lurching into chaos, what with commercial exploitation under the wonky protection of trade name legislation and the cutting loose of constraints on TLDNs.

The Internet is distinct from the Web, even though the Web is based on the Internet. The Internet is defined by the Internet Protocol (IP) binary names or addresses used for directing packets of data along paths within it. The unfortunate slowness of the transition from its IP version 4 to version 6 has confounded this end-to-end model. Its naming is, in the sense of being not necessarily fixed, no longer unique because there are too few IPv4 addresses to go around.

The Web is defined by its use of somewhat meaningful alphabetic domain names, which are used within uniform resource locators (wikipedia.org/wiki/Uniform_resource_locator) to point to where its resources are stored on the Web. In the URL *icann.org/topics/idn* the *icann.org* is the name of the "domain" where the resource *topics/idn* is stored.

When a program such as a Web browser needs to use a resource, it must have the Internet translate the domain name into an IP name so that the Internet can support the browser's use of the Web. To do this translation from meaningful name to binary name, the Internet provides a Domain Name System (wikipedia.org/wiki/Domain_name_system).

Thus the Internet uses culture-free binary names to manage its traffic, while the software that manages the Web uses cultural tokens as

Continued on page 110

Up-to-date, Relevant Information
Driving the Bottom Line
Fueling Imagination

"Findings indicate that IEEE journals are getting the newest, most revolutionary ideas in increasing numbers."

– Dr. Donald R. Scifres, holder of more than 140 patents, founder SDL Ventures, LLC

From Imagination to Market

Access the latest technical information from IEEE and give your team an edge on the competition.

- Periodicals and conference proceedings that define the future of innovation
- Over 1.5 million documents in the IEEE *Xplore*® digital library
- Top cited journals in the field

Free Trial!

Experience IEEE – request a trial for your company.
www.ieee.org/innovate

IEEE Information Driving Innovation

IEEE

NEWEST TITLES IN COMPUTING FROM WILEY & IEEE COMPUTER SOCIETY PRESS

The Web's Awake

An Introduction to the Field of Web Science and the Concept of Web Life

Philip D. Tetlow

978-0-470-13794-9 • April 2007 • Cloth
List: ~~$49.95~~ • Special: $39.96 • Wiley-IEEE Press

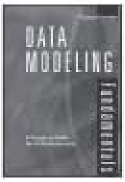
Has the World Wide Web evolved into a new life form? The author of this provocative book, Philip Tetlow, presents a very compelling argument that it indeed has. Drawing from theories originating in the natural sciences, mathematics, and information technology, *The Web's Awake* explores how the continued growth and increasing complexity of the Web are quickly outstripping our capability to control it. In other words, the Web has quite literally taken on a life of its own.

Emergent Information Technologies and Enabling Policies for Counter-Terrorism

Robert L. Popp, John Yen

978-0-470-77615-4 • May 2006 • Cloth
List: ~~$121.95~~ • Special: $97.56 • Wiley-IEEE Press

This book provides an examination of key Twenty First-Century technologies and corresponding enabling policies for countering terrorism – it is critical to consider both technology and policy together since the two are strongly interdependent and realistic change will not occur without considering them jointly. The chapters are written by nationally-recognized highly-regarded authorities and experts on counter-terrorism information technologies and policy, providing a practical reference for those charged with protecting our nation from terrorist attacks while protecting our civil liberties.



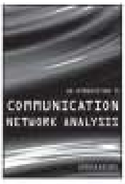
Data Modeling Fundamentals

A Practical Guide for IT Professionals

Paulraj Ponniah

978-0-471-79049-5 • June 2007 • Cloth
List: ~~$110.00~~ • Special: $88.00

This book provides a practical approach for IT professionals to acquire the necessary knowledge and expertise in data modeling to function effectively. It begins with an overview of basic data modeling concepts, introduces the methods and techniques, provides a comprehensive case study to present the details of the data model components, covers the implementation of the data model with emphasis on quality components, and concludes with a presentation of a realistic approach to data modeling. It clearly describes how a generic data model is created to represent truly the enterprise information requirements.

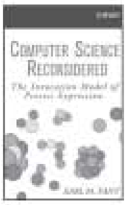


An Introduction to Communication Network Analysis

George Kesidis

978-0-471-37141-0 • June 2007 • Cloth
List: ~~$76.95~~ • Special: $61.56 • Wiley-IEEE Press

Introduction to Modeling and Performance Evaluation of Communication Networks is a quantitative text, which focuses on the real issues behind serious modeling and analysis of communications networks. The author covers all the necessary mathematics and theory in order for students to understand the tools that optimize computer networks today. The first three chapters form the mathematical section while the last three chapters explore networking.



Computer Science Reconsidered

The Invocation Model of Process Expression

Karl M. Fant

978-0-471-79814-9 • June 2007 • Cloth
List: ~~$89.95~~ • Special: $71.96

The Invocation Model of Process Expression argues that mathematics does not provide the most appropriate conceptual foundations for computer science, but, rather, that these foundations are a primary source of unnecessary complexity and confusion. It supports the argument that there is a more appropriate conceptual model that unifies forms of expression considered quite disparate and simplifies issues considered complex and intractable. This book presents that this model of process expression is an alternative theory of computer science that is both valid and practical.

Engineering Your Retirement

Retirement Planning for Technology Professionals

Mike Golio

978-0-471-77616-1 • November 2006 • Paper
List: ~~$34.95~~ • Special: $27.96 • Wiley-IEEE Press

How much money will I need to retire? How long will it take for me to accumulate it?

Engineering Your Retirement answers these and many more questions for the technical professional contemplating retirement.

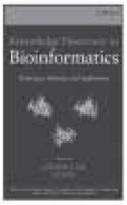
Automated Defect Prevention

Best Practices in Software Management

Dorota Huizinga, Adam Kolawa

978-0-470-04212-0 • August 2007 • Cloth
List: ~~$89.95~~ • Special: $71.96
Wiley-IEEE Computer Society Press

Focusing on a new methodology called Automated Defect Prevention (ADP), *Automated Defect Prevention: Best Practices in Software Management* is a practical and thorough guide to implementing and managing software projects and processes. Through the use of ADP, the author team delivers a comprehensive guide to effective project management and to more reliable and higher quality products. The text provides details on how to implement defect prevention strategies across a group and ensure that group members are following the prescribed practices.



Knowledge Discovery in Bioinformatics

Techniques, Methods, and Applications

Xiaohua Hu, Yi Pan

978-0-471-77796-0 • June 2007 • Cloth
List: ~~$99.95~~ • Special: $79.96

Knowledge Discovery in Bioinformatics Techniques, Methods and Applications brings together the ideas and findings of data mining researchers and bioinformaticians by discussing cutting-edge research topics, such as gene expressions, protein/RNA structure prediction, phylogenetics, sequence and structural motifs, genomics and proteomics, gene findings, drug design, RNAi and microRNA analysis, text mining in bioinformatics, modeling of biochemical pathways, biomedical ontologies, system biology and pathways, and biological database management.

ORDERING INFORMATION
1-877-762-2974 North America
+ 44 (0) 1243 843294 Rest of World
Log on to www.wiley.com/ieeecs
Enter Promotion Code CMP97 to receive 20% off featured titles at checkout.

WILEY Publishers Since 1807

IEEE

IEEE computer society