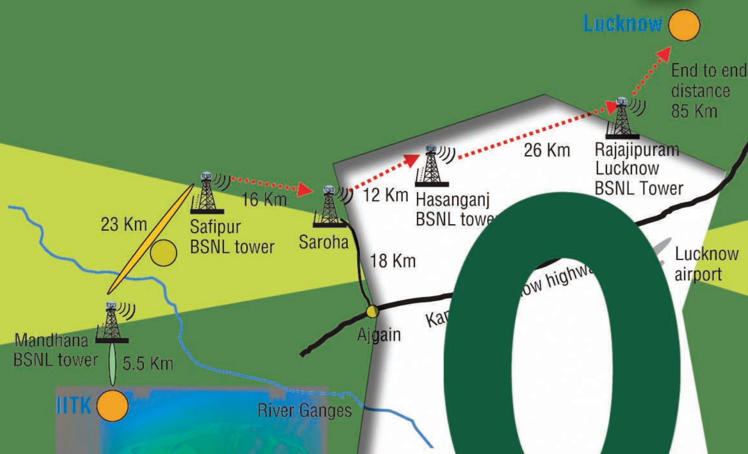
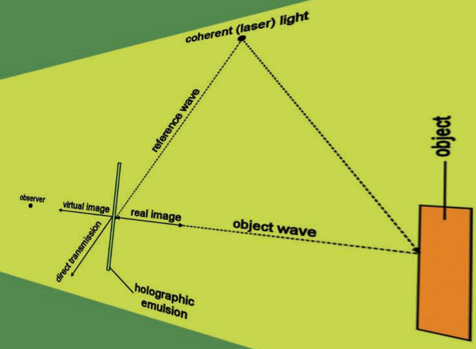
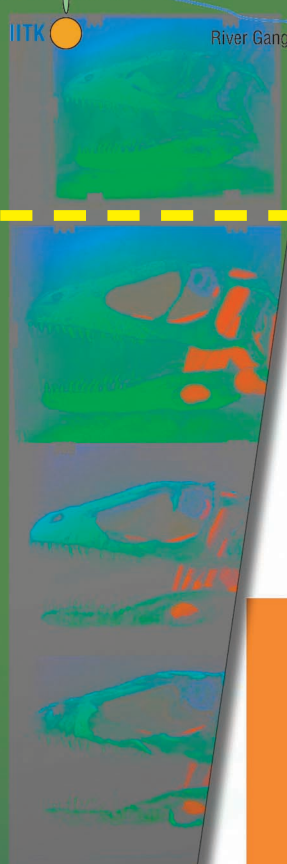
Innovative Technology for Computer Professionals

Computer

January 2004







OUTLOOK

IEEE Computer Society President's Message

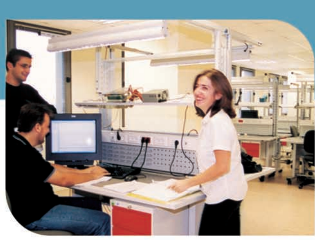
Design Fragility

Hybrid CPU/FPGA Chips

IEEE

IEEE Computer Society

http://www.computer.org



FIRST CALL FOR PAPERS

IEEE International Conference on Cluster Computing

http://grail.sdsc.edu/cluster2004/

CLUSTER 2004

IEEE

IEEE Task Force on Cluster Computing TFCC

photo credit: David Emig

SAN DIEGO, CALIFORNIA, U.S.A. 20-23 SEPTEMBER 2004

COMMITTEES

General Co-Chairs
Henri Casanova
University of California, San Diego, USA
Daniel S. Katz
Jet Propulsion Laboratory, Caltech, USA

General Vice Chair
Bernard Tourancheau
Sun Microsystems, France

Program Chair
Philip Papadopoulos
San Diego Supercomputer Center, USA

Deputy Program Chair
Rémy Evard
Argonne National Laboratory, USA

Program Vice Chairs
Carl Kesselman - Middleware
Information Sciences Institute & University of Southern California, USA
Satoshi Matsuoka - Hardware
Tokyo Institute of Technology, Japan
David Walker - Applications
Cardiff University, UK

Tutorials Chair
Jennifer M. Schopf
Argonne National Laboratory, USA

Workshops Chair
Rajkumar Buyya
University of Melbourne, Australia

Exhibits/Sponsors Co-Chairs
Jim Ang
Sandia National Laboratories, USA

Publications Chair
Joseph C. Jacob
Jet Propulsion Laboratory, Caltech, USA

Publicity Chair
Bertil Schmidt
Nanyang Technological University, Singapore

Posters Chair
Richard Barrett
Los Alamos National Laboratory, USA

Finance/Registration Chair
Vana Kalogeraki
University of California, Riverside, USA

Local Arrangements Chair
Greg Bruno
San Diego Supercomputer Center, USA

In less than a decade, commodity clusters have become mainstream and cost-effective computing platforms for executing a wide range of computation-, data-, and transaction-oriented parallel applications. In spite of countless deployed platforms and resounding application successes, many research and development challenges remain for achieving higher performance, scalability, and usability. The IEEE TFCC Cluster2004 conference, to be held in beautiful San Diego, provides an open forum for researchers, practitioners, and users to present and discuss issues, directions, and results that will shape the future of cluster computing.

Cluster2004 welcomes paper and poster submissions from engineers and scientists in academia and industry describing original research work in all areas of cluster computing. In addition, Cluster2004 will welcome proposals for tutorials and workshops to be held concurrently with the conference. Topics of interest include, but are not limited to:

Cluster Middleware
- Single-System Image Services
- Software Environments and Tools
- Standard Software for Clusters
- I/O Libraries, File Systems, and Distributed RAID

Applications
- Scientific and E-Commerce Applications
- Data Distribution and Load Balancing
- Scheduling and Parallel Algorithms
- Innovative Cluster Applications
- Tools and Environnments

Cluster Networking
- High-Speed System Interconnects
- Lightweight Communication Protocols
- Fast Message Passing Libraries

Performance Analysis and Evaluation
- Benchmarking and Profiling Tools
- Performance Prediction and Modeling
- Analysis and Visualization

Cluster Management and Maintenance
- Cluster Security and Reliabiity
- Tools for Managing Clusters
- Cluster Job and Resource Management
- High-Availability Cluster Solutions

Grid Computing and Clusters
- Grid / Clusters Integration
- Network-Based Distributed Computing
- Mobile Agents and Java for Clusters/Grids

PAPER SUBMISSION

Paper Format: Not to exceed 25 double-spaced, numbered pages (including title, author affiliations, abstract, keywords, figures, tables and bibliographical references) using 12 point font on 8.5x11-inch pages with 1-inch margins all around.

Submission: An on-line submission page will be activated on the conference web site two weeks before the submission deadline. Authors should submit a PostScript (level 2) or PDF file. Hard copy submissions will not be accepted.

JOURNAL SPECIAL ISSUE

Authors of the best papers from the conference will be invited to submit an expanded version of their papers for possible publication in a special issue of the International Journal of High Performance Computing and Networking (IJHPCN), which will appear in 2005.

IMPORTANT DATES

Workshop Proposals Due:	March 6, 2004
Paper Submissions Due:	April 16, 2004
Tutorial Proposals Due:	May 14, 2004
Paper Acceptance Notification:	June 1, 2004
Exhibition Proposal Due:	June 18, 2004
Final Paper Manuscripts Due:	July 2, 2004
Pre-Conference Tutorials:	**September 20, 2004**
Conference:	**September 21-22-23, 2004**

Seapine Software™
changing the world of software development

Developers love creating code... Managers crave process control... Bridge the gap with Seapine CM.

Software development is a team effort with developers, testers, and management all working toward one goal – delivering the highest quality product on time.

Built on award-winning TestTrack Pro and Surround SCM, Seapine CM brings structure to source control and issue management, improving communication while accelerating product development.

Seapine CM helps your team...

Define custom change request workflows, putting you in control of who makes changes and who authorizes the closure of issues.

Associate source code changes with defects or change requests.

Gain a thorough understanding of how much work remains before project completion.

View complete audit trails of what changed, why, and by whom.

Understand how close you are to release—how many issues are open, how quickly are you closing them, how many are re-opened?

Successful team-based development requires the proper process supported by the right development tools. Tools that are flexible, easy to use, secure, and scalable–like Seapine CM.

Features:

Complete source code control with private workspaces, automatic merging, role-based security and more.

Comprehensive defect management—track change requests, bug reports, feature requests and more.

Fast and secure remote access to your source files and defects—work from anywhere.

IDE integration with JBuilder, Visual Studio, Dreamweaver, and other leading development tools.

Advanced branching, triggers, email notifications, fully configurable workflows, and customizable fields put you in complete control of your process.

Open interfaces—triggers, email notifications, SOAP support, XML data exchange, and ODBC clients.

Scalable and reliable cross-platform, client/server solution supports Windows, Linux, Solaris, and Mac OS X.

Achieve major improvements in software development performance through better tool integration and process automation. Manage defects, development issues, and change requests with award-winning TestTrack Pro and gain complete control over your source code and change process with Surround SCM. Seapine's integrated change management tools are feature rich, highly scalable, Web enabled, and cross platform. Streamline your development process with Seapine CM and help your team deliver quality software products on time, every time.

Learn more about the Seapine CM suite at **www.seapine.com** or call 1-888-683-6456

Surround SCM Source Control

TestTrack Pro Defect Management

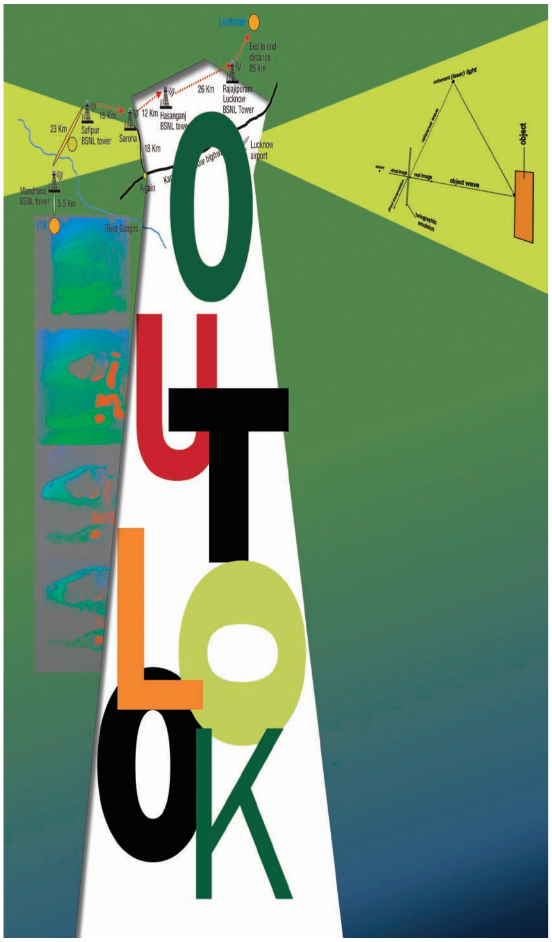
QA Wizard Automated Testing

software development 13th annual product excellence award JOLT

Innovative Technology for Computer Professionals

Computer

January 2004, Volume 37, Number 1



Cover design and artwork by Dirk Hagner

ABOUT THIS ISSUE

In this January Outlook issue, we continue our tradition of publishing articles that address advances with near-term possibilities. Topics covered include the ITRS assessment of the challenges awaiting the semiconductor industry; NASA's software reliability efforts at JPL and the Ames Research Center and its systems and software architecture for next-generation unmanned spacecraft; an innovative initiative for providing inexpensive wireless connectivity for rural populations in developing nations; an assessment of global IT employment prospects in 2004; and a novel system for combining optical holograms with interactive computer graphics. This issue also includes two essays on the future of software engineering and the end of science.

PERSPECTIVES

COVER FEATURES

Computer
IEEE Computer Society: http://computer.org
Computer: http://computer.org/computer
computer@computer.org
IEEE Computer Society Publications Office: +1 714 821 8380

OPINION

NEWS

MEMBERSHIP NEWS

COLUMNS

NEXT MONTH:
Ad Hoc Networking

DEPARTMENTS

Membership Magazine of the IEEE Computer Society

Innovative Technology for Computer Professionals
Computer

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 $14 for subscription of *Computer* magazine ($14 for students). Nonmember subscription rate available upon request. Single-copy prices: members $10.00; nonmembers $20.00.

Postmaster: Send undelivered copies and address changes to *Computer*, IEEE Service Center, 445 Hoes Lane, Piscataway, NJ 08855. Periodicals Postage Paid at New York, New York, and at additional mailing offices. Canadian GST #125634188. Canada Post Publications Mail (Canadian Distribution) Agreement Number 0487910. Printed in USA.

A Tale of Three Disciplines … and a Revolution

pp. 30-36

Jesse H. Poore

Despite years of hard work and dedicated champions, software engineering's educational and industrial constituents have not only failed to converge, the gap between them continues to grow.

Some disciplines, like circuit and genetic engineering, seem to evolve from theory to practice relatively responsibly. Software engineering, on the other hand, while not yet at the guillotine, has suffered a decided lack of direction. It may be time to storm the gates. It is never too late for a revolution, and circuit and genetic engineering provide two worthy role models.

The End of Science Revisited

pp. 37-46

John Horgan

Scientists need a certain degree of faith to bolster their confidence in the arduous quest for truth; lacking such faith, science would not have come so far so fast. But when researchers reflexively deny any evidence and arguments that challenge their faith, they violate the scientific spirit.

Although we have grown up in a period of explosive scientific and technological progress, reflected by such measures as Moore's law, science—especially pure science—might be entering an era of diminishing returns. Science itself tells us that there are limits to our knowledge.

2003 Technology Roadmap for Semiconductors

pp. 47-56

Don Edenfeld, Andrew B. Kahng, Mike Rodgers, and Yervant Zorian

Today, microprocessor advances abound: general-purpose digital microprocessors for personal computers have been joined by mixed-signal systems for wireless communication and embedded applications. Battery-powered mobile devices are replacing wall-plugged servers. SoC and system-in-package designs are supplanting in-house, single-source chip designs.

Software today can account for 80 percent of an embedded system's development cost, test cost has increased exponentially relative to manufacturing cost, and verification engineers outnumber design engineers on microprocessor project teams. Overcoming the many existing design technology gaps will thus require a concerted effort by the entire industry.

NASA's Reliable Software Mission

pp. 59-68

Patrick Regan and Scott Hamilton

Both predictable and unpredictable hazards await the spacecraft, robots, and scientific instruments that humans dispatch to explore our solar system. The toughest hazard may be the known presence of unknown bugs in even rigorously tested software.

By exploring new technologies and approaches to develop provably reliable software within tough constraints, NASA has a chance to advance the state of the art, contributing to computer science as well as software engineering. In addition, any successful spin-off that improves reliability while cutting development time and costs could, in principle, generate savings for US industry equal to the nation's budget for space exploration.

IT Employment Prospects in 2004: A Mixed Bag

pp. 69-77

Fred Niederman

Since 1999—when US business magazine cover stories described IT positions going unfilled and extensive congressional lobbying to increase quotas for overseas workers to fill them—the US IT job market has changed drastically, losing more than a million jobs. For US workers, this change raises two different but complementary questions. First, what are the prospects for the global IT workforce in the near and longer term? Second, how will IT jobs be distributed among competing labor markets around the world?

DakNet: Rethinking Connectivity in Developing Nations

pp. 78-83

Alex (Sandy) Pentland, Richard Fletcher, and Amir Hasson

What is the basis for a progressive, market-driven migration from e-governance to universal broadband connectivity that local users will pay for?

DakNet, an ad hoc network that uses wireless technology to provide asynchronous digital connectivity, is evidence that the marriage of wireless and asynchronous service may indeed be the beginning of a road to universal broadband connectivity. DakNet has been successfully deployed in remote parts of both India and Cambodia at a cost two orders of magnitude less than that of traditional landline solutions.

Combining Optical Holograms with Interactive Computer Graphics

pp. 85-91

Oliver Bimber

Holograms can reconstruct complete optical wavefronts, capturing images that have a three-dimensional appearance and can be observed from different perspectives. Museum exhibits often use optical hologram technology because it permits presentation of 3D objects with almost no loss in visual quality. Optical holograms are static, however, and lack interactivity. Combining 3D computer graphical elements with stereoscopic presentation techniques provides an alternative that allows interactivity.

THE PROBLEM WITH AI

I read Neville Holmes's column about artificial intelligence with great interest ("Artificial Intelligence: Arrogance or Ignorance?" Nov. 2003, pp. 120, 118-119).

The family patriarch, a rocket scientist by profession, once said, "Before one can have 'artificial intelligence', one needs 'natural intelligence.'" As Mr. Holmes opined, the implication is that the people attempting to implement AI would first have to understand the fundamentals of human intelligence. It was also a jibe at those who employ the noble moniker to describe trivial mechanisms.

The problem with today's so-called AI is largely that, as Mr. Holmes describes, the processes cannot yet adapt algorithms. While Cellary's assertion regarding the ability to use knowledge to make decisions seems to be on the right track, I doubt that normal people would consider someone who spews facts, yet is unable to assimilate new data, as being intelligent.

The key to having an intelligent process is that the entity has enough sensory inputs and meaningful outputs—a feedback loop as per Norbert Wiener—to validate internally formed theory on reality. Without the ability to train in a substantially closed-loop fashion, new things cannot be learned.

The debasing of the AI concept by ascribing the term to applications unworthy of the name is why AI pioneer Marvin Minsky declared, "AI has been brain-dead since the 1970s." The software industry has relegated "AI technology" to interactive help and video game opponents. Of course, scientists often use terms that don't have the same meaning to the general public. Perhaps AI has become one of those words.

I like the term "algoristics" that Mr. Holmes proposes to distinguish static expert systems from the type of intelligent systems we have yet to implement, and I hope that others will adopt it—

even if only to keep researchers on the same page.

Gerad Welch
Rochester, Minn.
gwelch@computer.org

Neville Holmes responds:
It is perhaps relevant to note that not so long ago, a column called Open Channel used to appear on the back page of *Computer*. Heading it was a quotation ascribed to Charles McCabe of the *San Francisco Chronicle*: "Any clod can have the facts, but having an opinion is an art."

DEFINING MEMORY

In his discussion of AI, Neville Holmes does not sufficiently take into account the results from current brain research. This research, which uses positron emission tomography and magnetic resonance imaging, reveals two distinct types of memory: declarative (memory) and nondeclarative (procedural).

Essentially, declarative memory is what we can express in words or bring to mind as a mental image; it is explicit or conscious memory. In contrast, nondeclarative memory is the collection of skills we acquire with practice or repetition—our habituation or conditioning. For example, artists, musicians, and athletes are masters of specific procedural knowledge. As Larry R. Squire and Eric R. Kandel explain in *Memory: From Mind to Molecules* (W.H. Freeman, 2000), much of what Holmes refers to is procedural memory.

Although these two divisions differ from Howard Gardner's independent dimensions of intelligence that Neville Holmes refers to, there is no conflict between them. They are just different ways to study memory. One is more structure based, while the other explains various memory features.

Human intelligence, call it a problem-solving capacity of any kind, uses both kinds of knowledge. Basically, human intelligence does not really know which knowledge domain it is using. Gardner's multidimensional view of intelligence comes into play here.

What we can put into a computer program directly, as software or as data, is basically of a declarative nature. A piece of software—whether AI or conventional—can collect procedural knowledge, but this occurs indirectly. The software must be prepared for that purpose and must be trained to perform a specific task, such as face or speech recognition.

This procedural knowledge is also a bit restricted by what can be captured as digital data—unlike human intelligence, which uses all knowledge facilities available. This restriction limits a program's intelligence, as does not knowing what data we use.

Jan Giezen
Delft, Netherlands
giezen@tpd.tno.nl

Neville Holmes responds:
The points that Jan Giezen makes all boil down to a fundamental issue that I have raised before: Computing people have much too simple ideas about the human brain ("Would a Digital Brain Have a Mind?" *Computer*, May 2002, pp. 112, 110-111).

To say that human memory is of only two kinds is a gross simplification. I haven't read the book that Giezen cites, but other books that I have read, such as *Memory: Phenomena and Principles* by Norman E. Spear and David C. Riccio (Allyn and Bacon, 1994), emphasize the complexity of human memory, and indeed that of other animals. Popular writings distinguish many other kinds of memory, episodic memory being one that springs to mind.

To equate human memory with intelligence is wrong. Dumb people can have good memories, and smart people can have bad memories. This is where Gardner's work comes in. There are many quite different and relatively independent intelligences. It seems to me that an intelligence in Gardner's sense is a talent for exploiting perception and memory to produce high-quality behavior in a particular area.

DIVISIBLE LOAD-SCHEDULING DISCOVERY

Since the publication of my article, "Ten Reasons to Use Divisible Load Theory" (*Computer*, May 2003, pp. 63-68), I have become aware of an article by R. Agrawal and H. V. Jagadish published in 1988 ("Partitioning Techniques for Large-Grained Parallelism," *IEEE Transactions on Computers*, Dec. 1988, pp. 1627-1634) that, independently of the earliest work mentioned in my article (also published in 1988), discusses divisible load modeling.

Although apparently unknown to many divisible load-scheduling researchers until recently, this article has some noteworthy features and firsts. It models the divisible load-scheduling problem using Gantt-like charts and includes solution reporting time in much the same way that others have done since then. This appears to be the first paper to discuss a linear programming solution (unlike the algebraic solution discussed in our first paper), a proof on the optimal order for solution reporting, and an experimental evaluation of divisible load scheduling. All in all, it is quite a forward-looking paper.

Thomas Robertazzi
Stony Brook, N.Y.
tom@ece.sunysb.edu

We welcome your letters. Send them to computer@computer.org. Letters are subject to editing for style, clarity, and length.

How to Reach *Computer*

Writers

We welcome submissions. For detailed information, visit www.computer.org/computer/author.htm.

News Ideas

Contact Lee Garber at lgarber@ computer.org with ideas for news features or news briefs.

Products and Books

Send product announcements to products@computer.org. Contact computer-ma@computer.org with book announcements.

Letters to the Editor

Please provide an e-mail address with your letter. Send letters to computer@computer.org.

On the Web

Explore www.computer.org/computer/ for free articles and general information about *Computer* magazine.

Magazine Change of Address

Send change-of-address requests for magazine subscriptions to address.change@ieee.org. Make sure to specify *Computer*.

Missing or Damaged Copies

If you are missing an issue or received a damaged copy, contact membership@computer.org.

Reprint Permission

To obtain permission to reprint an article, contact William Hagen, IEEE Copyrights and Trademarks Manager, at whagen@ieee.org. To buy a reprint, send a query to computer@computer.org.

Innovative Technology for Computer Professionals
Computer

Computer in 2004

Doris L. Carver, Louisiana State University

Welcome to the January 2004 issue of *Computer*. My vision for *Computer* in 2004 is that it will continue to serve as a leading information resource in the lifelong learning process of our readers. We remain committed to informing the computing community of the latest advances in research, technology, and applications, as well as providing timely information about professional issues, career trends, news, practices, and perspectives on computing-related topics.

***Doris L. Carver, Editor in Chief,* Computer**

COMPUTER'S COVERAGE

Increasing computational capabilities and decreasing computation costs combine to escalate the demand for complex applications that have far-reaching impacts on society, including new frontiers in medical diagnosis and treatment, weather forecasting, space exploration, e-commerce, and education. Advances in a diverse set of computing areas, such as ubiquitous computing, data integration, self-adaptive systems, and multimodal sensors, are fundamental to realizing these new frontiers. As the computing field continues to thrive and change to meet new challenges, *Computer* continues to evolve to provide coverage of these changes and challenges.

One of these challenges is providing coverage of all aspects of computing that our readers find meaningful. We initiate the process for determining our primary topics with a lengthy session at our summer editorial board meeting. We use the results of that session as the framework for developing our editorial calendar for the following year. Potential topics are assessed from numerous perspectives, including timeliness, breadth of coverage, and relevance to our readers.

I would like to clarify the role that the editorial calendar plays in our editorial process. In particular, I want to inform potential authors that we welcome submissions on any computing-related topic.

Our editorial calendar does not imply that the topics listed there are the only areas in which we wish to publish. Special issues focus on a specific topic while others are theme issues. Special issues are guest edited by experts in the specific area, and theme issues include multiple articles on the theme topic but are not the result of a specific call.

In both special and theme issues, we also include articles on other topics. So I invite and encourage potential authors to submit your manuscripts on all computing-related topics to *Computer*.

Reviewing 2003

Looking back at 2003, our topic coverage was diverse. A feature article in our January 2003 Outlook issue provided coverage of NASA's research in advanced computing technologies to help usher in a new area of autonomous space exploration. This issue also included an article describing NSF's TeraGrid computing infrastructure to support collaborative computational science and access to distributed resources. Another feature article reviewed advances in integrated circuits and MEMS that are enabling neurobiologists to link computer circuitry to neural cells in live animals and study the basis of animal behavior and intelligence.

Other issues published during the year highlighted topics such as commercial workloads, pervasive computing, agile software, piracy and privacy, nanocomputing, handheld computing, Web services, safety-critical systems, and power-aware computing.

Looking ahead to 2004

In this January Outlook issue we continue our tradition of publishing articles that address advances with near-term possibilities.

Topics covered in this issue include the International Technology Roadmap for Semiconductors' assessment of the challenges awaiting the semiconductor industry, especially as mixed-signal and application-driven technologies come to the fore. Another feature article reviews NASA's software reliability efforts at JPL and the Ames Research Center and its ambitious systems and software architecture for next-generation unmanned spacecraft. Other articles describe an innovative initiative for providing inexpensive wireless connectivity for rural populations in developing nations, provide an assessment of global IT employment prospects in 2004, and discuss a novel system for combining optical holograms with interactive computer graphics. Two provoca- tive essays consider the future of soft- ware engineering and ponder the end of science.

Our 2004 planned coverage includes, but is not limited to, computers and the aging; adaptive software and hardware; Internet security; next-generation search; sensor networks; software architectures; and Internet data centers.

Our column and department offerings will continue to include At Random, The Profession, Communications, Embedded Computing, Entertainment Computing, IT Systems Perspectives, Invisible Computing, Security, Standards, Web Technologies, News, Products, and Bookshelf. We also introduce a new column, 32 & 16, that looks back on computing technology and the profession through the lens of *Computer*.

If you have suggestions for other topics, please e-mail them to me at dcarver@lsu.edu.

COMPUTER AS A MEMBERSHIP MAGAZINE

Computer is the primary communication path between the IEEE Computer Society and its members. Through *Computer*, we regularly inform you about initiatives and events that are happening at the Society, such as the Distance Learning Campus (www.computer.org/distancelearning/); the Certified Software Development Professional program (www.computer.org/certification/); the Computer Society International Design Competition (www.computper.org/csidc); the distinguished recipients of the Computer Society awards; new IEEE Fellows; and volunteer opportunities such as editor-in-chief positions for Society publications. We plan to expand coverage of Society initiatives of interest to our readers in 2004.

EDITORIAL BOARD CHANGES FOR 2004

I would like to express my gratitude to the editorial board members whose term of service ended in 2003. My sincere appreciation to Gary Robinson for his service as editor of the Standards column, to Jerzy Rozenblit for his contributions as a member of the Advisory Panel, and to Bill Mangione-Smith for his service as area editor for high-performance computing. Gary, Bill, and Jerzy have provided valuable time, insights, vision, and guidance to the magazine.

We welcome your input and diligently strive to use it as part of our ongoing assessment process.

We welcomed two new members to the Board during 2003. Daniel E. Cooke joined the Board as an area editor for software. Dan, a professor and chairman of the Computer Science Department of Texas Tech University, previously served as the Program Manager for NASA's National Strategic Initiative for Intelligent Systems. Savitha Srinivasan assumed the role of area editor for multimedia. Savitha is the manager of content protection at the IBM Almaden Research Center, where she defines new research areas in content protection and is actively involved with content protection standards activities.

In 2004, Alfred Weaver is moving from his role as area editor of networking and e-commerce to serve on the Advisory Panel. I thank Alf for his dedication as an area editor and look forward to working with him as a member of the panel. My thanks to Dan, Savitha, and Alf for their willingness to contribute their expertise to *Computer*.

AN INVITATION

I invite you to contribute to *Computer* through any of the following methods:

- Submit your manuscript for consideration for publication at *cs-ieee-manuscriptcentral.com*. This site, which contains complete author information and submission details, provides a totally electronic process for manuscript submission and processing.
- Indicate your interest in serving as a reviewer by sending an e-mail message that contains your vitae to *computer-ma@computer.org*.
- Propose a special issue by contacting Bill Schilit, Special Issues Editor, at bill.schilit@intel.com.
- Provide feedback, including suggestions for topics, by sending an e-mail message to dcarver@lsu.edu.
- Visit our Web site at www.computer.org/computer, where you will find a dynamic, value-added extension of the magazine.

I look forward to your participation in 2004!

APPRECIATION

I wish to thank the members of the editorial board and editorial staff for their continued commitment and dedication to achieving *Computer*'s goals. They play a crucial role in making *Computer* happen. The volunteer and staff partnership that drives *Computer* creates a dynamic, fulfilling environment in which to serve as EIC.

To our authors and reviewers, please accept my deepest gratitude for your contributions. *Computer*'s value and success lie with authors who submit their work to us and to reviewers who help ensure that our standards are maintained. The result of our excellent authors and reviewers is that our acceptance rate for articles is approximately 25 percent.

Finally, thank you to our readers. We welcome your input and diligently strive to use it as part of our ongoing assessment process.

I hope you enjoy this January Outlook issue. ■

***Doris L. Carver** is the associate vice chancellor of Research and Graduate Studies and a professor of computer science at Louisiana State University. Contact her at dcarver@lsu.edu.*

Evolving the World with the World's Computer Society

Carl K. Chang, IEEE Computer Society 2004 President

In *Computer*'s first issue for 2004, I extend a warm welcome to the nearly 100,000 new and continuing IEEE Computer Society members. We reaffirm our commitment to offering you those products and services that will most benefit you in your careers as computing and IT professionals.

2003 ACCOMPLISHMENTS

Under the leadership of Past President Stephen L. Diamond, our Society made substantial progress in many areas last year. Our 2003 accomplishments were outlined in *Computer*'s December issue. On behalf of the Society, I wish to thank Steve for his dedicated efforts to develop new initiatives that focused on improving the Society's operational effectiveness. He also guided the advancement of our ongoing electronic projects, including distance learning and the Total Information Provider project.

Another 2003 initiative was the development of the Society's next strategic plan, SP-5. As president-elect, my responsibility was to lead the effort to rethink our future directions, and I engaged all Society Executive Committee members in a forum to create SP-5. This plan builds on a strong foundation established by my predecessors.

MOVING PAST THE GLORIOUS PAST

In recent years, the Society's strategic planning has been guided by its vision statement "to be the leading provider of technical information and services for the world's computing professionals." To accomplish our mission, we have led the IEEE in product and member service innovations.

Carl K. Chang
IEEE Computer Society President

The Society was the first within the IEEE to initiate an online digital library (www.computer.org/publications/dlib); offer distance learning courses as a basic member benefit (www.computer.org/distancelearning); and launch an e-version-only magazine, IEEE *Distributed Systems Online* (http://dsonline.computer.org).

We were the first among peer professional associations to define the body of knowledge of software engineering (www.computer.org/swebok), to offer a certification program for software professionals (www.computer.org/certification), and to launch a technical peer-reviewed magazine on security and privacy (www.computer.org/security). We also are the first IEEE society to provide our members with free access to an online library of reference books (www.computer.org/bookshelf).

While the Society as a whole has accomplished its mission in the past, our latest planning effort convinced us that it is time to move forward with an expanded vision and new strategic activities.

DON'T LEAVE HOME WITHOUT IT: VISA

During my presidential election campaign, I communicated with you about my VISA concept: Vision, Interoperability, Strategy, and Action. I am happy to report that during my year as president-elect, I began formulating VISA with strong support from your volunteer leaders and the Society's executive staff. As a result of this collective effort, we have articulated a new vision for the Society "to be the leading provider of technical information, community services, and personalized services for the world's computing professionals."

We decided to adopt interoperability—defined as organizing and collaborating to deliver an integrated set of service packages across organizational units—as the overarching theme for our planning effort. We then investigated ways of building a sound infrastructure to foster interoperability among all segments of the Society.

Much of our effort in defining new strategies centered on the concept of a community-driven, service-centric infrastructure, with interoperability as the "enabling technology." The Society has built or experimented with several online technical communities, most notably *DS Online* and the *IEEE SE Online*, a new software engineering community to be launched early this year.

Although not all communities are created equal, they do have some common features. A community typically represents a group of individuals with common interests who come together in a common space—in this case a Web-based environment—to collaborate and to exchange information.

The Society must devise better mechanisms to continue supporting technical activities that appear to become more fragmented day by day. The community concept was identified as an effective and efficient method to achieve this end.

By the same token, since our resources are limited, the Society can no longer operate in a production-centric mode, such as continuing to launch periodicals. We must transform our organization into a service integrator with less capital investment and a higher return.

We have made advances toward this goal. Recent examples include the Distance Learning Campus, created in association with KnowledgeNet, and the online bookshelf, powered by Books 24x7. This new Society infrastructure, which may take several years to fully implement, will lay the foundation for us to meet the challenges we will face in the next decade.

SP-5 will be presented for final approval at the February 2004 Board of Governors meeting, and, once approved, it will be ready for action.

2004 APPOINTMENTS AND GOALS

I am privileged to introduce the 2004 Executive Committee, a group of dedicated and capable volunteer leaders. These outstanding professionals and scholars, well known in their individual fields, share the Society's vision and common goals to serve you and the profession.

Joining me in 2004 are Past President Stephen L. Diamond, president and CEO of Picosoft Inc., and President-Elect Gerald L. Engel, a professor of computer science and engineering at the University of Connecticut. Both will interact with me on a daily basis and advise me on many fronts. I feel extremely fortunate to serve as president between the terms of a highly experienced industrial executive and a seasoned university scholar.

First Vice President Lowell G. Johnson, an independent consultant, will chair the newly established Electronic Products and Services Board. This board will create new dimensions of products and services that capitalize on emerging and mature electronic technologies. Opportunities for us to innovate in this area are definitely abundant.

Our vision is for the Society "to be the leading provider of technical information, community services, and personalized services for the world's computing professionals."

Second Vice President Richard A. Kemmerer, a professor and past chair of the Department of Computer Science at the University of California at Santa Barbara, will chair the Chapter Activities Board. Under his leadership, in 2004 the Society will bring chapter members closer to each other and narrow the gap between chapter leaders and the Society.

Oscar N. Garcia, a past president of the Society, who is a professor and founding dean of the college of engineering at the University of North Texas, will continue his faithful service as the secretary of the Board.

Rangachar Kasturi, the Hood Professor and chair of computer science and engineering at the University of South Florida, has made significant achievements while serving as the vice president for publications for the past three years. He will now head the Finance Committee as the Society's treasurer.

James W. Moore, a senior principal engineer for the MITRE Corporation, will continue chairing the Standards Activities Board as a vice president. Jim's goal is to provide a consistent set of Society offerings serving software engineers in their professional development. In addition, Jim wants to create models of membership roles to better structure products to serve our diverse membership.

Christina M. Schober, a product team leader/staff engineer at Honeywell Aerospace Electronic Systems, will be vice president for conferences and tutorials. Chris will continue her efforts to restructure the conference business model to create more incentives for conference organizers.

Murali Varanasi, who is starting an electrical engineering department at the University of North Texas and is a long-time volunteer for accreditation and curriculum development, will serve as vice president of educational activities. In view of the soon-to-be-completed computing curriculum projects (http://www.computer.org/education/cc2001/), including software engineering and computer engineering, Murali intends to plan for more curriculum assessment and development.

Michael R. Williams, professor emeritus of computer science at the University of Calgary and head curator at the Computer History Museum in Silicon Valley, will serve as vice president of the Publications Board. Mike will be instrumental in our ever-increasing effort to build synergy in the publications business with other IEEE entities.

Yervant Zorian, vice president and chief scientist of Virage Logic and chief technology advisor of LogicVision, will continue his appointment as vice president of technical activities. Yervant will focus his energy to help discern and build technical communities while strengthening the base of each existing technical committee and council. He also welcomes your input to help identify emerging technologies.

APPRECIATION AND EXPECTATION

I would like to especially thank Past President Willis K. King for his guidance and encouragement during my on-the-job training as president-elect. Willis shared with me many of his profound thoughts and wisdom, and he was always willing to help me prepare for my presidential term.

Special thanks also are due to Guylaine M. Pollock, who just completed her service as IEEE Division V Director. Key volunteers in the Society know well that Guylaine used her insight and skill to build alliances in the IEEE arena and to keep us engaged in deliberations about many critical issues. The Society benefited tremendously from her efforts.

I would like to thank Deborah K. Scherrer, who directs NASA science and technology-based education and public outreach programs at Stanford University, for her leadership during her term as first vice president for educational activities. Deborah conducted a review of EAB activities in an effort to focus on our educational program priorities.

I also recognize the contributions of Fiorenza C. Albert-Howard, an independent consultant, who just concluded two terms as a member of the Computer Society Board of Governors. I appreciate Fiorenza's participation in many Society activities, including audit, awards, chapters, conferences, and the Constitution and Bylaws Committee.

Doris L. Carver, associate vice chancellor of Research and Graduate Studies and a professor of computer science at Louisiana State University, will continue to lead our flagship magazine, *Computer*, as editor in chief. James D. Isaak, an assistant professor of information technology in the School of Business at Southern New Hampshire University, and Gene Hoffnagle, research and technology strategist, IBM Centers for Advanced Studies, will represent the Society at the IEEE as our IEEE division directors.

In addition, I would like to thank a newly elected member of the Board of Governors, Mark Christensen. Mark joined the industrial advisory board that I assembled back in 1991 when I served as the editor in chief of *IEEE Software*. Since then, he has volunteered in many ad-hoc assignments that were critical to the Society's operations. Most recently, Mark helped us complete a new business model in the SP-5 draft plan.

Last but not least, I would like to thank our staff. I have long been an advocate of volunteer/staff partnership. My favorite example is how I worked with Angela Burgess, now the Society's publisher. Angela and I formed an editor-in-chief/managing editor partnership that transformed *IEEE Software* into a genuine technology-transfer forum.

Bob Care, Violet Doan, Lynne Harris, John Keaton, Anne Marie Kelly, Dick Price, and other staff have all helped me in many different ways. I look forward the opportunity to deepen such partnerships during the year. The leadership of David Hennage, the Society's Executive Director, will be instrumental to our success.

I am looking forward to a productive 2004. We have a new vision. We have assembled an outstanding team of volunteer leaders and staff. Most importantly, we have more than 20,000 volunteers who carry out our mission throughout the year. I applaud those volunteers, for without them, our Society would not have advanced so far. We need their continuing involvement and support to grow our profession and evolve the world. ■

***Carl K. Chang** is chair of the Department of Computer Science, Iowa State University. Contact him at president@computer.org.*

Design Fragility

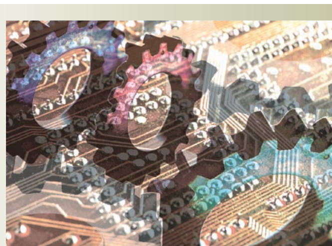
Bob Colwell

One of the things that makes engineering so interesting—besides the fact that they'll pay you to do it—is the interplay between so many contravening influences: features, performance, technology, risk, schedule, design team capability, and economics. You can find a perfect balance among all of these, and still fail, because you designed something the buying public happens not to want.

For instance, if your startup proposes to design the world's best typewriter, the venture capital community probably won't beat a path to your door. It's not 1998 anymore. But if you don't find a workable balance that's at least as good as the one your competitors arrived at, your project is in big trouble. Project-wide blue-screen-of-death. Abort, error, no retry. (Does it worry anyone but me that we all recognize these terms so quickly? This *must* be a bad thing.)

It is possible to achieve fragility without complexity. All you have to do is shave the design margins to the very edge of disaster so that the product is barely achieving its design target. The odd thing is that in many ways this is also the point of maximum operating efficiency and performance, two of the most commercially important aspects of any design. This translates into an inexorable force propelling a design team toward higher output at the expense of design margin, a recipe for complexity.

I believe there is a continuum of feasible design points between a demonstrably safe, solid, and possibly stodgy design at one extreme and a fast, complex, always-on-the-edge-of-disaster design at the other. One of the qualities that change as you move along that line is something I call *fragility*.



Tying conceptually separate functions together increases overall complexity and thus has a real price.

HORSES AND GUITARS

Consider the thoroughbred racehorse. Over hundreds of years, these horses have been bred to do two things: run a race course as fast as possible, and make more horses that are as fast or faster than themselves. The parallel between this sport and the microprocessor business is unsettling. The only differences are that the winning engineers don't get roses, and the losers don't often get turned into glue and gelatin.

But what qualities make a fast horse? Extreme musculature in the right places, a frame of the size and shape to support that musculature, and weight reduced to the bare minimum required by strength-of-materials considerations. Which means that the animal's bones are only just strong enough to get the job done under nominal conditions, thus placing it at greater risk for breakage under any others. A faster horse is a more fragile horse.

Another example comes from my personal experience of having built a guitar that happened not to implode upon initial stringing. Acoustic guitars are small masterpieces of mechanical engineering. They have a resonant cavity—the curvy body of the instrument—for amplifying the acoustical output of vibrating strings. These strings are coupled to the front surface of the resonant cavity via a bridge, a strip of dense material over which the strings bend sharply into the body itself, where pegs hold them in place. The other end of each string is connected to a tuning mechanism that adjusts its tension, and hence its resonant frequency/pitch.

An acoustic guitar with steel strings places approximately 100 pounds of force on the bridge plate—the piece of wood that is glued to the soundboard and has a slot to hold the bridge. Because the string "breaks over" the bridge, this force translates into a torque that tries to lift the bottom edge of the bridge plate and push the string end into the soundboard.

For cultural reasons that cello players and violinists don't understand, guitars are made with mirror finishes, and guitarists are generally extremely sensitive about scratching them because the mirror finish mercilously exposes all such flaws. This same finish lets you see the effects of the string tension on the guitar's soundboard. Between the soundhole and the bridge, a depression occurs in the wood of the soundboard as the torque pushes it down; behind the bridge, the soundboard wood bulges out slightly.

Luthiers face the basic challenge of making the soundboard thin enough to sound good, but thick enough that it can withstand the string torque for at least several decades of constant pull. Their solution is to make the soundboard a thin plane of wood, supported by a system of braces glued to the underside. In acoustic steel string guitars, at least two of these braces begin at the edge of the soundboard and cross in an X pattern, with the intersection artfully placed at the point of highest soundboard deflection just in front of the bridge. The braces are shaved down to the point where they have only enough mass and strength to provide the necessary support.

When finishing the instrument, commercial steel string guitar makers generally apply a thin polymer finish. However, very high-end classical guitars are finished with French polish, a manual-labor-intensive process that builds up a high gloss with an extremely thin layer of lacquer; the finish looks great and the guitar sounds great because there's less mass on the soundboard, but the finish can be scratched off with just a fingernail.

In other words, while inexpensive instruments are built like battleships, the best-sounding ones are, in a word, fragile.

BUILDINGS

Structural engineers and building architects must grapple with similar tradeoffs. In *Why Buildings Fall Down* (Norton, 1987), Mario Salvadori gives examples at both ends of the structural fragility continuum. He relates how a dome roof collapsed in 1960 in Bucharest, Romania, under a surprisingly small load just two years after a new pipe-based method of supporting the dome was tried. Salvadori attributes the failure to a "tendency to design very light dome structures [which] stemmed from the designers' virtuosity and competitiveness, but, more important, from economic considerations deriving from the high cost of materials...."

At the other end of the fragility scale, Salvadori relates an incident in which a building design engineer was called upon to inspect a floor that was sagging beyond its design specs. Upon inspection, the design engineer discovered that the contractor had been using the floor as a temporary storage location for extremely heavy travertine slabs, a load of 15 kN/m^2 on a floor built to code requiring 2 kN/m^2. As Salvadori points out, had the floor collapsed, the design engineer wouldn't have been to blame. But had the floor collapsed when the contractor was moving the travertine across it, he might have been at least partially culpable, even though the code "did not demand such a high live load."

There are no simple answers to where to target a design on the fragility scale.

The question of whether engineers should design strictly according to required specs or attempt to exceed those specs based on their own judgment and morals is a tricky one. As a project manager, I would expect that most design engineers would struggle to hit their stated targets, without allowing much time for anything beyond that. It would also worry me greatly to hear that some project engineers were adding to the product's feature set without having run those features through the project management and feature planning processes. Such activities could easily result in higher project risk.

HELICOPTERS, PLANES, CARS

In *Chickenhawk* (Viking Press, 1984), a harrowing account of his career as a helicopter pilot in the Vietnam War, Robert Mason relates many examples of having pushed his Bell Huey helicopter far beyond its design limits. For example, he once "landed" in a small river so that the rushing waters would clean the cargo deck off, knowing that only the electronics were really susceptible to any damage and that that equipment was high enough to be safe. Since most of the countryside in which Mason had to operate was hilly and heavily forested, his main rotor blades often accumulated both bullet holes and green, leafy vegetation. He had learned to judge from the air how thick the branches were so he would know whether rotor or tree branch would yield first in the event of a close encounter.

Flying machines place a premium on weight, but Mason gives the distinct impression that the Huey was designed to the limits of reliable operation first, resulting in an aircraft that could routinely outperform its specs in the areas that really mattered. In a very real sense, helicopters are fragile machines, but within that design space, the Huey designers got something right. The DC-3 and the A-10 Warthog are other aircraft that have achieved legendary status among their users for overall toughness and trustworthiness in a clinch.

There's no simple formula for achieving a high-performance, competitive product that will pass the test of time like these examples. For instance, in *Searching for Safety* (Transaction Publishers, 1988), Aaron Wildavsky points out how difficult it can be when a litigious public collides with basic physics and the laws of probability.

> A problem with crashworthiness cases is that juries focus on only one part of the automobile, and under only one set of circumstances. Since cars can strike or be struck from any angle, at varying speeds, it is extremely difficult to design a car that affords optimal protection in all types of accidents. In one case, an expert witness testified that injury was enhanced because a seat was anchored and failed to give way. In a different case, experts argued that failure to anchor a seat had led to greater injuries.

Automobile air bags that save many lives but cause death or serious injury in a small percentage of crashes are another example of this phenomenon. There are no simple answers to where to target a design on the fragility scale, other than this one: Don't sacrifice everything to performance. Make all such tradeoffs purposefully.

COMPUTERS

The average performance of computer systems has skyrocketed over the past three decades. We architects get more of the credit for this than we actually deserve. After all, the migration from the single-issue scalar processors of the 1970s to the out-of-order extremely deeply pipelined multithreaded firebreathing "funny cars" of today was achieved by a combination of microarchitecture improvements and process technology advances.

Architecture versus technology

According to *Modern Processor Design* by John Paul Shen and Mikko H. Lipasti (McGraw-Hill, 2003), microarchitecture advances have led from 0.1-instructions-per-cycle machines in 1970 to 2-IPC machines in 2003, a 20-fold improvement. In that same period, clock rates increased from 0.1MHz to today's 3GHz, a 30,000-fold improvement. So who gets the credit for faster chips—the architects or the process technologists?

This isn't quite fair, of course. Had the architects not provided microarchitectures capable of being run at those fast clock rates, numbers like 3GHz would not have been attained. And designing for fast clocks causes a decided loss in efficiency as measured by instructions/clock. Taking that into account, I still score process technology as being at least two orders of magnitude more important than architecture in terms of delivered performance over that time period.

It doesn't really matter. These speed improvements have made today's computers useful for a wide range of activities. But citing a 30,000-fold improvement hides too much detail. If we look behind that curtain, we see that over the same time period, the sheer number of software applications has increased by the same exponential amount, and the average delivered performance varies widely from one application to another.

Exponential complexity

It's not just a matter of language, or coding style, or compiler differences, although they're all important factors. The root cause is that overall computer system complexity has increased exponentially, and complex systems behave in complex ways. Some applications like caches; some like the L2 cache but won't fit in the L1; some map onto the branch prediction scheme perfectly, and a few are pessimal for it.

There's always a cost in tying conceptually separate functions together.

All of this leads to ever-widening variability in the performance extremes of applications on each new, more complex processor. In other words, we're inexorably moving toward the more fragile end of the scale at the system level.

There are ways to spend higher transistor budgets to move to the left, towards safety, on the fragility scale. Some microprocessors have been designed without integer multipliers, for example, because IMUL hardware is large and power-hungry. After all, the instruction set architecture already requires the chip to translate integer to floating point and back again.

I've never liked this idea. It may be faster than not having an IMUL, but not by much. Worse, it surprises programmers and compiler writers. Always remember: Surprises during design aren't evenly distributed into good news and bad news categories. By some as yet undiscovered law of nature, surprises unerringly make a beeline for the bad news camp, from which they proceed to make the project leader pull out more hair. Spend the higher transistor budget in keeping the integer and floating-point sides separated, and many benefits will accrue.

When I was first learning to fix televisions—yes, they were black and white back then—it bothered me that the horizontal sweep and the high voltage circuits were closely intertwined. I know why that was done—it's easier to generate high voltage from an alternating source because a simple step-up transformer followed by rectification suffices.

In terms of understanding televisions in the first place, it's easy to see that something must sweep the beam from top to bottom, something else from left to right, with some means of attracting the beam toward the front of the set from the hot cathode that emits the electrons. But armed only with that understanding, and facing a TV with no raster, you might be misled into thinking the set had a high-voltage problem when really it was the horizontal oscillator that had died.

There's always a cost in tying conceptually separate functions together. Such an expedient increases the overall complexity and thus has a real price.

BAD BREEDING

Reducing this to pithy T-shirt aphorisms, I propose the following formulation:

- Complexity breeds fragility.
- Fragility breeds surprises.
- Surprises are bad.

Strive at all times to maintain a sense of the overall complexity of a given design, and place great importance on minimizing it.

Early in my career, I was riding in a car with the chief architect of a microprocessor that was then in its late design stages, and he was holding his head in both hands, very depressed. When I asked him what was wrong, he said, "I feel I have lost all control on this project. All of the intellectual boundaries we placed on the various units have gradually been compromised. The designers are freely borrowing time and space from each other, or sharing functionality for the sake of die area. Nobody knows how the chip is going to work any more. I'm drowning in a sea of complexity, and I can't see how to get back out of it."

Only through a Herculean project management campaign and many silicon respins was that project salvaged. Insidiously, only the core architects and the project leaders could see the real culprit: the cost of complexity.

One reason that Chuck Yeager was a great test pilot was his innate sense of how close to the edge his aircraft was. He could feel it and hear it, and under emotional pressure that would crush most of us, he could relate that sense to his intellectual understanding of the aerodynamics of flight and take the right action. Boats that are low in the water and responding sluggishly to the helm give the same situational sense to an alert skipper. Cars that are on a slippery road or that are making odd noises are similarly signalling their drivers. In each case, the physics of the situation provides an operational margin that can be of great value.

We in the computer industry have an extra challenge in trying to design less fragile, more forgiving systems. Unfortunately, computer-controlled systems don't naturally fail gracefully. One moment, the computer is monitoring many different sensors and using that information to efficiently control actuators and motors; the next moment, the computer has failed completely and is accomplishing precisely nothing, leaving the chemical plant, or airplane engine, or elevator to run uncontrolled toward maximum entropy.

Graceless failure, often with no warning, is the hallmark of a computer system that was designed around performance targets without regard to system fragility.

The best-sounding guitars are the most fragile, but it's clear from the other examples cited that engineering designs must avoid fragility whenever possible. The real key is keeping the design objectives in sharp focus.

If physics dictates that fragile guitars sound best, then a luthier can direct her efforts to that balance point, and she can educate her customers about the best ways to handle that system weakness. In the end, it's that luthier's customers who will decide whether a better-sounding but less forgiving finish is the right choice.

In our world, if you must design a fragile system, then do it purposely, carefully, and with the same attention to detail that your luthier counterpart would have applied. Remember that unless you actively manage it throughout a design project, system fragility has a way of sneaking up on you. Eschew complexity. Surprises are bad. ■

Bob Colwell was Intel's chief IA32 architect through the Pentium II, III, and 4 microprocessors. He is now an independent consultant. Contact him at bob.colwell@comcast.net.

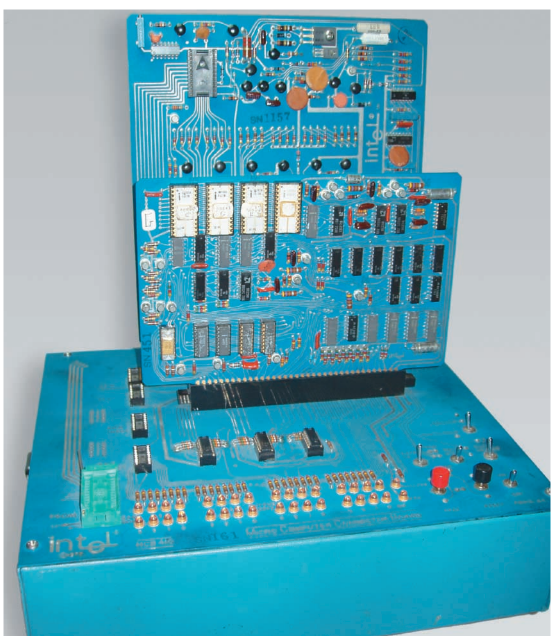
***Editor's Note:* The more we know about yesterday, the better we will be able to deal with today. *Computer* offers this bimonthly column providing excerpts from past issues to serve as a memory jogger for older members and as a perspective creator for newer members.**

JANUARY 1972

PUBLIC CONCERNS. "Addressing an audience of computer specialists, industry executives, management personnel, educators, and user industry representatives, Dr. Bruce Gilchrist, AFIPS Executive Director, stated, 'While 71% of those surveyed believe life is better today because of the use of computers, there is major concern in a number of areas. These include automation and unemployment, increased dependence on computers, dehumanization, the gathering of information on people by large organizations, and the possible misuse of computerized information files.'"

COMPUTER ON A CHIP. "Intel has introduced an integrated CPU complete with a 4-bit parallel adder, sixteen 4-bit registers, an accumulator and a push-down stack on one chip. It's one of a family of four new ICs which comprise the MCS-4 microcomputer system—the first system to bring the power and flexibility of a dedicated general-purpose computer at low cost in as few as two dual in-line packages.

"MCS-4 systems provide complete computing and control functions for test systems, data terminals, billing machines, measuring systems, numeric control systems and process control systems."



Used with permission from Blinkenlights Archaeological Institute

Intel's MCS-4, described in 1972 as the first system to offer a 4-bit parallel adder, sixteen 4-bit registers, an accumulator, and a push-down stack on one chip.

BATTLING TRASH. "Faced with moving a quarter million tons of trash a year off District streets, the Department of Environmental Services, Washington, D.C., has enlisted a land pollution index and an IBM computer to help the city's war on grime. Both the index and computer are part of phase II of Operation Clean Sweep, the city-wide program designed to give District residents clean streets and, at the same time, a new sense of community pride and spirit.

"Department inspectors will match every street against a series of photographs that set cleanliness standards. If a street doesn't measure up, it's reported to the Department so immediate action can be taken. The Department is currently using an IBM System/360 Model 50 to help keep track of the 83 trucks that travel the 165 routes and stop at 135,000 trash pick-up points each week.

"The computer is also saving manpower for the District by cutting the time it used to take to re-configure a route by hand from 15 days to only one."

JANUARY 1988

ELECTRONIC PUBLISHING. "The impact of electronics and modern computer technology on the creation and publication of written information has just begun to be felt. One can safely predict the non-death of paper; the paperless society is just not going to happen. In fact, expect just the reverse. The information glut will continue to expand. The new electronic media won't replace paper, but augment it by providing more efficient and effective access to information. And when that information is found, a paper copy will be made."

BAD ART. "A California graphics design company is sponsoring a 'Bad Art' contest to call attention to what the firm perceives as a lack of design standards in the electronic publishing industry. The 'winning' entries will be hung in the firm's Hall of Shame.

"Bruce Ryon, president of Design Access, said the contest was inspired by the recent flood of 'graphically atrocious' ads created with EP systems.

"'The more hapless efforts... of aesthetically impaired desktop publishing ... should be preserved for posterity, like TV bloopers, man-bites-dog headlines, or those early films of airplanes that never got off the ground,' Ryon said."

COMPUTER ON A CHIP. "Motorola has completed evaluation sampling of the 68030 32-bit microprocessor, called the '030.' The company announced availability of 16- and 20-MHz chips at the same time that it announced development of a 25-MHz version.

"According to the company, the 030 provides twice the performance of the 32-bit 68020 while remaining fully compatible. It is reputedly the first microprocessor to have on-chip data and instruction caches, parallel (Harvard-style) architecture, and dual modes of address." ■

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

Vendors Go to Extreme Lengths for New Chips

Steven J. Vaughan-Nichols

As demand has grown for wireless communications and multimedia devices—including cameras, MP3 audio players, IEEE 802.11 wireless LAN base stations, and cellular phones—so has the demand for specialized embedded chips that can handle these products' need for data-intensive, fast, power-efficient, reliable performance.

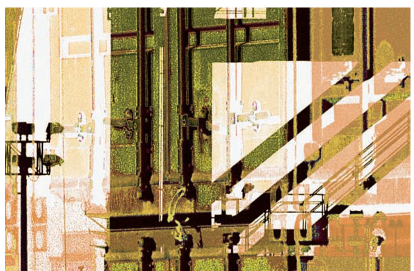
Neither communications nor the rendering of multimedia files tolerates chips that cause transmission delays and interruptions. At the same time, many of these processors must be energy efficient because they operate within battery-powered platforms. This makes traditional CPUs unsuitable in many cases.

To meet the demand for a different type of processor, vendors are beginning to offer unconventional *extreme chips*, explained Max Baron, principal analyst at In-Stat/MDR, a market research firm.

DOING MORE WITH LESS

Driving the development of extreme chips is the huge growth in wireless communications and the resulting demand for millions of mobile chips, said In-Stat senior analyst Tom Halfhill.

However, Halfhill noted, big bucks are not the only motivation. "It's also the engineering challenge," he explained. "Lots of engineers coming out of school want to take experimental designs and try them in the field." Also, he said, many young engineers prefer making new designs at a startup to writing device drivers for a major semiconductor firm.



And it's not just wireless and multimedia technologies that are driving the extreme-chip market. Mike Calise, president of extreme chip maker ClearSpeed Technology, said, "We're seeing an unbelievable number of compute-intensive operations in the biotechnology and scientific fields." For this market, ClearSpeed makes a 32-bit coprocessor for PCI-architecture Pentium class workstations. The chip runs at only 200 MHz but performs at 25.6 Gflops per processor because it uses 64 on-chip floating-point processors.

Traditional CPUs

According to Baron, there are two broad types of chips. The first is made by established companies such as Fujitsu, IBM, Intel, Motorola, and Texas Instruments that have the money to pay for expensive designs. Their processors for PCs and servers generally handle applications, such as Microsoft Word, that perform many different types of tasks and thus execute many types of instructions on relatively small data sets.

Baron explained, "Workloads for the chips made by major manufacturers tend to be instruction intensive. Therefore, a key way to improve these chips is to increase the number of executed instructions per clock cycle. The best way to increase the number of executed instructions per clock cycle is by increasing a chip's frequency."

Parallelization approaches don't work with the types of applications that run on traditional CPUs, Baron said, because the programs usually handle highly interdependent data. With interdependent data, a chip must resolve one group of information before starting on another.

In addition to higher frequencies, Baron said, the major chip makers improve their products via smaller feature sizes, more transistors, and complex architectures and design.

This entails fitting and interconnecting more components within the same amount of space, which requires large investments in computer-architecture research and more costly semiconductor-manufacturing processes.

Extreme embedded chips

Extreme chip vendors, Baron said, are often younger companies with less money and fewer employees. They typically design chips that perform only a few types of functions. They thus execute a small range of instructions on applications with a large amount of data that isn't interdependent.

For example, extreme chips in a device that handles multiple wireless LAN connections would process several similar data streams that could be handled independently.

In this case, improving performance entails increasing the amount of data—not the number of instructions—processed per unit of time.

The lack of data interdependency makes it easier to improve performance via parallelization, Baron explained. Increasing chip frequency could also increase performance, but it would come at a higher price and at the cost of greater power consumption and heat, all unacceptable for the devices that use extreme chips.

Therefore, extreme-chip vendors generally develop massively parallel processors that include multiple reduced-instruction-set computer (RISC) cores, called *processor engines*. Because they improve performance via parallelization, these chips can typically operate at lower frequencies than traditional CPUs. In essence, the key to extreme design is to miniaturize multiple simple processors and place them on a chip.

Peter Claydon, chief architect of picoChips Designs, an extreme-processor design consultancy, said, "Our PC102 contains 344 processors on a single chip that runs at [only] 160 MHz. These processors communicate with each other over an onboard, internal bus. And multiple tasks can be run at once. For example, a single PC102 [used in a cellular base station] can process the telecommunications data for hundreds of cellular users in parallel."

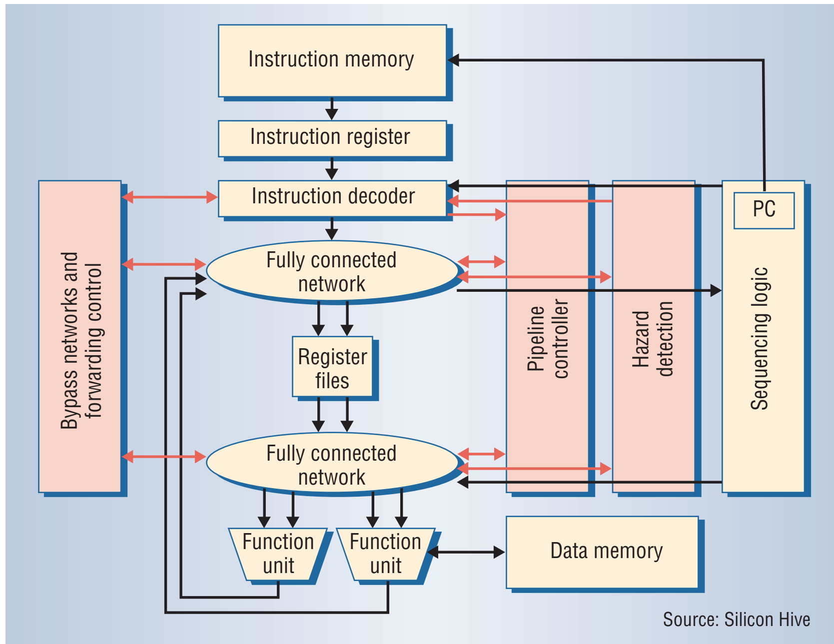


Figure 1. Silicon Hive's ultra-long-instruction-word extreme chip eliminates traditional processor elements such as bypass networking and forwarding control, pipeline control, and hazard detection. The compiler handles these jobs instead. This reduces control overhead, moves complexity to the compiler, and adds flexibility otherwise limited by a hardware-fixed instruction set.

Creative approaches

Several techniques have enabled the design of useful extreme chips.

Better design processes. Chip design was more difficult in the past, Halfhill said, largely because designers had to manually map out chip designs. Today, there are two main easy-to-use hardware-design languages that make the chip-design process easier and faster by letting developers design a chip as a software model: the Very High-Speed Integrated Circuits Hardware Description Language and Verilog. Once compiled, developers can manually tweak chip designs generated via these languages to make them more efficient.

Configurable processors. In the past, vendors put functionality in hardware, and when new functionality was needed, they had to redesign the hardware. "This was an expensive and time-consuming process," Halfhill noted. "Now, designers can make the processor configurable so it will be much easier to adapt to changing standards such as the constantly evolving 802.11 Wi-Fi security standards."

However, configurability can come at a price.

Jeroen Leijten, chief architect of the Philips subsidiary Silicon Hive, said that for its configurable, ultra-long-instruction-word Avispa+ processor to run properly, "all data pipeline staging must be explicitly scheduled in the compiler."

Thus, the vendor must program not only the chip's general operations but also how the chip will deal with a specific program. This yields a fast, highly customizable processor, as Figure 1 shows, but generates more work and expense for the vendor.

Creative parallel processing. MathStar uses a creative parallel-processing approach in its customized chip designs for communications and digital-signal processing.

According to the company's chief architect, Dirk Helgemo, instruction sets are designed into the chip, rather than streamed onto it as is usually the case. This lets MathStar chips use multiple external I/O to handle large amounts of data and thus enables massively parallel processing without massive internal data buses.

MathStar's Field Programmable Object Array uses data paths that are loosely coupled between multiple processors to form a multiple-instruction-multiple-data machine on a chip, enabling it to handle data-intensive applications.

Challenges to doing more with less

Extreme processors face several technical and marketplace challenges. For example, extreme chips may not appeal to some potential users who don't understand their unconventional architectures or don't want to deal with new vendors.

Development tools. Most extreme processors require special development tools. Baron explained, "You simply can't use a QNX, Wind River, or Gnu C because their compilers don't [work with] the new architectures." Many companies thus modify Gnu C because it's open source and already well understood, he noted.

Design issues. Extreme designs with massively parallel processors still lack efficient compiling and efficient designs for communication between the on-board chips, according to Baron. Designers now must both create the chip and try to work out the best way for the processor engines to network with each other, which entails additional time and cost.

Marketplace alternatives. Users who want to gain some of extreme chips' benefits can turn to alternative types of hardware in some cases.

"High-performance RISC processors and DSPs [digital signal processors] are easier to program than [extreme] processors," Halfhill said, "but some applications may require multiple RISC chips and DSPs to match the performance of a single unconventional processor." However, using multiple chips like this would cost too much, consume too much power, and take up too much space for many host devices.

There are also application-specific integrated circuits and systems on chip, which have high performance and are easier to program than extreme chips but require a long time to develop and thus may be obsolete by the time they are finished. Field-programmable gate arrays are flexible but expensive.

Industry observers say there are enough different uses and enough demand for these types of chips, as well as for extreme chips, to provide a market for each.

The embedded market, Halfhill noted, has a huge range of applications, as disparate as rocket guidance systems, pagers, and antilock brakes. He said there will have to be multiple architectures to manage these many different types of demands. Thus, he concluded, there won't be the standardization in the extreme-chip market that you see in the PC or server chip market.

Meanwhile, stated Carnegie Mellon University Assistant Professor Seth Copen Goldstein, demand for high-performance, low-energy consumption, and other capabilities that gave birth to extreme chips will continue, ensuring the technology's ongoing popularity.

However, added Baron, extreme designs will continue to face competition from traditional established chips. In the end, he said, the marketplace will decide which designs will flourish and which will fall by the wayside. ■

***Steven J. Vaughan-Nichols** is a freelance technology writer based in Arden, North Carolina. Contact him at sjvn@vna1.com.*

Editor: Lee Garber, *Computer*, 10662 Los Vaqueros Circle, PO Box 3014, Los Alamitos, CA 90720-1314; l.garber@computer.org

CALL FOR PAPERS

The 1st IEEE International Conference on Mobile Ad hoc and Sensor Systems (MASS-2004)

Sponsored by
IEEE TC on Distributed Processing and IEEE TC on Simulation

Recent Research on wireless networks and mobile computing has concentrated on single-hop networks, where network nodes communicating directly to a fixed infrastructure such as cellular or satellite systems. While in Ad hoc and sensor networks multi-hop scenarios such as conference, hospital, battlefield, rescue and monitoring scenarios are covered, where network nodes communicate via other network nodes. This conference covers the area of ad hoc and sensor systems, from physical issues to applications aspects.

Possible topics of interest include, but are not limited to:

- MAC protocols for ad hoc and sensor networks (801.11, Bluetooth, etc)
- Routing protocols (unicast, multicast, broadcast, geocast, etc)
- Power-aware and energy-efficient design
- Directional and smart antennas for ad hoc networks
- QoS, security, and fault-tolerant issues
- Location and mobility management
- Topology construction and coverage maintenance
- Distributed Algorithms (self-configuration, scheduling, synchronization, etc)
- Data gathering, fusion, and dissemination
- Operating system and middleware support
- Complexity analysis of algorithms
- Analytical, mobility, and validation models
- Applications of ad hoc and sensor networks

Paper submission

All submissions will be handled electronically. Authors should prepare a PDF of their full papers. Papers must not exceed 10 single-spaced and two-column pages using at least 11 point size type on 8.5 * 11 inches pages. Electronic submission instructions will be published on the website http://www.ececs.uc.edu/~cdmc/mass together with style files.

Tutorials

Proposals for half and full day tutorials are solicited. The selection criteria include the expertise and experience of the instructors and relevance to the central themes of the conference. Proposals of at most 4 pages, including 1-page biographical sketch, should be submitted to the Tutorial Chair, Stephan Olariu (olariu@cs.odu.edu), by April 30, 2004.

Important dates

Manuscript Submission due: April 15, 2004
Acceptance Notification due: July 20, 2004
Final Manuscript Due: August 30, 2004

GENERAL CHAIR
Dharma P. Agrawal
University of Cincinnati

PROGRAM CO-CHAIRS
Mario Gerla
University of California, Los Angles

Jie Wu
Florida Atlantic University

PROGRAM VICE CHAIRS
Marco Conti (MAC layer and lower layers)
CNR-IIT, Italy

Prasant Mohapatra (Network layer)
University of California, Davis

Ivan Stojmonvic (Algorithms and Topology)
University of Ottawa, Canada

Yu-Chee Tseng (Applications)
National Chiao Tung University, Taiwan

TUTORIAL CHAIR
Stephan Olariu
Old Dominion University

PUBLICITY CHAIRS
Jean-Pierre Hubaux
EPFL, Switzerland

Kia Makki / Chi Zhou
Florida International University

Tom La Porta
Pennsylvania State University

LOCAL ARRANGEMENT CHAIR
Mihaela Cardei
Florida Atlantic University

FINANCE CHAIR
Anup Kumar
University of Louisville

PUBLICATION CHAIR
Xiuzhen Cheng
George Washington University

REGISTRATION CHAIR
Hong-Yi Wu
University of Louisiana

INDUSTRY LIAISON
Bob O'Dea
Motorola

IEEE TC CHAIRS
Chita Das (Distributed Processing)
Anup Kumar (Simulation)

Is Peer-to-Peer Secure Enough for Corporate Use?

George Lawton

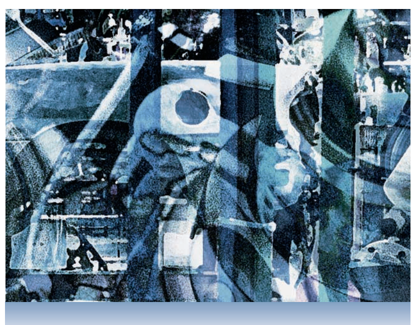
Peer-to-peer technology is being used more frequently for corporate activities. P2P enables work teams, often geographically dispersed, to share information and coordinate projects online, noted Nimisha Asthagiri, development manager for security projects at P2P software vendor Groove Networks.

In a February 2003 study, market research firm Frost & Sullivan estimated that revenue from the sale of corporate P2P systems will increase from \$42.8 million this year to \$4.53 billion by 2007. As Figure 1 shows, Frost & Sullivan also predicted that the number of enterprise P2P clients in the US alone will rise from about 100,000 this year to about 1.8 million in 2007.

Within a company, explained Rice University Assistant Professor Dan Wallach, "P2P technologies might be used for instant messaging, file storage, remote backup, or a number of other applications."

Many large vendors are beginning to integrate P2P applications with the core business programs they sell, such as their customer-relationship management systems. And vendors such as IBM, Intel, Microsoft, and Sun Microsystems are integrating P2P functionality into other products.

Moreover, there is a growing number of stand-alone corporate P2P applications, including Avaki Data Grid, Groove Workspace, and Joltid's Peer Enabler. And if vendors agree on a solid, standards-based, widely supported infrastructure that provides interoperability and platform independence, corporate P2P adoption could increase even more.



However, now that P2P is being used for more important functions, security has become a critical issue. Thus, companies such as BitTorrent, Intel, Joltid, Microsoft, and Sun are creating platforms that simplify the development and deployment of secure P2P applications. Vendors such as Applied Meta, Entropia, Groove, Popular Power, and United Devices have integrated standardized security approaches into their P2P products.

These developments lend credibility to P2P's suitability for corporate use, as well as for sharing music files, said Jarad Carleton, leader of Frost & Sullivan's Internet infrastructure program.

SECURITY THREATS

Clients, not network administrators, control P2P networking, which is described in the sidebar "The Ins and Outs of Peer-to-Peer Networks." Individual users are responsible for securing their files and directories. P2P thus circumvents many types of corporate security, including controls over who is allowed to enter workspaces and which system resources visitors have access to.

Companies that run or are considering running P2P programs are concerned that users could accidentally share important corporate data. Jim Lowrey, a security specialist at Endeavors Technology, said another major security challenge is controlling the incoming P2P traffic that can enter a company's network. Users are concerned that P2P applications could let viruses or hackers enter systems.

"One of the newer concerns is lawsuits occurring from [a company] having copyrighted material on its network and servers, facilitating the trading of this material. Companies are getting [hit] with million-dollar settlements because they have not prohibited the use of these materials," explained Frost & Sullivan security analyst Jason Wright.

Kevin Rowney, chief technology officer of security vendor Vontu, said, "Most of the corporations we deal with would prefer to see P2P use stopped entirely on their networks. They frequently view P2P systems as exposing companies to an excessive amount of risk."

Sharing sensitive information

A critical vulnerability is that most P2P applications can turn a computer into a network file server. This increases the chances of improper sharing of intellectual property, confidential corporate information, and other sensitive documents.

A recent US Congressional study, "File-Sharing Programs and Peer-to-Peer Networks: Privacy and Security Risks," found that many users of consumer P2P applications accidentally made highly personal information—such as tax returns, financial documents, medical records, and attorney-client communications—available to other users. Similar problems could occur with corporate P2P networks.

In addition, P2P software bugs could cause systems to accidentally make sensitive information available. "A P2P client is also a server, in that it will accept connections from other machines on the network," explained Rice University's Wallach. "Like any server, it needs to be robust against buffer overflows and the like. Any vulnerabilities in the P2P client would, like vulnerabilities in a Web server, allow an attacker to break in and take control of the computer."

According to Wallach, the majority of P2P systems don't include encryption and thus could let eavesdroppers with sniffers or similar devices find and then view material sent by unsuspecting users.

Attackers could also steal private information or even activity log files from a client by using a P2P application to scan for an accessible directory on a victim's system or by exploiting system vulnerabilities.

As is the case with other applications, the most difficult type of P2P-based data theft for organizations to prevent involves authorized users misappropriating or being careless with data.

Usability problems

Network administrators can set up P2P applications with varying degrees of centralized control. However, users frequently maintain control of managing their own file directories and the permission process for accessing them.

For P2P applications to be secure, users must know how to configure them properly. For example, they must ensure that they set up only the proper directories for sharing with other P2P users and that all data is identified as to whether it can be shared and if so, with which groups or individuals.

The failure to perform these tasks either correctly or in compliance with company policies can create serious security problems.

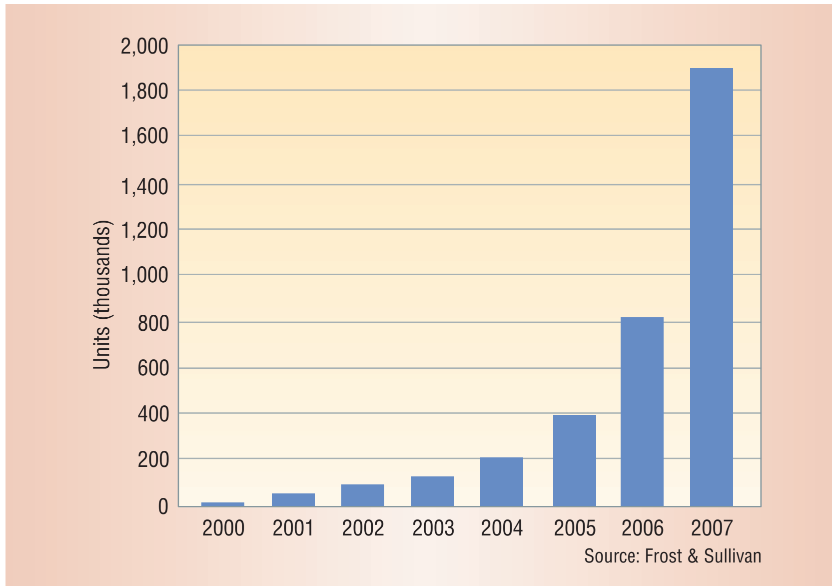


Figure 1. Frost & Sullivan, a market research firm, predicts that the number of corporate peer-to-peer clients in the US will increase steadily until 2005, when it will start rising more quickly.

The Ins and Outs of Peer-to-Peer Networks

Peer-to-peer software lets users access files on other computers by creating an interface to files that other users have made available via the P2P network.

In some cases, systems implement P2P communications by giving each node both server and client capabilities. Users start by downloading and executing a P2P networking program. After launching the program, users can enter the IP addresses of other computers on the P2P network with whom they want to communicate.

Users then decide which of their files they want to share and which they want to password-protect. When they want to communicate with another network member, their computer establishes a connection online with the member via the appropriate IP address.

Most consumer and many corporate P2P applications communicate as HTTP traffic on TCP/IP port 80, making communications from outside an organization difficult to stop with standard firewalls, which generally leave the port open to give users access to the Web.

Sender-anonymizer and encryption technologies make it even harder for firewalls and other technologies to analyze P2P messages and determine whether to stop them.

Spreading malicious code

P2P users could accidentally download viruses, worms, or other malicious software (as well as system-clogging, activity-tracking spyware) as part of files they access. They could also quickly spread malware they accidentally acquire from e-mail or other sources.

Consumer P2P systems have already seen proof-of-concept worm attacks such as VBS_GNUTELWORM and W32/Gnuman. There is also concern that P2P could spread malware in part because of software-design flaws.

Sharon Ruckman, senior director at antivirus vendor Symantec's Security Response center, said its biannual Internet Threat Report found that of the most prevalent 50 viruses and worms, the number that could attack P2P systems increased from four in 2002 to 19 this year.

External attacks

There is considerable concern about P2P systems opening a network to external attacks. As is the case with Web servers or e-mail servers, P2P software can have bugs that hackers can exploit by creating problems such as buffer overflows, noted Rice University's Wallach.

According to Frost & Sullivan's Wright, "If you are running a P2P app at home and using a [virtual private network] into the [company's] network, you are essentially providing a pathway for anyone who wants to scan you as a host. There are ways to manipulate that situation to gain access to a corporate network through a VPN tunnel."

PROPOSED SOLUTIONS

Vendors are looking at various ways to provide security for P2P systems. For example, in its P2P products, Endeavors uses WebDAV (Web distributed authoring and versioning), an Internet Engineering Task Force standard that provides HTTP extensions to facilitate collaborative editing and file management between users working together over the Internet.

A new security model would periodically redirect P2P data from the client, where the user keeps it, to a central server for auditing, Groove's Asthagiri said. This auditing could identify problems that have already occurred and security trends that need to be watched.

In P2P communications, as in many other types of transactions, users must limit the access that other parties have to system resources. This requires authorization, in which a system defines a user's or group's ability to access specific directories or sets of data.

Authentication

Because basic P2P applications lack their own security, including identity authentication for those involved in P2P communications, users can't assume that other parties are who they say they are, explained Kevin Beaver, chief technology officer of security vendor Principle Logic.

Vendors are looking at various ways to provide security for P2P systems.

Digital signatures can help authenticate P2P communications. The signatures are digital codes that are attached to an electronically transmitted message and that uniquely identify the sender.

PKI. Peers can identify each other by directly exchanging credentials, such as digital certificates.

They can also use public-key infrastructure (PKI) technology, in which a trusted third-party authority issues digital certificates. A user who wants to send an encrypted message applies for a digital certificate from a certificate authority. The authority issues an encrypted digital certificate containing the applicant's public encryption key and other identifying information.

The message recipient uses the authority's own, readily available public key to decode the certificate attached to the sender's message. The recipient verifies that the authority actually issued the certificate and then gets the sender's public key and identifying information from the certificate. With this information, the recipient can send an encrypted reply that only the original sender can decrypt with a private key, which corresponds mathematically to the public key.

Although effective, PKI can be complex, expensive, and time-consuming to implement.

Directory services. Companies could also use their existing enterprise directory services, which list the resources available within a network, to identify users, find their public keys, and otherwise provide some P2P authentication. Companies such as Endeavors and Groove plan to take this approach with their products.

Users could accomplish directory-based authentication by various means, including the lightweight directory access protocol, which is a set of protocols for accessing information directories.

A directory usually covers a single administrative domain, generally representing a single organization or department, and thus would help authenticate P2P users from that domain but not from other domains.

Encryption

Encryption can play several roles in P2P applications. For example, it can protect a particular transmission between peers on an unsecured network.

Administrators can also use encryption to create a VPN for communications over unsecured networks. In addition, by using keys unique to specific users, encryption can support the authentication process.

A few smaller companies, such as Endeavors, have developed corporate P2P systems that use the encryption-based secure sockets layer (SSL) technology, which provides security for Internet messaging. Other firms are working with Pretty Good Privacy encryption, used extensively with e-mail. Other options include cryptographic technologies such as the RSA public-key encryption system with 2,048-bit keys and the 192-bit-key version of the Advanced Encryption Standard algorithm.

Intel's Peer-to-Peer Trusted Library

Intel has a free, open source developers' library that companies can download to add security to P2P systems they develop. The Peer-to-Peer Trusted Library (http://sourceforge.net/projects/ptptl) supports digital certificates, peer authentication, secure storage, encryption, and digital signatures. The library is built on the open source OpenSSL toolkit (www.openssl.org/).

Application-layer firewalls

Traditional firewalls evaluate the header information on packets to determine whether to admit them to a network. Frost & Sullivan's Wright said a new generation of application-layer firewalls, from vendors such as Check Point Software Technologies and Sanctum, use deep packet inspection to examine a message's contents.

This could help the firewalls better detect and block sensitive P2P content being improperly sent outside a corporate network.

Unlike today's faster technologies, slower firewalls couldn't handle the overhead of such extensive inspections.

Said Vontu's Rowney, "The enormous power of [P2P] systems to enhance the creative exchange of information, ideas, and content can't easily be brushed aside by security concerns. Over time, companies will find productive ways to conduct business using these technologies."

"If vendors of P2P platforms commit themselves to addressing the reasonable security concerns of the enterprise, the future for this technology could be very bright," he added. "On the other hand, if vendors consistently fail to standardize around a platform that addresses these concerns, network administrators will work to stop these deployments." ■

***George Lawton** is a freelance technology writer based in Brisbane, California. Contact him at glawton@glawton.com.*

Editor: Lee Garber, *Computer*, 10662 Los Vaqueros Circle, PO Box 3014, Los Alamitos, CA 90720-1314; l.garber@computer.org

E-Mail Security Gets Personal

A US company has developed and released a simplified, less expensive e-mail security system that derives its public encryption key from a message sender's own e-mail address.

Voltage Security's SecureMail software suite, which works with desktops or laptops, transparently executes the cryptography, making it as simple as sending unencrypted e-mail, said company spokesperson Wasim Ahmad.

SecureMail's server-based software uses one algorithm to convert an e-mail address into a large number that functions as a public encryption key and another algorithm to create the user's private key based on the public one. In addition to providing public-key techniques, Ahmad said, Voltage's technology also offers strong elliptic-curve cryptography.

Voltage says its approach automates and simplifies the authentication process, reversing the trend of increasing complexity in technology that establishes trust between two communicating parties. This could attract users who want more secure e-mail but are concerned about the amount of work necessary to implement public-key infrastructure technology, the traditional way to encrypt e-mail.

"You can't roll out traditional PKI to customers," said Ahmad, because the technology is expensive and difficult to use.

By automatically using an e-mail address to create keys that enable trusted, secure communications, Voltage's technology eliminates PKI's requirement that communicating parties always enroll with and receive certificates from a third party attesting to their identities.

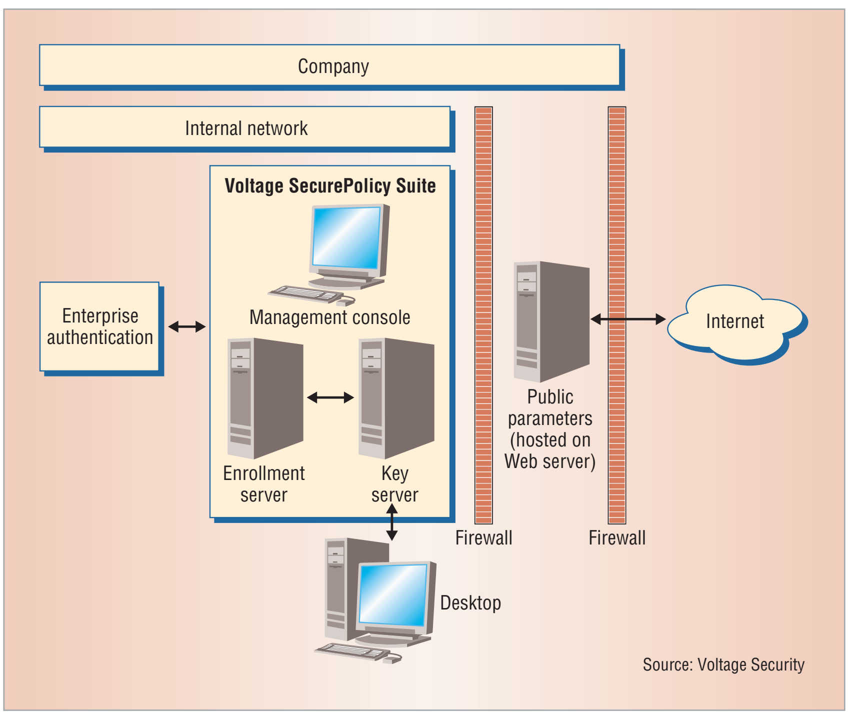
With PKI, senders who have received certificates must find a recipient's public key in a directory and use that key to scramble a message. Recipients then must unscramble it using a private key that is mathematically related to the public one. Thus, PKI entails a complex, costly process that requires overhead such as certificate management.

With Voltage, only the sender needs the software in advance. Recipients running various e-mail applications can click on a link to download free Voltage software that automatically decodes the message and lets them send an encrypted response. They can also use a no-download reader that lets them read the mail in a browser but not send an encrypted response.

Because of the direct communication between senders and recipients, the Voltage system further reduces complexity and potential management problems by not requiring a dedicated server to handle the secure transmissions.

To succeed, Voltage must determine how to get enough users to work with the technology for it to be effective, explained Phebe Waterfield, analyst for the Yankee Group, a market research firm. And when widely used, she added, the approach must prove to be as transparent as Voltage claims. The system has yet to be rigorously and independently tested, she noted. ■

—Linda Dailey Paulson



Voltage Security has released an e-mail security system that automatically authenticates users and derives its cryptographic keys for encoding messages from a user's own e-mail address. This is simpler than public-key-infrastructure technology, a traditional but complex way to secure e-mail. Voltage users who work with the system for the first time authenticate themselves to the enrollment server. Once authenticated, they receive keys from the key server. The public parameters contain all the information an organization needs to create keys for any user.

Ceiling for Controversial US H-1B Visas Drops

The US has reduced the number of people admissible under its controversial H-1B visa program, which lets skilled foreign workers—many in the computer technology field—take jobs in the country for up to six years.

The US Congress refused to change a policy, called for by the American Competitiveness in the 21st Century Act (AC21), that decreased the annual visa ceiling beginning this fiscal year to 65,000. The cap the past three years was 195,000.

Paul L. Zulkie, president-elect of the American Immigration Lawyers' Association (AILA), which supports higher H-1B caps, predicts the US will award all 65,000 visas by March 2004. He said this would leave many companies without the qualified employees they need until the next fiscal year starts on 1 October.

The H-1B visa program has sparked ongoing controversy. Many technology companies say they hire US workers when possible but need higher visa caps to fill important positions requiring skills for which qualified domestic workers aren't readily available.

H-1B opponents, on the other hand, have assailed the program for taking away scarce technology jobs from US workers and giving them to workers from other countries who, they say, often accept lower salaries.

Zulkie said AC21's temporarily higher visa caps failed to provide a long-term solution to US employers' need to hire foreign professionals. However, president John Steadman of the IEEE-USA, which opposes high H-1B, said the unemployment rate for US IT professionals is high enough that many domestic engineers are available to hire.

Harris Miller, president of the Information Technology Association of America, an H-1B supporter, said the US Congress didn't want to consider approving a higher ceiling at a time when domestic unemployment is a politically sensitive issue.

The AILA estimates 900,000 H-1B employees, 35 to 45 percent from India, currently work in the US.

Steadman said US technology firms are currently finding some of the overseas employees they want by using L-1 visas, which permit intracompany transfers of foreign managers and workers with specialized skills.

Some businesses say that they plan to cope with lower H-1B visa caps by outsourcing work overseas. ■

—Linda Dailey Paulson

Technique Speeds Data Transmission in Computers

Sun Microsystems researchers have discovered a way to transmit data inside a computer 60 to 100 times faster than with traditional techniques by letting information pass directly between chips placed next to one another. The approach would also reduce energy consumption.

The Sun researchers say they sent data at 21.6 Gbits per second between chips using a pared-down version of their new technology. Principal investigator Robert J. Drost said this prototype was a proof of concept that used chips with 0.35-micron feature sizes rather than chips using today's smaller feature sizes. Speeds could be even greater using current chip technology, he noted.

Traditionally, data passes between elements in a computer via wiring. With Sun's approach, chips sit next to each other and transmit data between connection points, which include tiny transmitters and receivers, via *capacitive coupling*. When chips are brought into alignment, explained Drost, the capacitance between them carries the signal across the short open space between the connection points with very little of the latency that occurs when signals have to pass through wires.

A system's total throughput would increase with more interchip connections. Thus, Drost said, when Sun's technique is used fully in computers, data rates could exceed a trillion bits per second.

Currently, the 25-micron wiring that takes data into and out of chips is soldered to processor edges at ball-bond pads with pitches 250 microns wide. With Sun's approach, connection points would be only 50 microns wide. Thus, there is room for more data connections, which leads to higher transfer rates.

Meanwhile, because the system only has to push signals across a small opening between connection points, it uses less power.

Sun's approach would also improve performance by letting computer designers pack chips into a system far more densely than is possible today.

However, because the chips would be in close proximity, there is concern that the approach could generate excessive heat and that the connection points could create interference. Some also question whether the technique can be implemented effectively on an assembly line.

"It's an extremely difficult problem. [Sun's approach] is unique and holds a great deal of promise," said Christopher Willard, research vice president with market research firm IDC.

Although Sun hasn't fully tested its technique, Drost said, the US Defense Advanced Research Projects Agency has expressed interest in it. Meanwhile, the company says it wants to find commercial implementations as soon as possible. ■

—Linda Dailey Paulson

Viewing the World through Interactive Panoramic Images

Photographs limit viewers to seeing the image captured in the viewfinder, no matter how interesting the surrounding areas might be. However, interactive panoramic images are beginning to solve this problem and create a new art form.

The increasingly popular panoramic images let viewers move their cursor across a photograph and get a 360-degree view of a scene beyond the small image visible in a browser window.

Panoramas used to be made primarily by hobbyists who took a series of photos across a scene and then used a computer to assemble them horizontally. They displayed the finished product in a browser window in which viewers could move from side to side to see the entire view.

Panoramic-image creation now appears to be expanding for several reasons, said Web designer and photographer Erik Goetze.

For one thing, high-resolution digital cameras have minimized the work needed to build panoramas from individual photos by producing images that don't have to be digitized first. Also, Goetze said, getting images on a Web server is easier and quicker than in the past. It's now possible to produce a working panorama perhaps 5 minutes after shooting it, he explained, although more work improves the final product.

Creating a panorama also requires stitching multiple images together and editing the image to make sure the stitching was done precisely. Stitching applications—such as Apple's QuickTime VR Authoring Studio, PanoTools, and RealViz Stitcher—analyze image pairs and work with common features to match them properly.

Some applications let designers create spherical panoramas in which a viewer can look up and down as well as side to side. Meanwhile, Goetze noted, fast Internet connections improve the viewing process by quickly delivering detailed, full-screen images.

Other photographers are experimenting with ways to expand the viewer's experience by providing video-like transitions to link separate but related sets of panoramic images.

Also, companies such as Zoomify are developing software that can zoom in on panoramic images and stream additional detail into a scene as the viewer moves "closer."

A few businesses are exploring augmented panoramas, which are designed to make a scene more interesting by adding animated characters or streaming Webcasts of people. However, this approach is not common because it is time-consuming.

Fred Ritchin, director of the online magazine Pixel Press and an associate professor of photography and communications at New York University's Tisch School of the Arts, said designers haven't begun to explore panoramic images' full potential.

For example, he said, they could be used with text and other types of content to create a more interesting Web experience. ■

Web Sites that Offer Panoramic Images

The following Web sites offer panoramic images, some of which require plug-ins to view:

- The Picture Project's 360degrees site is a social-documentary project on the US criminal justice system: www.360degrees.org.
- Australian commercial photographer Peter Murphy started a Weblog showcasing panoramas: www.mediavr.com/blog. He also has panoramic images at www.mediavr.com/bronte1.htm and http://culture.com.au/virtual/.
- 360 VR Studio, with New Jersey commercial panoramic photographer Jook Leung, offers panoramic images shot in the New York City area: www.360vr.com.
- Danish commercial photographer Hans Nyberg maintains a Web site with links to other sites containing 30,000 panoramas: www.panoramas.dk.
- Belgian photojournalist Tito Dupret has panoramas of places such as Beijing's Forbidden City and Cambodia's Angkor Wat on the World Heritage Tour Web site: www.whtour.net.
- An augmented panoramic image can be found at www.throbbing.com/bikerace.

Linda Dailey Paulson is a freelance technology writer based in Ventura, California. Contact her at ldpaulson@yahoo.com.

Editor: Lee Garber, *Computer*, 10662 Los Vaqueros Circle, PO Box 3014, Los Alamitos, CA 90720-1314; l.garber@computer.org

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 The IEEE Computer Society's Web site, at **www.computer.org**, offers information and samples from the society's publications and conferences, as well as a broad range of information about technical committees, standards, student activities, and more.

OMBUDSMAN Members experiencing problems—magazine delivery, membership status, or unresolved complaints—may write to the ombudsman at the Publications Office or send an e-mail 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 the Publications Office:

- 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)

To check membership status or report a change of address, call the IEEE toll-free number, +1 800 678 4333. Direct all other Computer Society-related questions to the Publications Office.

PUBLICATIONS AND ACTIVITIES

Computer. An authoritative, easy-to-read magazine containing tutorial and in-depth articles on topics across the computer field, plus news, conferences, calendar, industry trends, and product reviews.

Periodicals. The society publishes 12 magazines and 10 research transactions. Refer to membership application or request information as noted at left.

Conference Proceedings, Tutorial Texts, Standards Documents. The Computer Society Press publishes more than 160 titles every year.

Standards Working Groups. More than 200 groups produce IEEE standards used throughout the industrial world.

Technical Committees. Thirty TCs publish newsletters, provide interaction with peers in specialty areas, and directly influence standards, conferences, and education.

Conferences/Education. The society holds about 100 conferences each year and sponsors many educational activities, including computing science accreditation.

COMPUTER SOCIETY OFFICES

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

Publications Office
10662 Los Vaqueros Cir., PO Box 3014
Los Alamitos, CA 90720-1314
Phone:+1 714 821 8380
E-mail: help@computer.org
Membership and Publication Orders:
Phone: +1 800 272 6657 Fax: +1 714 821 4641
E-mail: 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
E-mail: tokyo.ofc@computer.org

IEEE

A Tale of Three Disciplines ... and a Revolution

Jesse H. Poore
University of Tennessee

Some disciplines, like circuit and genetic engineering, seem to evolve from theory to practice relatively responsibly. Software engineering, on the other hand, while not yet at the guillotine, has suffered a decided lack of direction. It may be time to storm the gates.

Despite years of hard work and dedicated champions, software engineering still struggles to become the discipline its founders envisioned. Its educational and industrial constituents have not only failed to converge, but the already large gap between software education and industry practice continues to grow. Curricula are diffuse, software systems continue to have ever larger error counts, and the financial risks of taking on a software project are growing.

Not that efforts haven't been valiant. Vic Basili, Fred Brooks, Barry Boehm, Harlan Mills, and David Parnas, in particular, have worked hard to align software engineering foundations with curricula and curricula with industry practice. As accomplished scientists and engineers, they have articulated fundamentals, written textbooks, taught in universities, led significant programs in industry, and modeled good practices from positions of prestige. Their efforts produced one success after another as software engineering emerged into an academic discipline and profession.

But I believe even they would agree that academic software engineering has had too little impact on the software industry. In part, the evolutionary environment was to blame. Like Dickensian protagonists, software engineering suffered through the best and worst of times. The best was when its advocates embraced economic opportunity; the worst was when they struggled with complexity. Wisdom and foolishness alternated as we pitted formal abstractions against individual ingenuity. As the bonds of theory, practice, and tools fused and broke apart by turns, we progressed in the light or wandered aimlessly in the dark. And like the heroes of the French Revolution, we look to a future that will bring us everything or nothing, depending on the public trust.

But other disciplines evolved in similar environments and have either become successful or have established the framework to become successful.

THE CHARACTERS

Circuit engineering is a discipline that has done the job right, and genetic engineering is at least *trying* to do the job right. Both have managed vast complexity and achieved a high level of public trust. Software engineering could take lessons from either discipline—and this is not simply an academician's lament; the cost of software is enormous and the risk to public safety daunting.

Published by the IEEE Computer Society

Circuit engineering

Electronic circuit engineering began with Claude Shannon's 1938 thesis. From this point, engineers recognized that a circuit was a rule for a function—a Boolean function. Pioneers of the field were strongly motivated to proceed in a thoughtful, methodical way. Electronic components and hardware construction were expensive. Repair—destruction and reconstruction—cost both money and time.

Shaped by talented mathematicians and engineers, the field fully exploited the Boolean algebra associated with circuits. In fact, the entire process of circuit design was a mathematical activity, and the hardware mirrored the algebraic operations AND, OR, and NOT. Theorems made it possible to build any circuit from just NAND gates, which proved economical. Mathematicians developed normal forms for functions using specific operators.

Innovators recognized the power of visual graphics early on and devised a notation that the community adopted and sustained. From the beginning, the correspondence between circuit diagrams and functional equations was exact. Only rank amateurs in their home radio shacks would construct devices without first sketching diagrams, scribbling functions, and consulting parts catalogs. The idea of building every new function from scratch—from AND, OR, and NOT gates—was ludicrous.

Engineers defined larger units (registers, shift registers, adders, multiplexers, demultiplexers, and so on) mathematically, constructed them physically, and reused them routinely. Reuse was always a smart idea. In fact, developers precisely prescribed and cataloged each component's functional input-output relationship for just that purpose.

This process streamlined the innovation of better, cheaper, faster components, since with the functional precision of each catalog description, manufacturers could hide internal secrets from the user. Moreover, the assembly of a new circuit from components was always clear because a function represented each component, and all the functions were of the same family. The engineer would simply formulate the new function at the highest level of abstraction.

Innovations from physics and engineering paced progress, spanning devices from vacuum tubes to transistors to integrated circuits. At the same time, the mathematical theory flourished. In the 1960s, as progress accelerated, the theory of finite state machines as sequence recognizers paved the way for scaling up complexity. By the 1980s, circuit complexity (still rules for functions) had vastly increased.

Enormous state space characterized this complexity, yet engineers treated it explicitly. The number of active components also characterized complexity, which innovators managed by continually adhering to a description of components assembled according to the mathematics of functional composition.

Computer-aided design, engineering, and manufacturing tools were at the industry's heart, making abstraction practical and enforcing an affinity of theory, practice, and tools. Testing was an integral part of the engineering, and design for testability was a fundamental consideration. Design tools and standard best practice were always available and automated in step with hardware evolution.

From the beginning, the correspondence between circuit diagrams and functional equations was exact.

Reverse engineering was also a consideration, with its focus on determining a circuit's function by observing its input-output behavior or even divining its internal state design. However, circuit engineering was and still is mostly a matter of forward design using established components and leaving a documented design trail.

As the field progressed, its proponents kept the research frontier linked to undergraduate training, textbook notation, and industry practice. Tools were generally in step with demand. The vocabulary retained the mathematics, physics, and engineering concepts that engineers knew and used. There was a consensus as to what constituted good professional practice, and practitioners generally managed projects well because everyone understood a worker's qualifications in terms of training and experience.

Today, the complexity of circuit engineering matches or exceeds that of any other field.

Genetic engineering

Although its evolutionary pace is swifter than circuit engineering's, genetic engineering is following a similar path. The discovery of DNA's double-helix structure, attributed to James Watson and Francis Crick in 1953, revolutionized life sciences in general. Although the details vary among organisms, we now know that genomes carry a complete set of instructions for an individual's lifetime.

Humans have trillions of cells, each of which has 23 pairs of chromosomes. Each chromosome has a DNA strand, a regular double-stranded helix linked by bonds between guanine and cytosine on the one hand and thymine and adenine on the other. These linkages are base pairs. The gene itself is a segment of a DNA molecule and can have a few hundred to several million base pairs. The

Although genetic engineering is mostly a matter of reverse engineering, it holds many parallels with circuit engineering.

human genome, for example, comprises some three billion base pairs. Genetic instructions are encoded in these pairs, and like the circuit, provide the rule for a function.

In genetic engineering, the circuit is already defined, the program already written; the *lifeware* function (or relation) is already implemented in the gene. It is true that some genetic engineering occurred before the DNA discovery, but the process was trial and error, guided by experience, intuition, and experimental principles. The mathematical model for DNA waited just under the surface to be recognized, and once it was, scientists and engineers embraced it.

At this stage, genetic engineering is mostly a matter of reverse engineering. Still, it holds many parallels with circuit engineering. Genetic engineers want to determine what a gene does, where it is located, or which gene causes an observed phenomenon (disease or talent). They want to know how to modify genetic circuits so that the organism will respond differently. As with electronic circuits, these modifications can be physically small but produce dramatic behavioral or physical changes in the organism.

Genetic engineers are also interested in learning how to make new genes from other genes—in essence, how to reuse gene segments as components. Lifeware differs from hardware in that it self-replicates, can self-repair, uses novel information-processing modes, and can develop diversity from undifferentiated state.

The life sciences have certainly benefited from mathematical thinking in the past, as in the contributions of R.A. Fisher. However, most life scientists were not trained in the mathematical specialties that current field research and practice require. Modern genetic engineers must confront numbers arising from combinatorial complexity on a par with the state space that circuit designers routinely address. For example, consider all possible combinations of base pairs in double strands of length 120,000. Then, consider all possible cuts of length, 1, 2, 3, ..., n, depending on the application and all possible reconnections of cuts of compatible length that might lead to a crossover having certain properties. Imagine the complexity if genetic engineers achieve their goals and determine all the genes in the DNA sequences of multiple organisms and develop intellectual and laboratory tools to use this information with precision.

The reaction by scientists in genetic engineering to this situation is exemplary. First, they observed the size of the problem and concluded that their only hope was to follow scientific fundamentals and best engineering practice, which requires engineers to catalog and reuse all results and conduct peer reviews of all working methods and work products. In other words, they recognized both a sequence-processing problem and a substantial information systems problem.

Next, the scientific-industrial leaders of the field turned to mathematics to characterize problems and to computer science for the appropriate technology. Obviously, considering every sequence was not possible, so scientists adopted a mathematical approach to successively partition the space and work productively. The mathematical work is not complete, but scientists continue to study the algebras and operators. An important remaining task is to build sequence recognizers that can divine inner structure strictly from input-output behavior.

Complex as these issues are, they are somewhat similar to problems in electrical engineering and automata theory. In the final analysis, a genetic-engineering effect is a (possibly nondeterministic) transformation, and a gene is a rule for a function (or relation). Moreover, because most genetic-engineering transformations are embarrassingly parallel, engineers can use algorithms that exploit parallelism. And finally, researchers who understand both mathematics and genetics are writing the code. Verification and test will proceed carefully because errors could lead to false research conclusions, harm people and the environment, and culminate in a huge financial loss. For the same reasons, the community is highly likely to share accepted codes and catalog established results for subsequent use.

The genetic-engineering industry and academia are committed to the idea that students will be trained in the appropriate life science, mathematics, and computer science fundamentals. Curricula changes update the mathematics and computer science courses. The industry and research community evolves tools that enforce genetic engineering's foundations and consensus on processes. The vocabulary retains reference to the mathematics and cell science on which genetic engineering is built. Industry and the research community intend to go forward with qualified workers trained in the fundamentals (Elias Zerhouni, "The NIH Roadmap," *Science*, vol. 302, 2003, p. 63).

Software engineering

Many of us regard the 1968 NATO conference as the birth of software engineering, which was then regarded as the design of computer programs and software-intensive systems within a performance, quality, and economical framework. (Ironically, the source material for Frederick P. Brooks' *The Mythical Man Month* was shipping in quantity at that time, so perhaps the conference assumed that it is never too late for a revolution.)

Harlan Mills was the keynote speaker at the 1975 ACM national conference, where he presented "The New Math of Computer Programming" (*Comm. ACM*, Jan. 1975, pp. 43-48)—one of his many efforts to remind us of the mathematical basis for software development. Like a circuit, a program is a rule for a function and subject to an algebra of program construction, which Mills defined. All the benefits that accrued to the circuit design industry could have accrued to the software industry had researchers and practitioners simply acted similarly on this basic fact.

Yet, despite our knowledge, the software industry progressed by dint of nonconformist intellect and massive amounts of money rather than by disciplined science and engineering.

A FAR BETTER THING

In each of the three disciplines, the steps from conception to final product are similar. For specifications in circuit engineering, we have functions over fixed-length sequences of a two-symbol alphabet with Boolean axioms. For genetic engineering, we have functions over arbitrary-length sequences of a four-symbol alphabet with DNA axioms. For software engineering, we have functions over arbitrary-length sequences of an arbitrary alphabet with axioms that must be derived from the application itself.

In all three cases, specifications give rise to state machines. Prescriptions for implementation in various systems, or the code, map the state machine to the target implementation. Code can be correct by design in all cases. We can design circuits with testability built in. We can design genes with a known way to test them. And we can design software for testability—but we generally don't. When we can't fully automate exhaustive testing, we can use statistical models to assure that the circuit/gene/software meets quality standards by statistical protocol.

Like circuit and genetic engineers, software engineers should look first to the underlying science for vocabulary, methods, and tools. Half a century later and after a correct start, the software industry remains disconnected from university training, and university training is disconnected from software's mathematical foundations.

We must work harder at becoming disciplined: Whereas physics and hardware realities hold circuit engineering to fundamentals, and chemistry and life imperatives restrict genetic engineering, software engineering must freely elect and self-impose its intellectual principles. Nonetheless, the complexities are more or less equal.

We can design software for testability—but we generally don't.

Complexity in the software industry grew more slowly than in genetic engineering, yet today, the genetic engineering industry is ahead of software engineering in principles, methods, training, notation, vocabulary, tools, and productivity! Why? Because both genetic engineering and circuit design follow good science and engineering practices.

Software engineering had a later start than either circuit or genetic engineering, so it had the chance to model itself after either one. We knew from John von Neumann that software development is an inherently mathematical activity—just as much so as circuit or genetic engineering—and we understood many of the mathematical facts from the outset. We grasped that we ask the same questions as circuit or genetic engineers: What must the new program do? What does this program do? Where is the program that does such and such? How can we modify the program to behave differently or construct new programs more efficiently?

Ironically, software engineering uses little computer power relative to circuit engineering or genetic engineering to support design and testing. All three disciplines are (or could be) based on very similar symbol manipulation, component repositories, design and validation, state space management, and certification testing.

Circuit and gene designers are generally willing to pay the computational price for design and testing, but software engineering will cop a plea rather than turn to supercomputers when the state space grows large or (statistically) exhaustive testing is in order. This may sound harsh, but we have much at stake.

I'm not denying the software accomplishments, the existence of virtuoso programmers and effective development teams, or the use of tools such as editors, compilers, and leak detectors. (Indeed, many have debunked the "productivity paradox.") But neither am I willing to ignore the chaos, failures, and risks to public safety nor the enormous sums wasted.

We can manage only so much complexity ad hoc, and we can absorb only so much financial waste and tolerate only so much risk to public safety. We have compelling reasons to try to get it right, espe-

> **Erroneous and arbitrary solutions are too often accepted as the designer's creative prerogative.**

cially where correctness, performance, quality, and economy really matter.

The software engineering industry would be better off today if it had followed the pattern of circuit engineering, and it will be better off tomorrow if it follows the path of genetic engineering. I find it bizarre that, given software's origin and its very nature, we could have such a gross disconnection among the field's foundations, university curricula, and general practice. We can only partially blame the lack of physical or chemical constraints, legions of extremely bright independent thinkers, and unlimited sums pushed into software development and maintenance. Also, the industry has become complacent, enjoying an inexplicable degree of blind public trust and acceptance of low quality. It has had little motivation to connect practice with preparation.

MADAME DEFARGE'S SCARF

Every revolution starts with a list of culprits that must be beheaded or at least retrained. In software engineering, the culprit is our academic programs. Across the university, from accounting to zoology, we are teaching students to be proud of writing small programs badly (comparatively few of the population design little circuits and little genes) and to be grateful for shabby application software.

I recently examined several of the software packages included with nutrition textbooks as well as software that health clinics actually use. I found that although the textbooks had a rather uniform look and feel, the software packages were very different, ranging from intuitive and easy to use to inscrutable. The programs were easy to hang and most lost state awareness during use. Moreover, when I entered the same data using each package, I got materially different results. We seem to have no quality watchdog for this kind of software, since book production attends to the book's quality but ignores the software that comes with it.

In our primary computer science, computer engineering, information science, information engineering, and even software engineering curricula, we fail to introduce and reinforce with urgency the mathematical foundations throughout the degree program: disciplined system and program design, the seriousness of design errors, and designs that satisfy a social responsibility, not just the author's ego. Erroneous and arbitrary solutions are too often accepted as the designer's creative prerogative.

A look at history from the 1965 ACM curriculum to current offerings makes it clear that academia's early efforts to lead gave way to frantic efforts to appear relevant to an industry unencumbered by the baccalaureate experience.

This failure has many ugly faces, and the list of have-nots is long: We have no

- standard notation with mathematical fidelity that everyone will use,
- normal forms and cataloged components,
- experimentally validated best practices
- productive programming languages in widespread use,
- tools evolved systematically over decades that facilitate abstraction and enforce software's foundations in industrial practice,
- vocabulary with a mathematical and engineering basis (instead we have beans),
- curricula in close rapport with the workforce,
- standards for product usability and correctness,
- workforce trained in the fundamentals and experienced in standard practice based on fundamentals, and
- industry that will employ only qualified workers and pay only for quality work.

Software is the hope and means for progress in science and in better industrial products, yet software engineering seems to be languishing in a dark prison of its own making.

VIVE LA RÉVOLUTION

It will take much hard work and planning to get academia and industry into alignment and make software engineering worthy of public trust. I don't see that we have a choice. We can't afford not to try. Worldwide software industry waste is measured in hundreds of billions of dollars annually, excluding physical (death) and social (identity theft) costs. Must heads roll before we do something?

To start this revolution, we must advance three fronts simultaneously.

Certification

Professional societies, industry associations, and government regulators must certify products in narrow specializations. We could begin with our oldest specialty, mathematical software. There is a worldwide trust of numerical analysts who have a deep understanding of algorithms, performance and error in calculating transcendental functions, numerical integration and differentiation, basic linear algebra operations, and so on. People in the know can name the mathematicians, software, hardware, and organizations to be trusted. It would be a relatively simple step for this society to define scientific certification protocols for such software-hardware implementations and to issue a seal of approval for each use by a compiler or application package. We could then relax and trust the calculations behind a spreadsheet GUI or application package. This is especially important when the industry experiences major shifts, for example, when the mainframes of the 1970s gave way to the workstations of the 1980s and when vector and parallel processors arrived on the scene.

Programming language compilers is another area that is sufficiently mature to support effective certification protocols. Existing professional societies could step up to this task. In view of the 50 years of programming language research and the potential for accumulating knowledge and skill in compiler implementation, we have little justification for using languages that fail to enforce sound practices or for tolerating compiler errors or variations among compilers of the same language.

Industry associations could establish certification protocols and centers for many consumer products that are software intensive but not a threat to public safety. Perhaps the controls for home appliances would become simpler, more uniform, and less prone to transcendental behavior.

In one certification area—medical devices and other products that directly relate to public safety—we should begin immediately. Government regulatory agencies may not do this type of certification expertly at first, but in time they should be able to perform it at least as well as they now do for prescription drugs and Florida orange juice.

A few other areas—statistical packages built on mathematical software, transaction processing and database management systems, and embedded real-time systems—are also mature enough for a meaningful measure of product certification by cohorts of well-qualified practitioners.

Licensing

Regulatory authorities must license certain practitioners and processes, such as those associated with medical devices and financial systems. Software-intensive systems now dominate many devices and processes that impinge on social and physical public well-being. The risks are unacceptably high.

Surgeons require formal training and a license, but the authors of the software inside surgical instruments do not. Software runs medical devices from pacemakers to radiation-therapy systems, but no public authority is thoughtfully and conservatively certifying that the software's authors are dependable, capable, and working according to an approved process for the application in question. Airline pilots and mechanics require formal training and a license, but not so for the authors of the avionics software systems that fly the planes. Pharmaceuticals must stand up to statistical protocols in clinical trials, but the software systems that aid in the manufacture, distribution, and tracking of these pharmaceuticals go unchecked.

The examples are endless: digital controls of nuclear power plants, air-traffic-control systems, automotive braking and (soon-to-come) steering systems, and so on and so on.

Curriculum reform

While certification and licensing apply pressure from the top down, curriculum reform attacks the problem from the bottom up. McMaster University's software engineering curriculum is the best example of how to align foundations with curricula and curricula with engineering practice (David L. Parnas, "Software Engineering Programmes Are Not Computer Science Programmes," *Ann. Software Eng*., vol. 6, 1998, pp. 19-37). The curriculum holds that software developers should understand the products or systems they are developing. It also relies on an understanding of physical systems and control engineering to exert discipline on the software. Finally, it focuses on knowing and teaching what is true of software.

I sincerely hope that after apprenticeships under masterful software engineers, graduates of the McMaster program will be armed with an advantage beyond knowledge, more than a college degree and more than successful experience—an engineering license that permits the holder to do things that those without the license cannot do.

The process of declaring individuals qualified or unqualified to participate in the development of specific software-intensive products would go a long way toward driving consensus among academics worldwide to get the curricula right.

Curricula reflect the degree of disciplinary consensus and some degree of local circumstances and strategy. If there are three schools of thought, there will be three flavors of curricula. Accreditation and standards are both a blessing and a curse: If applied too soon, they stifle development and creativity; if applied too late, the field becomes woefully suboptimal. If applied first to the best programs, they become goals for all to achieve; if applied first to weak programs, they become company to avoid.

Programming language compilers is an area that is sufficiently mature to support effective certification protocols.

The software field must find a way to bind university preparation with industry practice.

Curriculum reform is, of course, a matter of building consensus where we have too little. Mathematics is taught across the university, often to avoid foundations and treat applications directly, but a hard problem is gladly taken to a member of the mathematics faculty for expertise and assistance. English is used and taught across the university, but a member of the English faculty will be the final arbiter of a dispute on grammar, however pedantic the experience might be. Electrical engineering is the place to get advice on circuit design, and the life sciences own genetic engineering. Computer and information science and engineering, on the other hand, seem to be in everyone's job description. Worse, when a good physicist becomes a bad software developer, it is with unstinting confidence and resolve that he knows no peer. Thus, it is no surprise that when the nutritionist encounters an array of software packages that supposedly do the same job but seem to have an arbitrary design, he wonders what the computer scientists are teaching.

Curriculum reform must start with the freshman service courses. We must communicate the essence and importance of the field's foundations and at the same time differentiate some software development methods as superior in quality and productivity to the efforts of those unschooled in the foundations. We must establish both in fact and message, "Our way isn't just different; it's better in ways that matter."

It is fair to ask whether circuit engineering and genetic engineering resorted to certification and licensing. In some sense the answer is yes. There are professional engineering licenses, home wiring codes, appliance design standards, and independent laboratory testing. When electrical shock is not a threat, the threat of product recall drives high standards or, more recently, drives the functionality from hardware into software. Because of societal concern for genetically altered foods, the role of genetics in medical therapies, and the public fear of uncontrolled experimentation with life forms, we may see extensive government involvement in most aspects of genetic engineering—perhaps all except the software.

We have seen with circuit engineering—and will see with genetic engineering—close attention to preparing students for field participation and careful integration of workers in industrial practice. The software field must find a similar way to bind university preparation with industry practice.

It is never too late for a revolution, and in circuit and genetic engineering we have two worthy role models. Given the state of affairs in software engineering today and the pervasive and intrusive role of high-risk software in public welfare, I, for one, am willing to be a Sydney Carton—to act quickly to save something I cherish. In this case, the price is one of calculated restriction to save the reputation of software engineering and ensure that it is never the weakest link. Theoretically, software is the only component that can be perfect, and this should always be our starting point. ■

Jesse H. Poore is a professor of computer science at the University of Tennessee and holds the Ericsson-Harlan D. Mills Chair in Software Engineering. He is the Director of the University of Tennessee–Oak Ridge National Laboratory Science Alliance. His current interests are the economical production of high-quality software and federal science policy. Poore received a PhD in information and computer science from the Georgia Institute of Technology. He is a member of the IEEE, the IEEE Computer Society, and the ACM and is a fellow of the AAAS. Contact him at poore@cs.utk.edu.

The End of Science Revisited

John Horgan

Because science has advanced so rapidly over the past century or so, we assume that it can and will continue to do so, possibly forever. But science itself tells us that there are limits to our knowledge.

One of my most memorable moments as a science journalist occurred in December 1996, when I attended the Nobel Prize festivities in Stockholm. During the white-tie banquet for 1,500 presided over by Sweden's King and Queen, several prizewinners stood to give brief speeches. David Lee of Cornell University, a winner in physics, decried the "doomsayers" who were claiming that science is ending; the work for which he and his colleagues had been honored showed just how vital physics is.

As the audience applauded, several people at my table whispered and jabbed their fingers at me. They knew Lee was alluding to my book, *The End of Science* (Addison-Wesley, 1996), which had been stirring up trouble since its release six months earlier. The book argued that science—especially pure science, the grand quest to understand the universe and our place in it—might be entering an era of diminishing returns. As the *New York Times* put it in a front-page review, "The great days of scientific discovery are over; what science now knows is about all it will ever know."[1]

The book was denounced by eminences such as Bill Clinton's science advisor, the British minister of science, the heads of NASA and the Human Genome Project, the editors of *Science* and *Nature*, and dozens of Nobel laureates. These denunciations usually took the form not of detailed rebuttals of my arguments but of declarations of faith in scientific progress. Scientists need a certain degree of faith to bolster their confidence in the arduous quest for truth; lacking such faith, science would not have come so far so fast. But when researchers reflexively deny any evidence and arguments that challenge their faith, they violate the scientific spirit.

Perhaps recognizing this fact, some pundits reacted thoughtfully to my book. Moreover, I suspect that more than a few scientists who publicly assailed my views privately acknowledged their merit. David Lee was a case in point. When I introduced myself to him at the Nobel banquet to tell him how flattered I was that he had mentioned my book, he said he hoped that I hadn't been offended. He had enjoyed the book and had agreed with much of it, particularly the argument that achieving fundamental discoveries is becoming increasingly difficult. He just felt that the only way for scientists to truly know the limits of science is to keep trying to overcome them.

BUMPING UP AGAINST LIMITS

Today, scientists and nonscientists alike still find it hard to accept that science may be bumping up against limits. I can understand why. We have grown

0018-9162/04/$20.00 © 2004 IEEE

The more we know about the Earth, the less there is to discover.

up in a period of explosive scientific and technological progress, reflected by such measures as Moore's law. When I started working as a journalist in 1983, I wrote on an IBM Selectric typewriter. I literally cut and pasted (actually Scotch-taped) manuscripts, and slathered Whiteout on like housepaint. My first computer, a "portable" Kaypro, was as bulky as a filing cabinet and had a 64K memory. Ten years later, I was writing *The End of Science* on a Macintosh laptop. Talk about progress!

Because science has advanced so rapidly over the past century or so, we assume that it can and will continue to do so, possibly forever. But science itself tells us that there are limits to our knowledge. Relativity theory prohibits travel or communication faster than light. Quantum mechanics and chaos theory constrain our predictive ability. Evolutionary biology keeps reminding us that we are animals, designed by natural selection not for discovering deep truths of nature but for breeding. Perhaps the most important barrier to future progress in science—especially pure science—is its past success.

Postmodernist philosophers will find this a terribly naive comparison, but scientific discovery in some respects resembles the discovery of the Earth. The more we know about the Earth, the less there is to discover. We have mapped out all the continents, oceans, mountain ranges, and rivers. Every now and then, something interesting turns up. Scientists find a new species of lemur in Madagascar or exotic bacteria living in deep-sea vents. But at this point we are unlikely to discover something truly astonishing, like the lost continent of Atlantis or dinosaurs dwelling inside the Earth.

In the same way, scientists might be unlikely to discover anything surpassing the big bang, or quantum mechanics, or relativity, or natural selection, or DNA-based genetics. Nobel Prizes reflect the trend toward diminishing returns. The Russian physicist Pyotr Kapitsa discovered superfluidity in liquid helium in 1938 and won a Nobel Prize for that finding 40 years later. David Lee and his two colleagues won the 1996 prize for showing that superfluidity also occurs in a helium isotope, He-3.

If we accept that science has limits—and science tells us that it does—then the only question is when, not if, science reaches them. The American historian Henry Adams observed a century ago that science accelerates through a positive feedback effect.[2] Knowledge begets more knowledge; power begets more power. This so-called acceleration principle has an intriguing corollary: If science has limits, then it might be moving at maximum speed just before it hits the wall.

NANOTECH AND FUSION

Some researchers grant that the basic rules governing the physical and biological realms may be finite, and that we may already have them more or less in hand. But they insist that we can still explore the consequences of these rules forever and manipulate them to create an endless supply of new materials, organisms, technologies. Proponents of this position often compare science to chess. The rules of chess are quite simple, but the number of possible games that these rules can give rise to is virtually infinite.

This point is reasonable, but some enthusiasts—particularly in the trendy field of nanotechnology—take it too far. As espoused by evangelists such as Eric Drexler, nanotechnology resembles a religion more than a field of science.[3] Drexler and others proclaim that we will soon be able to reconstruct reality from the atomic scale on up in ways limited only by our imaginations. Our alchemical power to transform matter will help us achieve infinite wealth and immortality, among other perks.

Nanotechnology has also inspired some entertaining science fiction about nanobots running amok, such as Michael Crichton's novel, *Prey* (Harper Collins, 2002), and an essay by Bill Joy of Sun Microsystems titled "Why the Future Doesn't Need Us" (*Wired,* Apr. 2000). But to the extent that researchers have experimental experience at the nanoscale level, they tend to doubt the more far-fetched claims—positive or negative—that nanotechnologists make. "The level of hard science in these ideas is really low," the Harvard chemist George Whitesides remarked recently.[4] After all, nanotechnology is just a glossy wrapping for nitty-gritty work done in chemistry, molecular biology, solid-state physics, and other fields that investigate nature at small scales. Experiments often reveal that what should work in principle fails in practice.

Take nuclear fusion. Physicists such as Hans Bethe elucidated the basic rules governing fusion—the process that makes the sun and other stars shine—more than 60 years ago. By the 1950s, this knowledge had spawned the most fearsome technology ever invented: thermonuclear weapons. Next, physicists hoped to harness fusion for a more benign application: a clean, economical, boundless source of energy.

When my career began in the early 1980s, fusion-energy research was a staple of the science beat: Keep the money coming, researchers promised, and in 20 years we will give you energy too cheap to meter. Twenty years later, the US has drastically reduced its budget for magnetic-confinement fusion, formerly the leading candidate for power generation.

Inertial-confinement fusion—in which giant lasers blast tiny fuel pellets—has fared better, but only because its main application is nuclear-weapons research. Surveys of sustainable-energy methods rarely even give fusion a courtesy mention any more—and please don't bring up cold fusion. Diehards may still cling to the dream, but realists acknowledge that fusion energy is effectively dead, the victim of technical, economic, and political constraints.

COMPUTERS AND CHAOPLEXITY

Of course, technological advances often enable researchers to overcome seemingly insurmountable obstacles. Computers in particular have vastly increased scientists' capacity for data acquisition, analysis, storage, and communication. Innovations such as optical and quantum computing may extend the reign of Moore's law indefinitely—although, as the astute computer theorist Rolf Landauer often warned, quantum computers may be so sensitive to thermal noise and other disruptions that they remain a laboratory curiosity.[5]

But adherents of certain computer-driven fields—notably artificial life, chaos, and complexity—seem to view computers not as tools but as wands that will magically solve even the toughest puzzles. In *The End of Science* I lumped chaos and complexity together under a single term, *chaoplexity*, because, after talking to scores of people in both fields, I realized there is no significant difference between them.

Chaoplexologists have argued that with more powerful computers and mathematics they can solve conundrums resistant to conventional scientific reductionism, particularly in "soft" fields such as ecology, psychology, economics, and other social sciences. Stephen Wolfram recently reiterated these claims in his magnum opus, *A New Kind of Science*, which touts cellular automatons as the key that will unlock all the riddles of nature.[6]

As many critics have noted, Wolfram's "new" science actually dates back at least to John von Neumann, the inventor of cellular automatons. Von Neumann and many others have shown that simple rules, when followed by a computer, can generate patterns that appear to vary randomly as a function of time or scale. Let's call this illusory randomness "pseudonoise." Two paradigmatic examples of pseudonoisy systems are the Mandelbrot set, discovered by Benoit Mandelbrot, and "Life," a cellular automaton devised by John Conway.

Chaoplexologists such as Wolfram assume that much of the noise that seems to pervade nature is actually pseudonoise, the result of some underlying, deterministic algorithm. This hope has been nurtured by certain genuinely significant findings, such as Mitchell Feigenbaum's discovery more than 20 years ago that gushing faucets and similar turbulent systems, although they seem hopelessly noisy, actually adhere to a rule called period doubling.

But a gushing faucet is laughably simple compared to a stock market, a human brain, a genome, or a rain forest. These hideously complex phenomena—with their multitudes of variables—have shown no signs of yielding to the efforts of chaoplexologists. One reason may be the notorious butterfly effect, elucidated by Edward Lorenz in the 1960s. The butterfly effect limits both prediction and retrodiction, and hence explanation; specific events cannot be ascribed to specific causes with complete certainty. This is something that has always puzzled me about chaoplexologists: According to the butterfly effect—one of their fundamental tenets—achieving many of their goals may be impossible.

SEEKING ARTIFICIAL COMMON SENSE

One would think that chaoplexologists would also be chastened by the failure of artificial intelligence to live up to expectations. AI researchers have come up with some useful inventions, including devices that can translate languages, recognize voices, judge loan applications, interpret cardiograms, and play chess. But these advances pale beside the hopes that AI pioneers once had for their field.

AI researchers have come up with some useful inventions, but these advances pale beside the hopes that AI pioneers once had for their field.

In 1984, I edited an article for *IEEE Spectrum* in which AI expert Frederick Hayes-Roth predicted that expert systems were going to "usurp human roles" in professions such as medicine, science, and the business world.[7] When I called Hayes-Roth recently to ask how he thought his predictions had held up, he cheerfully admitted that the field of expert systems, and AI generally, had stalled since their heyday in the early 1980s.

Not all AI'ers have conceded defeat. Hans Moravec and Ray Kurzweil still prophesize that machines will soon leave flesh-and-blood humans in their cognitive dust.[8,9] (In *The End of Science*, I called this sort of speculation "scientific theology.")

> **There are clear signs that mathematics is already outrunning our limited cognitive capabilities.**

These cyberprophets cite Garry Kasparov's loss to the IBM computer Deep Blue in 1997 as a portent of AI's impending triumph.

Actually, that contest underscored the limitations of artificial intelligence. Chess, with its straightforward rules and tiny, Cartesian playing field, is a game tailor-made for computers. Deep Blue, whose five human handlers included the best chess programmers in the world, was a prodigiously powerful machine, capable of examining hundreds of millions of positions each second. If this silicon monster had to strain so mightily to beat a mere human at chess, what hope is there that AI engineers will ever create HAL, the lip-reading killer in the film *2001*?

Some prominent AI'ers acknowledge that computers will probably never be as smart as HAL. Although computers excel at tasks that can be precisely defined, such as chess, they will never acquire the flexible, all-purpose intelligence—the ordinary common sense—that every normal human acquires by the age of five or so.

In *HAL's Legacy: 2001's Computer as Dream and Reality*,[10] a recent collection of essays by AI experts, David Kuck stated flatly, "Under any general definition ... AI so far has been a failure." Roger Shank declared that HAL "is an unrealistic conception of an intelligent machine" and "could never exist." The best that computer scientists can hope to do is to create machines "that will know a great deal about what they are supposed to know about and miserably little about anything else." Even Marvin Minsky, who had predicted in the mid-1960s that computers would be as smart as humans within three to eight years, admitted that "we really haven't progressed too far toward a truly intelligent machine."

One AI'er still pursuing the original vision of a computer with an all-purpose rather than highly specialized intelligence is Douglas Lenat. For 20 years, he has been trying to create a software program that mimics common-sense knowledge. Lenat's goal was for this program, called Cyc, to become more or less autonomous, capable of acquiring new knowledge by scanning newspapers, books, and other sources of information. In 1997, Lenat predicted that by 2001 Cyc would become a "full-fledged creative member of a group that comes up with new discoveries. Surprising discoveries. Way out of boxes."[11]

The world is still waiting. Rodney Brooks of MIT has complained that Cyc's intelligence has nothing in common with the human variety; far from interacting with the world in flexible or creative ways, Cyc is really just a fancy dictionary. When I interviewed him in 1997, Brooks said, "Ultimately you have to ground it out. You have to attach it to some other sensory motor experience, and that's what I think he's missing."

THE END OF MATHEMATICS?

Some of the harshest attacks on *The End of Science* came from scientists whose fields I hadn't bothered to denigrate.

When I encountered him at a New York Academy of Sciences meeting in November 1996, chemist and Nobel laureate Dudley Herschbach commented that not only had I failed to include a chapter on chemistry, my index included only two measly references to the subject. I didn't have the heart to tell Herschbach that chemistry just seemed too passé to dwell on. In a 1992 interview, Linus Pauling assured me that he had laid out the basic principles by 1930, and who was I to disagree with Linus Pauling? (Also, to be honest, I have always found chemistry excruciatingly dull.)

Similarly, the mathematician John Casti complained in *Nature*[12] that I had neglected to include a chapter titled "The End of Mathematics." Actually, I had planned to include such a chapter; I just ran out of gas. I would have agreed with Casti that there are no limits, in principle, to mathematics, because mathematics is a process of invention rather than of discovery; in that sense, mathematics resembles art or music more than pure science.

Moreover, Kurt Godel's incompleteness theorem established that any moderately complex system of axioms gives rise to questions that cannot be answered with those axioms. By adding to their base of axioms, mathematicians can keep expanding the realm within which they play, posing new conjectures and constructing new proofs, forever. The question is, how comprehensible will these proofs be?

There are clear signs that mathematics is already outrunning our limited cognitive capabilities. The largest conventional proof ever constructed is the classification of finite simple groups, also called "the enormous theorem." In its original form, this theorem consisted of some 500 separate papers, totaling more than 20,000 pages, written by more than 100 mathematicians over a period of 30 years. It has been said that the only person who really understood the proof was Daniel Gorenstein of Rutgers University, who served as a kind of general contractor for the project. Gorenstein died in 1992.

A growing number of mathematical proofs are constructed with the help of computers, which can carry out calculations far beyond the capability of mere mortals. The first such proof, constructed in 1976, demonstrated the truth of the four-color theorem, which states that four hues are sufficient to color even an infinitely broad map so that no identically colored countries share a border. The proof depended on a calculation that took a computer 1,000 hours to complete.

Mathematicians like to wax rhapsodic about the elegance, beauty, and depth of proofs. But computer proofs yield truth without insight or understanding. So yes, mathematics can, in principle, continue forever. The problem is that no mere human will be able to understand it. In 1997, the mathematician Ronald Graham said to me,"We're not very well adapted for thinking about the space-time continuum or the Riemann hypothesis. We're designed for picking berries or avoiding being eaten."

THE UNDISCOVERED MIND

Some criticism of *The End of Science*, I admit, was on target. When I met the eminent British biologist Lewis Wolpert at a biology conference in London in 1997, he declared that my book was "appalling, absolutely appalling!" He was particularly upset by the chapter titled "The End of Neuroscience." How dare I dismiss all the vast and vital research on the brain in a single chapter, which focused not on genuine neuroscientists but on Francis Crick, a molecular biologist, and Roger Penrose, a physicist? Neuroscience was just beginning, not ending!

Wolpert stalked away before I could tell him that I thought his objection was fair. I had already decided that mind-related science had so much potential to alter our world, both intellectually and materially, that it deserved a more serious, detailed critique than it got in *The End of Science*. I offered such a critique in my second book, *The Undiscovered Mind: How the Brain Defies Replication, Medication, and Explanation* (Free Press, 1999). As the subtitle suggests, this book looks at attempts to explain the mind, treat its disorders, and replicate its functions in computers. In addition to neuroscience, this book also covers psychiatry, psychopharmacology, behavioral genetics, evolutionary psychology, and artificial intelligence.

My conclusion was that these fields have largely failed to live up to their advertising. In spite of all this investigation, the mind remains largely undiscovered. In *The End of Science*, I coined the term "ironic science" to describe science that never gets a firm grip on reality and thus doesn't converge on the truth. Ironic science is more like philosophy, literary criticism, or even literature than like true, empirical science. Ironic science crops up in the so-called hard sciences, such as physics and cosmology. Superstring theory is my favorite example of ironic science. It's science fiction with equations.

But ironic science is most pervasive in fields that address human cognition and behavior. Theories of human nature never really die—they just go in and out of fashion. Often, old ideas are simply repackaged. The 18th-century pseudoscience of phrenology is reincarnated as cognitive modularism. Eugenics evolves into behavioral genetics. Social Darwinism mutates into sociobiology, which in turn is reissued as evolutionary psychology.

Neuroscience's most important discovery may be that different regions of the brain are specialized for carrying out different functions.

One astonishingly persistent theory is psychoanalysis, which Freud invented a century ago. Once defined as "the treatment of the id by the odd," psychoanalysis has been subjected to vicious criticism since its birth. Some Freud-bashers imply that while French philosophers and other fuzzy-brained sorts may still fall under Freud's spell, real scientists are immune to his charms. Actually, many prominent neuroscientists—such as Nobel laureates Gerald Edelman[13] and Eric Kandel[14]—still defend Freudian theory.

Freud's ideas have persisted not because they have been scientifically confirmed but because a century's worth of research has not produced a paradigm powerful enough to render psychoanalysis obsolete once and for all. Freudians cannot point to unambiguous evidence of their paradigm's superiority, but neither can proponents of more modern paradigms.

THE HUMPTY DUMPTY DILEMMA

Neuroscience was supposed to deliver us from this impasse. Neuroscientists have acquired an astonishing ability to probe the brain with microelectrodes, magnetic resonance imaging, positron-emission tomography, and other tools. Neuroscience is clearly advancing. It's getting somewhere. But where? So far, neuroscience has had virtually no payoff in terms of diagnosing and treating such complex mental illnesses as schizophrenia and manic depression. It has failed to winnow out all the competing unified theories of human nature, whether psychoanalysis, behaviorism, connectionism, or evolutionary psychology.

Neuroscience's most important discovery may be that different regions of the brain are specialized for carrying out different functions. For example, the visual cortex contains one set of neurons dedicated to orange-red colors, another to objects with

Mystical experiences can't give us the kind of absolute knowledge that we crave.

high-contrast diagonal edges, and still another to objects moving rapidly from left to right. The question is, how does the brain coordinate and integrate the workings of these highly specialized parts to create a mind? Neuroscientists have no idea. This is sometimes called the binding problem, but I prefer to call it the Humpty Dumpty dilemma. Neuroscientists can take the brain apart, but they can't put it back together again.

Particle physicists once faced a similar dilemma. In the 1950s, the number of particles detected in accelerators proliferated wildly. Theorists trying to make sense of it all were baffled. Then a brilliant young physicist named Murray Gell-Mann showed that all these different particles were made of a few more fundamental particles called quarks. Order emerged from chaos.

But in terms of sheer complexity, particle physics is a child's game compared to neuroscience. All protons, neutrons, and electrons are identical; a theory that applies to one particle applies to all. But each brain is unique, and it changes every time its owner is spanked, learns the alphabet, reads *Thus Spake Zarathustra*, takes LSD, falls in love, gets divorced, undergoes Jungian dream therapy, or suffers a stroke. Scientists cannot simply ignore each individual's uniqueness, because it is central to our humanity. This fact immensely complicates the search for a unified theory of the brain and mind.

Some scientists have reluctantly concluded that science may never fully solve the mysteries of the brain and mind. This position is sometimes called "mysterianism." One well-known mysterian is the psychologist Howard Gardner, who contends that neither psychology, neuroscience, nor any other field has provided much illumination of such perennial riddles as consciousness and free will.[15] These subjects "seem particularly resistant to [scientific] reductionism," Gardner said. He suggests that psychologists may advance by adopting a more "literary" style of investigation and discourse—the style that Freud exemplified.

A surprising number of scientists have proposed that investigations of mystical states of consciousness—such as those induced by meditation, prayer, or psychedelic drugs—might yield insights into the mind that complement or transcend those of science. This notion has inspired a series of highly publicized meetings—most recently at MIT last September—between prominent scientists and the Dalai Lama, the leader of Tibetan Buddhism. I explore this possibility in my most recent book, *Rational Mysticism* (Houghton Mifflin, 2003). While researching the book, I spoke to a wide variety of scholars studying mysticism, including neuroscientists, psychologists, and anthropologists.

Early on, one of these sources warned me that you can't comprehend mystical experiences if you've never had one. With that in mind, I learned Zen meditation. I had my temporal lobes electromagnetically tickled by a device called the "God machine." I consumed a nauseating psychedelic brew called ayahuasca, which serves as a religious sacrament for Indians in the Amazon. I concluded that mystical experiences can't give us the kind of absolute knowledge that we crave. Quite the contrary. Mystical experiences do not give us The Answer to the riddle of our existence; rather, they let us see just how truly astonishing the riddle is. Instead of a big "Aha!" we get a big "Huh?"

WHAT'S MY POINT?

After I gave a talk on the limits of science at Caltech, a neuroscientist in the audience angrily asked me what my point was. Did I think he and his colleagues should simply give up their research? Should Congress take its funding away? Good questions.

I hope I don't sound disingenuous when I say that I would hate to see my prophesies become self-fulfilling—not that there is any chance of that anyway. I always encourage young people to become scientists, and I would be delighted if my children pursued that path someday.

Grant for a moment that I am right—that science will never again yield revelations as monumental as the theory of evolution, general relativity, quantum mechanics, the big bang theory, DNA-based genetics. For example, physicists will never discover a unified theory that reveals "the mind of God," as Stephen Hawking once put it.[16] And grant that some far-fetched goals of applied science—such as immortality, superluminal spaceships, and superintelligent machines—may forever elude us.

I nonetheless have no doubt that researchers will find better treatments for cancer, schizophrenia, AIDS, and other diseases; more benign sources of energy (other than fusion); and more convenient contraception methods. In the realm of pure science, scientists will surely gain a better understanding of how galaxies formed, how life began on Earth, how Homo sapiens became so smart so fast, how neural processes generate awareness, how a single fertilized egg turns into a fruit fly or a congressman.

But I would like to see a greater recognition of science's limitations—particularly in mind-related fields, where our desire for self-knowledge can make us susceptible to pseudoscientific cults such as Marxism, social Darwinism, eugenics, and psychoanalysis. Science is never more dangerous than when it seeks to tell us what we are, what we can be, and even what we should be.

My goal is to foster an attitude that I call "hopeful skepticism." Too much skepticism culminates in a radical postmodernism that denies the possibility of achieving any truth. Too little skepticism leaves us prey to peddlers of scientific snake oil. But just the right amount of skepticism—mixed with just the right amount of hope—can protect us from our lust for answers while keeping us open-minded enough to recognize genuine truth if and when it arrives. ■

References

1. N. Angier, "Dark Days in the Laboratory," *New York Times*, book review, 30 June 1996.
2. H. Adams, *The Education of Henry Adams*, Houghton Mifflin, 1918.
3. K. E. Drexler, *Engines of Creation: The Coming Era of Nanotechnology*, Anchor, 1986.
4. D.E. Duncan, "Discover Dialogue: Chemist George Whitesides," *Discover*, Dec. 2003, p. 24
5. R. Landauer, "The Limits of Scientific Knowledge," Santa Fe Institute meeting, 25 May 1994.
6. S. Wolfram, *A New Kind of Science*, Wolfram Media, 2002.
7. F. Hayes-Roth, "The Machine as Partner of the New Professional," *IEEE Spectrum*, June 1984, pp. 28-31.
8. H.P. Moravec, *Robot: Mere Machine to Transcendent Mind*, Oxford Univ. Press, 1998.
9. R. Kurzweil, *The Age of Spiritual Machines: When Computers Exceed Human Intelligence*, Oxford Univ. Press, 1998.
10. D.G. Stork, *HAL's Legacy: 2001's Computer as Dream and Reality*, MIT Press, 1997.
11. S. Garfinkel, "Happy Birthday, HAL," *Wired*, Jan. 1997, p. 188.
12. J.L. Casti, "Lighter than Air," *Nature*, 29 Aug. 1996, p. 769.
13. G.M. Edelman, *Bright Air, Brilliant Fire: On the Matter of the Mind*, Basic Books, 1993.
14. E.R. Kandel, "A New Intellectual Framework for Psychiatry," *Am. J. Psychiatry*, Apr. 1998, pp. 457-469.
15. H. Gardner, "Scientific Psychology: Should We Bury It or Praise It?" *New Ideas in Psychology*, vol. 10, no. 2, 1992, pp. 179-190.
16. S.W. Hawking, *A Brief History of Time*, Bantam Books, 1998.

***John Horgan**, a former senior writer at* Scientific American, *has written for the* New York Times, Discover, Slate, Time, *and other publications. He has posted essays, articles, and profiles at www.johnhorgan.org. Contact him at jhorgan@highlands.com.*

2004 IEEE Computer Society Professional Membership/Subscription Application

Membership and periodical subscriptions are annualized to and expire on 31 December 2004.
Pay full or half-year rate depending upon the date of receipt by the IEEE Computer Society as indicated below.

Membership Options*

All prices are quoted in U.S. dollars

	FULL YEAR Applications received 16 Aug 03 - 29 Feb 04	HALF YEAR Applications received 1 Mar 04 - 15 Aug 04
1 I do not belong to the IEEE, and I want to join **just** the Computer Society	$ 99 ❑	$50 ❑
2 I want to join **both** the Computer Society and the IEEE:		
I reside in the United States	$189 ❑	$95 ❑
I reside in Canada	$170 ❑	$85 ❑
I reside in Africa/Europe/Middle East	$166 ❑	$83 ❑
I reside in Latin America	$159 ❑	$80 ❑
I reside in Asia/Pacific	$160 ❑	$80 ❑
3 I already belong to the IEEE, and I want to join the Computer Society. (IEEE members need only furnish name, address, and IEEE number with payment.)	$ 42 ❑	$21 ❑

Are you now or were you ever a member of the IEEE?
Yes ❑ No ❑ If yes, provide member number if known: _______________

Add Periodicals**

	ISSUES PER YEAR	FULL YEAR Applications received 16 Aug 03 - 29 Feb 04 PRINT	ELECTRONIC	COMBO	HALF YEAR Applications received 1 Mar 04 - 15 Aug 04 PRINT	ELECTRONIC	COMBO
IEEE Computer Society Digital Library BEST DEAL	n/a	n/a	$109 ❑	n/a	n/a	$55 ❑	n/a
Computing in Science and Engineering	6	$42 ❑	$34 ❑	$55 ❑	$21 ❑	$17 ❑	$28 ❑
IEEE Computer Graphics and Applications	6	$39 ❑	$31 ❑	$51 ❑	$20 ❑	$16 ❑	$26 ❑
IEEE Design & Test of Computers	6	$37 ❑	$30 ❑	$48 ❑	$19 ❑	$15 ❑	$24 ❑
IEEE Intelligent Systems	6	$37 ❑	$30 ❑	$48 ❑	$19 ❑	$15 ❑	$24 ❑
IEEE Internet Computing	6	$39 ❑	$31 ❑	$51 ❑	$20 ❑	$16 ❑	$26 ❑
IT Professional	6	$40 ❑	$32 ❑	$52 ❑	$20 ❑	$16 ❑	$26 ❑
IEEE Micro	6	$37 ❑	$30 ❑	$48 ❑	$19 ❑	$15 ❑	$24 ❑
IEEE MultiMedia	4	$35 ❑	$28 ❑	$46 ❑	$18 ❑	$14 ❑	$23 ❑
IEEE Pervasive Computing	4	$41 ❑	$33 ❑	$53 ❑	$21 ❑	$17 ❑	$27 ❑
IEEE Security & Privacy	6	$41 ❑	$33 ❑	$53 ❑	$21 ❑	$17 ❑	$27 ❑
IEEE Software	6	$44 ❑	$35 ❑	$57 ❑	$22 ❑	$18 ❑	$29 ❑
IEEE/ACM Transactions on Computational Biology and Bioinformatics NEW	4	$35 ❑	$28 ❑	$46 ❑	$18 ❑	$14 ❑	$23 ❑
IEEE/ACM Transactions on Networking†	6	$44 ❑	$33 ❑	$55 ❑	$22 ❑	$17 ❑	$28 ❑
IEEE Transactions on:							
Computers	12	$41 ❑	$33 ❑	$53 ❑	$21 ❑	$17 ❑	$27 ❑
Dependable and Secure Computing NEW	4	$31 ❑	$25 ❑	$40 ❑	$16 ❑	$13 ❑	$20 ❑
Information Technology in Biomedicine†	4	$35 ❑	$25 ❑	$44 ❑	$18 ❑	n/a	$22 ❑
Knowledge and Data Engineering	12	$43 ❑	$34 ❑	$56 ❑	$22 ❑	$17 ❑	$28 ❑
Mobile Computing	4	$30 ❑	$24 ❑	$39 ❑	$15 ❑	$12 ❑	$20 ❑
Multimedia†	6	n/a	n/a	$38 ❑	n/a	n/a	n/a
NanoBioscience†	4	$30 ❑	$24 ❑	$38 ❑	$15 ❑	n/a	$19 ❑
Parallel and Distributed Systems	12	$40 ❑	$32 ❑	$52 ❑	$20 ❑	$16 ❑	$26 ❑
Pattern Analysis and Machine Intelligence	12	$44 ❑	$35 ❑	$57 ❑	$22 ❑	$18 ❑	$29 ❑
Software Engineering	12	$38 ❑	$30 ❑	$49 ❑	$19 ❑	$15 ❑	$25 ❑
Visualization and Computer Graphics	6	$34 ❑	$27 ❑	$44 ❑	$17 ❑	$14 ❑	$22 ❑
VLSI Systems†	12	n/a	n/a	$28 ❑	n/a	n/a	$14 ❑
IEEE Annals of the History of Computing	4	$31 ❑	$25 ❑	$40 ❑	$16 ❑	$13 ❑	$20 ❑

Choose PRINT for paper issues delivered via normal postal channels.
Choose ELECTRONIC for 2004 online access to all issues published from 1988 forward.
Choose COMBO for both print and electronic.

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

____
Expiration date (month/year)

____
Signature

USA-only include 5-digit billing zip code

☐☐☐☐☐

* Member dues include $17 for a 12-month subscription to *Computer.*
** Periodicals purchased at member prices are for the member's personal use only.
*** Canadian residents add 15% HST or 7% GST to total. AL, AZ, CO, DC, NM, and WV add sales tax to all periodicals. GA, IN, KY, MD, and MO add sales tax to print and combo periodicals. NY add sales tax to electronic periodicals. European Union residents add VAT tax to electronic periodicals.
**** 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.
† Not part of the IEEE Computer Society Digital Library. Electronic access is through www.ieee.org/ieeexplore.

For fastest service, apply online at http://computer.org/join

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
PO Box 3014
Los Alamitos, CA 90720-1314 USA
Phone: +1 714 821 8380
Fax: +1 714 821 4641
E-mail: help@computer.org

Allow up to 8 weeks to complete application processing. Allow a minimum of 6 to 10 weeks for delivery of print periodicals.

IF4A

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

APPLICATION MUST BE SIGNED

NOTE: In order for us to process your application, you must complete and return both sides of this form.

IEEE

IEEE COMPUTER SOCIETY

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, 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 ____________

2003 Technology Roadmap for Semiconductors

This update to the 2001 ITRS Roadmap shows the industry shifting its focus toward systems on chip, wireless computing, and mobile applications.

Don Edenfeld
Intel

Andrew B. Kahng
University of California, San Diego

Mike Rodgers
Intel

Yervant Zorian
Virage Logic

Two years ago, our report titled "2001 Technology Roadmap for Semiconductors" (*Computer*, Jan. 2002, pp. 42-53) summarized our assessment of the opportunities and challenges awaiting the semiconductor industry in its continued pursuit of Moore's law. The 2003 edition updates this effort and reveals a significant shift in the industry's focus.

Today, the introduction of new technology solutions is increasingly application-driven, with products for different markets using different technology combinations at different times. General-purpose digital microprocessors for personal computers have been joined by mixed-signal systems for wireless communication and embedded applications. Battery-powered mobile devices are as strong a driver as wall-plugged servers. System-on-chip (SoC) and system-in-package (SiP) designs that incorporate building blocks from multiple sources are supplanting in-house, single-source chip designs.

SYSTEM DRIVERS

The International Technology Roadmap for Semiconductors (ITRS) provides quantified, self-consistent models of canonical products—*system drivers*—that drive the semiconductor industry. These models support extrapolation of future technology requirements for basic circuit *fabrics*—processor, analog/mixed-signal, and embedded memory—as well as the SoC products they comprise. The 2003 ITRS system drivers chapter presents an overarching SoC context for future semiconductor products, along with new discussions of technology requirements for analog/mixed-signal and embedded memory fabrics.

Systems on chip

The SoC product class is a yet-evolving design style that integrates technology and design elements from other system driver classes into a wide range of high-complexity, high-value semiconductor products. SoC manufacturing and design technologies are typically developed originally for high-volume custom drivers such as processors, field-programmable gate arrays (FPGAs), or memories. SoCs integrate building blocks from the other system driver classes and increasingly subsume the application-specific integrated circuit (ASIC) category.

SoCs exist to provide low cost and high integration. Lower implementation cost requires greater reuse of intellectual property as well as platform-based design, silicon implementation regularity, or other novel circuit and system architecture paradigms. Cost considerations also drive the deployment of low-power process and low-cost packaging solutions, along with fast-turnaround-time design methodologies. The latter, in turn, require new standards and methodologies for IP description and test,

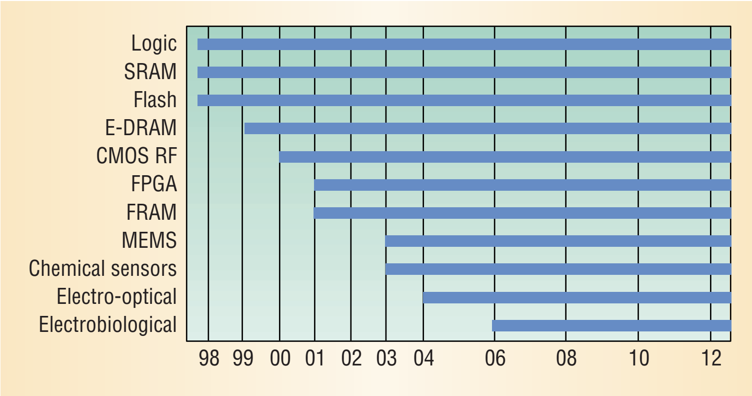


Figure 1. Existing and predicted first integrations of SoC technologies with standard CMOS processes.

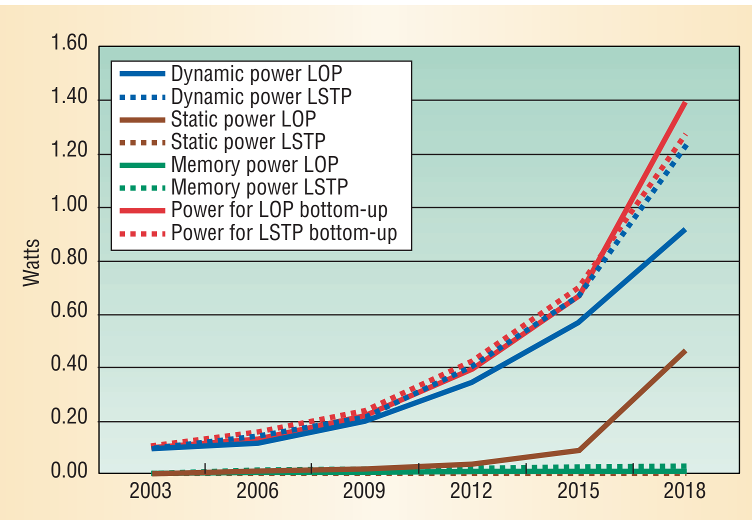


Figure 2. Total chip power trend, in watts, for a SoC-LP personal-digital-assistant application, for both low-operating-power (LOP) and low-standby-power (LSTP) modes.

block interface synthesis, and so on. Heterogeneous integration requires merging reprogrammable, memory, analog and RF, MEMS, and software elements, as well as chip-package-system cooptimization. Thus, SoCs provide the driver for convergence of multiple technologies not only in the same system package, but also potentially in the same manufacturing process. Overall, SoC designs present many challenges to design, test, process integration, and packaging technologies. Most daunting among these challenges are

- design productivity improvement of greater than 100 percent per node, including platform-based design and integration of programmable logic fabrics;
- power management, especially for low-power, wireless, multimedia applications;
- system-level integration of heterogeneous technologies, including MEMS and optoelectronics; and
- development of SoC test methodology, including test reusability and analog/digital built-in self-test (BIST).

Multitechnology integration. Considerations such as cost, form factor, connection speed and overhead, and reliability drive the need to build heterogeneous systems on a single chip. As process technologists seek to meld CMOS with MEMS, bio and chemical sensors, and so on, process complexity and production cost become major constraints. The total cost of processing is difficult to predict for future materials and new combinations of processing steps. However, at present, cost considerations limit the number of technologies on a given SoC: Processes are increasingly modular, but the modules are not generally stackable.

CMOS integration with other technologies depends not only on basic technical advances but also on being more cost-effective than multiple-die SiP alternatives. Figure 1 shows how first integrations of each technology within standard CMOS processes—not necessarily together with other technologies, and not necessarily in volume production—might evolve.

The power challenge. Recent ITRS editions project the implications of *power management*—a grand challenge for the semiconductor industry—on the achievable space of SoC designs. From the functionality of a canonical low-power multimedia PDA application, an architecture of processor, memory, and interface blocks can be inferred. Chip power requirements can then be estimated bottom up from the implied logic and memory content, and from the roadmap for process and circuit parameters.

Figure 2 shows the bottom-up *lower bound* for total chip power at an operating temperature of 100° C, even given the assumption of multiple device technologies integrated within a single core to afford greater control of dynamic power, standby power, and performance. The salient observation is that significant advances are required in architecture and design technology to meet the system targets of 100 mW peak and 2 mW standby power.

Mixed-signal chips

Analog/mixed-signal (AMS) chips at least partially deal with input signals whose precise values matter. This broad class includes RF, analog, analog-to-digital, and digital-to-analog converters—and, more recently—many mixed-signal chips in which at least part of the chip design must measure signals with high precision. These chips have design and process technology demands that differ from those for digital circuits. Technology scaling is not necessarily helpful for analog circuits because dealing with precision requirements or signals from a fixed voltage range is more difficult with scaled voltage supplies. In general, AMS circuits and process technologies present severe challenges to cost-effective CMOS integration.

The need for precision also affects tool requirements for analog design. Digital circuit design creates a set of rules for correct logic gate function; as long as developers follow these rules, precise calculation of exact signal values is unnecessary. Analog designers, on the other hand, must concern themselves with many second-order effects to obtain the required precision they require. Relevant issues include coupling and asymmetries. Analysis tools for these issues are mostly in place but require expert users, while synthesis tools are immature. Manufacturing test for AMS circuits poses a problem that remains essentially unsolved.

Most analog and RF circuitry in today's high-volume applications resides in SoCs. The economic regime of a mainstream product is usually highly competitive: It has a high production volume and hence requires a high level of R&D investment; thus, technology requirements drive mixed-signal technology as a whole. Mobile communication platforms are the highest-volume circuits driving the needs of mixed-signal circuits.

Mixed-signal evolution. The interplay between cost and performance determines mixed-signal driver evolution, including its scope of application. Although mixed-signal *performance* is an important metric, *cost* of production is also critical for practical deployment of AMS circuits. Together, cost and performance determine the sufficiency of given technology trends relative to existing applications. They also determine the potential of given technologies to enable and address entirely new applications.

Cost estimation. In mixed-signal designs, chip area determines only one of several cost factors. The area that analog circuits occupy in a SoC typically ranges from 5 to 30 percent. Economic forces to reduce mixed-signal area are therefore not as strong as those affecting logic or memory. Related considerations include the following:

- sometimes, shifting the system partitioning between the analog and digital parts can reduce the analog area;
- introducing high-performance analog devices increases process complexity so that solutions occupy less area at the expense of increased total cost;
- technology choices can impact design cost by introducing greater risk of multiple hardware passes;
- parametric yield sensitivities can impact manufacturing cost; and
- a SiP multidie solution can be more cost-effective than a single SoC solution.

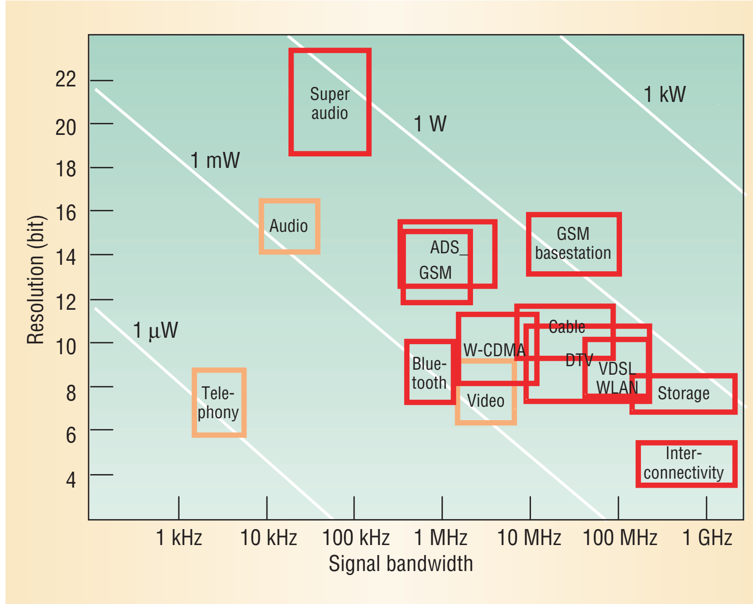


Figure 3. Recent analog-to-digital converter (ADC) performance needs for important product classes.

Such considerations make mixed-signal design cost estimation difficult. We can attempt to quantify mixed-signal cost by first restricting our attention to high-performance applications, since these also drive technology demands. Because analog features are embodied as high-performance passives or analog transistors, that area can be taken as a proxy for cost. Since improving digital density drives transistor scaling, analog transistors can simply follow, and it isn't necessary to specifically address their layout density. At the same time, embedded passives determine total area in most current AMS designs, whose area consumption dominates a system's mixed-signal costs.

Estimating technology sufficiency. Figure 3 shows requirements for analog-digital converter (ADC) circuits in terms of a power-performance relationship. Under conditions of constant performance, a straight line with slope of –1 represents constant power consumption. Increasing performance—achievable with better technology or circuit design—is equivalent to a shift of the power consumption lines toward the upper right. The data shows a very slowly moving technological barrier line for ADCs for a power consumption of 1 W. Most of today's ADC technologies lie below the 1 W barrier, and

The challenge for compound semiconductors is to increase the number of devices per unit area and integrate them with CMOS processing.

near-term solutions for moving the barrier more rapidly are unknown.

Although ADC performance has improved at a rate adequate for handset applications, it clearly has failed to do so for applications such as digital linearization of GSM base stations, or handheld and mobile high-data-rate digital-video applications. Assuming progress at recent rates, manufacturing ADCs with adequate performance in volume will not occur until 2010 or later. Silicon and silicon germanium technologies have the necessary bit resolution, but not the speed, to provide such performance now; on the other hand, III-V compound semiconductor technologies have the speed but not the bit resolution. The challenge for compound semiconductors then is to increase the number of devices per unit area and to integrate them with CMOS processing.

Enabling new applications. In the semiconductor industry, improving technology and design performance enables new applications, which pushes the industry into new markets. Researchers also can use mixed-signal design analysis to estimate design needs and feasibility for future applications and new markets.

Alternatively, when they know the specifications of a new product, researchers can design the technology needed to fulfill these specifications and estimate the timeframe in which the semiconductor industry will be able to build that product with acceptable cost and performance. The ability to build high-performance mixed-signal circuitry at low cost will continuously drive the semiconductor industry to develop new products and markets.

Mixed-signal challenges. For most of today's mixed-signal designs, a voltage difference represents the processed signal so that the supply voltage determines the maximum signal. Decreasing supplies—a consequence of constant-field scaling—mean decreasing the maximum achievable signal level. This strongly impacts mixed-signal product development for SoC solutions. Typical development time for new mixed-signal parts takes much longer than for digital and memory parts, which makes lack of design resources another key challenge.

Overall, the most daunting mixed-signal challenges are decreasing supply voltage, increasing relative parametric variations, increasing numbers of analog transistors per chip, increasing processing speed, increasing crosstalk, and shortage of design skills and productivity.

Embedded memory

SoC designs contain an increasing number and variety of embedded RAM, ROM, and register file memories. Interconnect and I/O bandwidths, design productivity, and system power limits all point to a continuing trend of high levels of memory integration in microelectronic systems. Driving applications for embedded memory technology include code storage in reconfigurable applications, data storage in smart or memory cards, and the high memory content and high-performance logic found in gaming or mass storage systems.

Opportunities and constraints. The balance between logic and memory content reflects overall system cost, power and I/O constraints, hardware-software organization, and overall system and memory hierarchy. With respect to cost, the device performance and added mask levels of monolithic logic-memory integration must be balanced against chip-laminate-chip or other SiP integration alternatives. Levels of logic-memory integration also reflect tradeoffs in hardware-software partitioning as well as code-data balance.

I/O pin count and signaling speeds determine how system organization trades off bandwidth versus storage: Memory access can be made faster at the cost of peripheral overhead by organizing memory in higher or lower bank groups. Access speed also depends on how pin count and circuit complexity are balanced between high-speed, low-pin-count connections and higher-pin-count, lower-speed connections.

In the traditional processor architecture domain, memory hierarchy is crucial in matching processor speed requirements to memory access capabilities. This has led to the introduction of several layers of hardware-controlled caches between main memory and foreground memory in the processor core. Typically, one physical cache memory is present at each layer. However, the choice of hierarchy also has strong implications for power. Conventional architectures increase performance largely at the cost of energy-inefficient control overheads by using, for example, prediction and history mechanisms and extra buffers arrayed around highly associative caches. From the system viewpoint, the embedded multimedia and communication applications that dominate on portable devices can profit more from software-controlled and -distributed memory hierarchies.

Different layers of the hierarchy require different access modes and internal partitionings. The use of page, burst, and interleaving modes, and the physical partitioning into banks, subarrays, and divided-words and bitlines generally must be optimized per layer. Increasingly dominant, leakage power constraints also lead to more heterogeneous memory hierarchies.

Embedded memory challenges. At the circuit level, scaling issues such as amplifier sense margins for SRAM and decreased I_{on} drive currents for DRAM present two clear challenges. Smaller feature sizes allow integration of more devices into a single product, which leads to increased parametric yield loss with respect to both noise margins and leakage power. Future circuit topologies and design methodologies will need to address these issues.

Another challenge, error tolerance, becomes severe with process scaling and aggressive layout densities. Embedded memory's soft-error rate increases with diminishing feature sizes and affects both embedded SRAM and DRAM. Moving bits in nonvolatile memory can also suffer upsets. Particularly for highly reliable applications such as those in the automotive sector, error correction will remain a requirement that entails tradeoffs of yield and reliability against access time, power, and process integration. Finally, cost-effective manufacturing test and BIST is a critical SoC requirement for both large and heterogeneous memory arrays.

DESIGN

Previous ITRS editions documented a *design productivity gap* in which the number of available transistors grows faster than the ability to meaningfully design them. Yet, investment in process technology has far outstripped investment in design technology. Fortunately, progress in design technology continues: The 2003 ITRS estimates that the design cost of the low-power SoC PDA is approximately $20 million in 2003, versus $630 million had design technology improvements since 1993 not been realized. However, embedded software, manufacturing test, and design verification continue to escalate into crises, even as traditional complexity challenges persist. In the 2003 ITRS, the design chapter includes new material on AMS-specific design technology, as well as the future evolution of design processes.

Facets of the complexity challenge

As our January 2002 ITRS article described (*Computer*, pp. 48-49), design technologists are caught between two basic challenges. *Silicon complexity* refers to the emergence of nanometer-scale variability and physical effects that place long-standing paradigms at risk, while *system complexity* refers to the exponentially increasing transistor counts that are enabled by smaller feature sizes and spurred by consumer demand for increased functionality, lower cost, and shorter time to market. Together, these challenges imply exponentially increasing design process complexity. New paradigms are needed in every area of design technology—design process, system-level design, logical-physical-circuit design, design verification, and design for testability—as we move to the 65-nm technology node and beyond. Examples from the 2003 ITRS include:

Embedded software, manufacturing test, and design verification continue to escalate into crises, even as traditional complexity challenges persist.

- *Communications-centric design.* As it becomes impossible to move signals across a large die within one clock cycle, or to run control and dataflow processes at the same clock rate, design will likely shift to asynchronous or globally asynchronous and locally synchronous styles. A new focus on on-chip communication architectures and protocols will usher in an era of communications-centric design.
- *Robustness.* Network-oriented paradigms, as well as yield and synchronization issues, suggest that future SoC design will contend with potentially lossy communication and node failures, placing a new focus on robustness. For example, implementing a completely specified function in an inherently imperfect fabric demands new approaches to design, mapping, and verification.
- *Scalability.* Tools that operate at the lowest abstraction levels—such as logical and physical design and verification tools—face instance complexities that at least double with each technology node. Scalability requires new ways to manage data, search solution spaces, and map optimizations onto distributed and parallel computational resources. Construct-by-correction methodologies and the rise of reuse demand tool runtimes that are proportional not to the size of their inputs, but rather to the size of *changes* from the previous version of the inputs.
- *Cost optimization.* Multitechnology integration in SoC or SiP brings analog and mixed-signal into the mainstream, as well as codesign of die, package, and substrate levels. System cost optimization must span many degrees of freedom: multiple-die and stacked-die options, package and board interconnects, IP reuse across process generations, and the use of reprogrammable blocks.

AMS scaling and migration faces many challenges, including decreasing supply voltages and increasing relative parametric variations.

- *Cooptimization*. Implementation tools create logic and timing structures concurrently with constraint budgets and spatial embedding. For example, today's register transfer level (RTL) floorplanning and global interconnect planning tools define repeater insertion and pipelining, along with detailed layout of global signals. As logical-physical and layout-clock-test synthesis unifications continue, a near-term goal is cooptimization of timing structure, signaling strategies, logic optimization, placement, and routing in a single environment.

Analog/mixed-signal challenges

Analog and RF circuits differ from digital circuits in that they lack quantized information represented in a defined range of voltage levels for the two states of a bit and during a defined discrete time interval. Rather, analog and RF circuits process continuous signals to a much higher degree of precision or smaller tolerance both in time and amplitude. Therefore, factors such as linearity, noise, parasitic elements, and electrical nonuniformity directly cause distortion and noise in analog or RF signals.

Speed issues, or simply that a signal-recovery circuit produces more noise and distortion than it prevents, make these factors much more challenging and less straightforward in the analog domain. AMS scaling and migration faces many challenges, including decreasing supply voltages; increasing relative parametric variations; increasing numbers of analog transistors per chip; increasing signal, clock, and intrinsic device speeds; increasing leakage and crosstalk in SoC integration; and a shortage of design skills and automation.

In system-level design, the critical AMS challenges are analog circuit nonscalability and analog behavioral modeling and synthesis. Automated analog circuit synthesis and optimization is needed, along with language-level modeling methodologies that permit analysis of overall system function and interfaces with other integrated technologies. Issues include coping with vastly varying time scales in simulation, creating heterogeneous test benches and ensuring coverage, achieving systematic top-down constraint propagation, and mixing functional and structural representations.

Analog synthesis provides the key challenge in logical, physical, and circuit design. Scalable SoC design requires eliminating the analog design bottleneck and leveraging reusable, retargetable analog IP generators. More general techniques can augment today's specialized automatic circuit syntheses for particular circuit classes. Syntheses must also handle future regimes of increased manufacturing variability using, for example, hybrid analog-digital compensation for device mismatch.

In design verification, AMS circuits require checking to specification, not structure. New verification solutions must include statistical techniques, better compact models that speed up simulation while increasing accuracy, and new acceptance criteria. AMS designs also force the issue of performing hybrid-systems verification into the near term. This creates an immediate challenge to improve the current ad hoc approaches and find a path toward more powerful techniques.

Design process trends

Four major trends govern future leading-edge chip design processes and their supporting design system structures:

- *Tight coupling*. Design processes are evolving into collections of modular applications that operate concurrently and share design data in memory. In modern methodologies, optimization loops can no longer contain slow file accesses, and the plethora of design issues requires simultaneous optimization of multiple criteria. Further advances will be needed to avoid noise problems, minimize power dissipation, and ensure manufacturability.
- *Design for manufacture*. More intelligent data preparation requires new characterizations of manufacturing process and cost tradeoffs. While inspection and repair form the largest component of mask-making cost and delay, manufacturers perform the process without insight into design intent, wasting effort in satisfying identical tolerances for every shape. We need a standard framework to communicate design data criticality to the mask-making process and to feed manufacturing complexity back to the design process.
- *Increasing abstraction levels*. Most design today takes place at the gate level for greater productivity and at the RTL to specify design in a modern flow. Continued designer productivity improvements require an emerging system-level of design, well above RTL. Higher abstraction levels allow verification to discover problems much earlier in the design process, which reduces time to market and lowers costs.

- *Increasing automation levels*. Using new models to specify design intent leads to opportunities for introducing other tools, such as synthesis. This trend replaces designer guidance with constraint-driven optimization to reduce the number of iterations in later process steps. In future technology nodes, the system-level specification must include both software and hardware and become the controlling representation for constraint-driven implementation.

Detailed implications of these precepts are given in the context of a new ITRS canonical design flow, which serves as a framework for specification of future design technology innovations. For example, in system-level design the simplification of mixed hardware and software system specification, verification, and implementation requires a new level of abstraction above the familiar RTL. This necessitates the following advances:

- *Reuse-based design*. Reusable, high-level functional IP blocks offer the potential for productivity gains estimated to be at least 200 percent. Preverification and reusable tests reduce design complexity, and libraries of reusable software modules can speed embedded software development.
- *Platform-based design*. An extension of core-based design creates highly reusable groups of cores to form a complete hardware platform, further simplifying the SoC design process.
- *System-level verification*. Raising the abstraction level will require a single notation for system-level design. Several years of experimentation with C, C++, and Java variants has led to the recent emergence of SystemC as an option for building interoperable system models of hardware and software for simulation.
- *Microarchitecture synthesis*. Although system synthesis is extremely difficult, progress will likely start with automatic creation of an effective RTL specification from a slightly higher-level representation of the hardware in microarchitecture-specification form.
- *Hardware-software cosynthesis*. Achieving the best overall solution will require the ability to concurrently synthesize a hardware and software implementation. By 2007, we anticipate that an automated step will replace the manual process of mapping a behavioral specification to a software program and a hardware microarchitecture.

In general, solutions to silicon complexity challenges will entail restrictions on design rules as well as continued improvement in analyses. Potential solutions to system complexity challenges will strive for increased capacity, use of hierarchical methods, and a higher abstraction level of design. Joining these two trends is the evolution of the design process and design system architecture itself, as depicted in Figure 4. Tool integrations must minimize unproductive data translation time and data redundancy and support synthesis-analysis tool architectures that execute concurrently on distributed or SMP platforms. Some specific predictions for the future follow.

In general, solutions to silicon complexity challenges will entail restrictions on design rules and continued improvement in analyses.

Designer productivity will improve through multilevel automation, increased reuse, and freedom *from* choice. To systematically advance design technology in this direction, design must be treated as science, not art, with designers adopting a measure-to-improve mindset. Eventually, predictive models of application- or driver-specific silicon implementation will be needed, while complementary innovations should measure and improve design technology productivity.

Approaches to reduce time to market will include use of reprogrammable and structured-ASIC fabrics, reuse of predesigned cores and platform architectures, and pervasive automation. New tools will be required to support a find-and-try style of reuse-centric design-space exploration and design optimization. Development and formalization of design rules that, if followed, assure reusability, will require associated design and analysis software.

In the long term, building on the *design for manufacturability* trend, we will need restricted design rules, as well as the integrated design and manufacturability optimizations indicated in Figure 4. Even today, the focus on yield-driven layout makes manufacturability a standard design criterion.

TEST

The dramatic increase in SoC and SiP designs, largely targeted to commodity consumer applications, has consequently increased pressure to reduce the cost of test for mixed-technology designs. These designs break the traditional barriers between digital, analog, RF, and mixed-signal test equipment capability requirements, resulting in a trend toward highly configurable, one-platform-fits-all test solutions. The first generation of this equipment combined leading-edge technology from all segments, with the consequence of increasing test cost due to the high capital cost of this

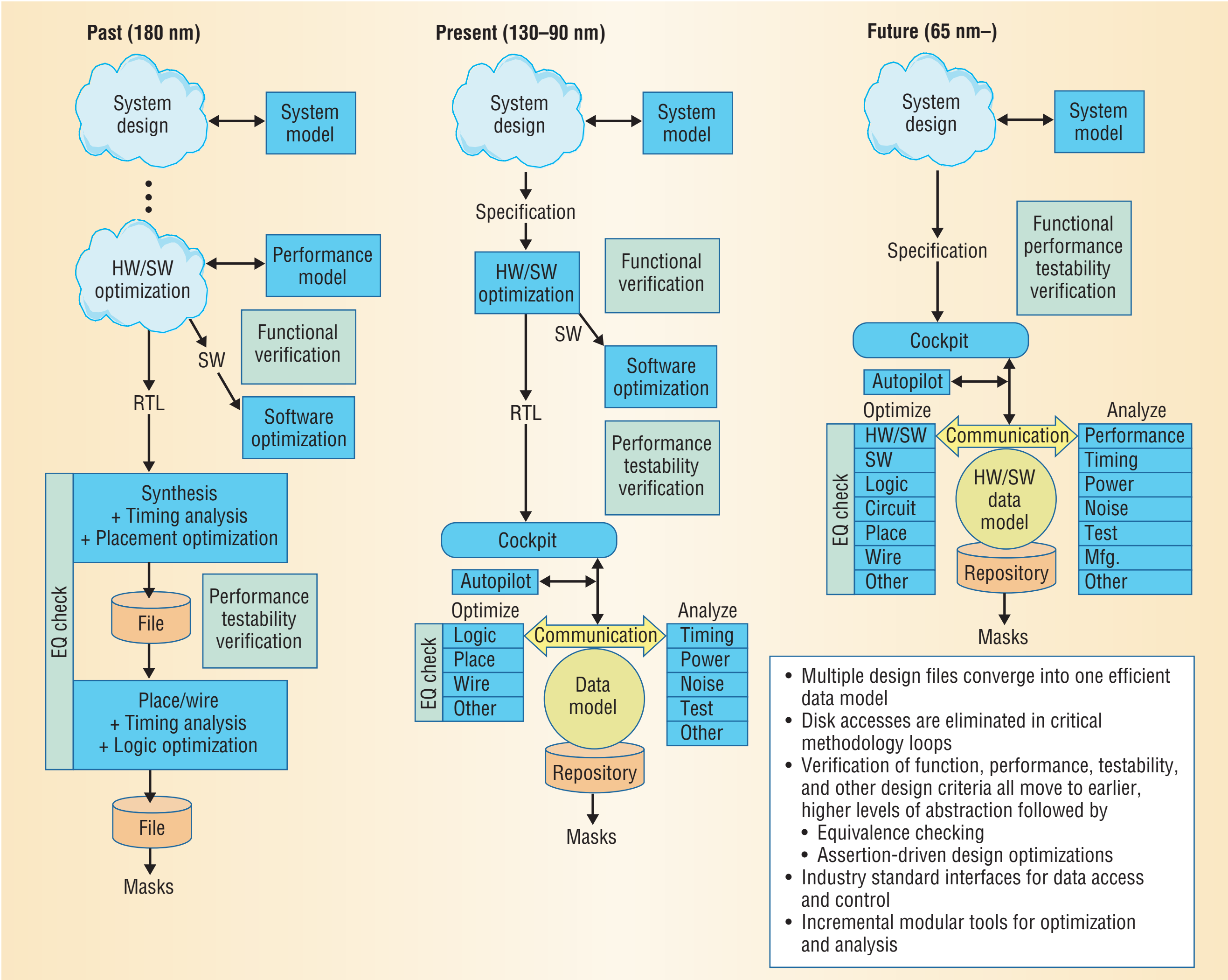


Figure 4. Required evolution of design system architecture.

approach. Low-cost-equipment solutions targeting DFT-enabled devices do not scale into the mixed-technology space today. The next logical step is to increase test system configurability and flexibility to achieve a more appropriate cost performance point—which is leading to a fundamental shift in test equipment architecture.

This revision of the Roadmap finds the test-equipment industry at the beginning of a significant shift from traditional test-architectures to *universal-slot* architectures with high levels of test-instrument encapsulation and modularity. The continued evolution of technology and increased integration level of design components has enabled this shift. In many cases, a single FPGA or ASIC can take the place of entire electronics subsystems.

Encapsulation and modularity have led to the open-architecture concept: the ability to mix and match test instruments from multiple suppliers into a single tester hardware and software environment. This concept presents significant business model challenges to the test equipment industry, which today is based on full vertical integration and proprietary platforms. However, an open-architecture approach offers several potential advantages. It would

- focus research and development investment, both dollars and effort, on the test instrument itself rather than test infrastructure;
- base differentiation on test capability rather than platform;
- focus supplier efforts on developing solutions within their particular core competency, reducing cost and speeding time to market;
- reduce the investment in reengineering infrastructural elements; and
- eliminate the need for each supplier to be everything to everyone.

Ultimately, the industry will drive this approach's success. No one can dispute, however, that the next several years represent a turning point for many test-solution suppliers. Regardless of the open-architecture approach's success, emerging platforms offer a new level of capability, flexibility, and longevity that will significantly impact the industry and test deployments. Quite possibly, the next significant equipment selection decision could identify the test platform deployed for *all* products through the next decade.

While for many years the cost of the tester has overwhelmed all other parameters—with the single exception of throughput—significant progress has been made to address this aspect of the overall device-test cost equation. Focus is shifting to address the numerous secondary costs that now become large contributors. This analysis must encompass all aspects of test cost, including the design nonrecurrent engineering associated with design for test, handler or prober equipment, device interface hardware, and facilities cost, among many other factors.

SoC test

SoC test is dependent on a highly structured DFT methodology to enable observability and controllability of individual cores. Increasingly, SoC design will rely on a database of preexisting IP cores that encapsulate the design itself, interfaces to other blocks, and test.

A fundamental challenge of SoC test is the need to combine test requirements from multiple sources with differing testability approaches and methods. Opportunities exist to define standards for test to conform to a hierarchical methodology; these standards are most easily imposed on internally designed cores. However, when IP is purchased or licensed from a third party, it is typically the test methodology that must adapt. Many electronic design automation (EDA) tools already leverage a standard format for logic designs. This standard must be extended to other core types, such as analog circuits.

Structured use of IP core wrappers and test access mechanisms must be developed for testing of individual cores within a SoC. These should be developed carefully to enable functional, at-speed parametric and interconnect testing. Further, these methods should be standardized with the interface language for interoperability of EDA tools. One such effort is the Core Test Language development within the IEEE P1500 standard. The high complexity of SoC design creates design-and-test development productivity and test-quality challenges. EDA tools and data interchange standards must be developed to aid management of this complexity.

High-frequency I/O

High-speed serial interfaces have been used in the communications market segment for many years. While the communications market is expected to maintain a significant frequency lead, penetration of high-speed serial protocols into the microprocessor, ASIC, and SoC markets in the form of multilane buses has accelerated dramatically. This trend brings a complex test problem, previously limited to the high-speed networking environment, into the mainstream.

Key lessons from this market segment indicate the need to execute extensive testing such as jitter tolerance and jitter transfer on these interfaces. Such testing is done today in the analog domain through a rack-and-stack or mixed-signal tester approach. These solutions impose significant manufacturing cost considerations due to test time and equipment capital cost, and they support a relatively limited number of high-speed serial ports on a single device. As these interfaces proliferate to many ports on a single device, the traditional analog test approach will fail due to the scalability of analog instrumentation.

As the frequency of these interfaces and the number of interfaces on a single design continue to increase, alternative equipment solutions and test methods will need to be developed to enable cost-effective, high-coverage engineering and manufacturing test.

The high complexity of SoC design creates design-and-test development-productivity and test-quality challenges.

Reliability screens

The test process is responsible for the screening of manufacturing defects that affect device functionality, performance, and reliability to reduce customer perceived defects per million. A portion of the test flow is dedicated to the acceleration of latent defects that do not appear as test failures but would manifest as longer-term reliability failures. Traditional techniques for screening of reliability failures include I_{DDQ}, burn-in, and voltage stress.

The effectiveness of these methods is challenged by the continued device scaling with each successive process generation. Increases in device quiescent current in the off state is raising the level of background current to the milliampere and, in some cases, ampere range. These levels of background current increase the difficulty of identifying microampere to milliampere level I_{DDQ} fault currents. Continued extension of techniques such as delta-I_{DDQ} that have enabled extensions into current process generations is uncertain.

At the same time, the effectiveness of voltage and temperature acceleration methodologies used by burn-in and voltage stress is declining due to the reduced margin between operational and overstress conditions. Costs associated with burn-in techniques continue to rise and, in some cases, now dominate manufacturing costs for high-power products.

The increasing cost and declining effectiveness of current techniques for latent defect acceleration combine to create one of the most critical challenges facing the industry for future process generations. Extensions to current techniques may prove adequate for the next several years, but fundamental research in the development of new methodologies is required.

Automated test program generation

Correct-by-construction test programs and patterns have long eluded test development teams. Increasing design complexity and demands on team productivity require significant improvement in test program generation automation to limit or reduce time-to-market impact. The EDA industry has developed and deployed many tools intended to aid the entire process of test development, content creation, and equipment translation. However, device makers seldom have a homogeneous environment provided by a single EDA supplier, and the general lack of interoperability standards among tools creates significant challenges requiring effort by the device maker to enable automation. Cooperation among EDA and test equipment suppliers is increasing, and a focus on tool interface and interoperability standards is growing. The challenges increase with every process generation and related growth in design integration.

Achieving full automation of test-program-generation will require increased standardization of the test-equipment software environment itself. Historically, each equipment supplier has taken a holistic and proprietary approach to definition and development of its specific software environment. Similar to EDA, most device makers do not have a homogeneous test environment of equipment provided by a single supplier. Increased standardization of the test-equipment software environment where appropriate would lower the entry barrier for suppliers as well as simplify the porting of test content between platforms as required by the device maker.

Today's environment of platform-unique supplier software solutions and homegrown tools for equipment programming, automation, and customization will drive unacceptable growth in test- development engineering and factory-integration efforts. Automation of common tasks and decreasing test-platform integration time demand a focus on standards to enable more efficient use of resources in line with shrinking product development life cycles. New tool development must comprehend the end use to ensure that the resulting effort is indeed reduced over existing methods and that pre- and postprocessing of data do not eliminate throughput gains.

Previous ITRS editions documented a design productivity gap, with the number of available transistors growing faster than the ability to meaningfully design them. Yet, investment in process technology has by far dominated investment in design technology.

The good news is that enabling progress in design technology continues. The bad news is that software can account for 80 percent of an embedded system's development cost, test cost has increased exponentially relative to manufacturing cost, and verification engineers outnumber design engineers on microprocessor project teams. Overcoming many existing design technology gaps will require a concerted effort by the entire industry. ■

Acknowledgments

We acknowledge the efforts of the many individuals who contributed to making the 2003 edition of *The International Technology Roadmap for Semiconductors* a successful endeavor.

Don Edenfeld *is responsible for test equipment definition and selection at Intel. He received an MS in electrical engineering from the University of Virginia. Edenfeld is a member of the IEEE Computer Society. Contact him at donald.e.edenfeld@intel.com.*

Andrew B. Kahng *is a professor of computer science engineering and electrical and computer engineering at the University of California, San Diego. He received a PhD in computer science from the University of California, San Diego. He is a member of the ACM, the IEEE, and SPIE. Contact him at abk@ucsd.edu.*

Mike Rodgers *is responsible for test technology integration at Intel. He received a BS in electrical engineering from the University of Illinois. Rodgers is a member of the IEEE Computer Society. Contact him at michael.j.rodgers@intel.com.*

Yervant Zorian *is the vice president and chief scientist of Virage Logic. He received a PhD in electrical engineering from McGill University. Zorian is a Fellow of the IEEE and a member of the IEEE Computer Society. Contact him at zorian@viragelogic.com.*

NASA's Mission Reliable

Like industrial development organizations, the US space agency struggles with the challenge of creating reliable software. NASA's deep space community is attacking its software crisis via two complementary approaches—one stressing the power of engineering discipline, the other the potential of automated code generation and verification.

Patrick Regan
New Jersey Network Public Television and Radio

Scott Hamilton
Computer

Both predictable and unpredictable hazards await the spacecraft, robots, and scientific instruments that humans dispatch to explore our solar system. The toughest hazard to harden a space probe, orbiter, or rover against may be one of the most predictable: the known presence of unknown bugs in even the most rigorously tested software.

As this article goes to press, spacecraft from Europe, Japan, and the United States are converging on Mars, and some sophisticated software systems are approaching their ultimate test. With all eyes turning toward the red planet, some observers will recall the software-related failures of the Mars Polar Lander and Mars Climate Orbiter missions in 1999—and pay special attention to the performance of NASA's Mars Exploration Rovers, twin robotic science platforms about the size of a golf cart.

Meanwhile, far beyond Mars, the Cassini spacecraft cruises toward a rendezvous in March with Saturn. The Voyagers launched in the 1970s report back from the boundary region between our sun's sphere of influence and interstellar space. Orbiting observatories take the measure of the universe. And back in California, the next billion-dollar project is already in development at Jet Propulsion Laboratory, managed for NASA by Caltech. Scheduled for launch in 2009, the Mars Science Laboratory mission aims to land a rover as big as a car for a tour lasting more than a year. Many more missions in the $200 million range are in flight, in development, or on the drawing boards.

HIGHER STAKES, GREATER RISKS

On increasingly frequent and ambitious missions, these remote laboratories radio back pictures of other worlds; measure radiation, particles, and fields in the interplanetary environment; probe the chemistry and dynamics of planetary bodies; and, above all, equip scientists to test theories, answering long-standing questions and raising new ones.

NASA and other space agencies rely on ever larger, more complex software systems to do more challenging science. Because of the nature of their missions and the distance from ground control—with communications requiring from tens of minutes to hours of "round-trip light time"—spacecraft and rover systems demand increasing autonomy that introduces new dimensions of complexity.

This trend raises the stakes and increases the risks that a problem older than the expression "software engineering" poses: It's not getting any easier to produce reliable software.

Within the earthbound labs responsible for designing and flying NASA's space missions, computer scientists and software engineers have been

Mars Exploration Rover—artist's conception. Courtesy NASA.

mobilized as never before to attack this problem. It may be that the perpetual "software crisis," first declared at NATO's 1968 conference on software engineering, will humble NASA as it has other organizations that have tried to address it. On the other hand, the urgency and unique resources that NASA brings to the problem might produce breakthrough solutions.

Despite conspicuous and even tragic failures, the US space agency has engineered the seemingly impossible and made it look routine. If NASA can solve its own software crisis, it may help boost software engineering and commercial development to a higher stage of maturity.

FROM GROUND CONTROL TO AUTONOMOUS SYSTEMS

Traditional mission control and planning have been done on the ground, with humans in the loop generating a time-based command sequence that tells the spacecraft what to do and when. Notwithstanding the occasional surprise, this approach has been quite successful; however, with the many small missions on the books, traditional ground operations costs have become prohibitive.

Past spacecraft systems included limited autonomy programs such as attitude control, fault protection, and orbit insertion, entry, and landing that were tried and tested over decades and hardcoded into the systems. However, new science mission requirements and more frequent missions have necessitated efforts to generalize autonomous spacecraft control across multiple missions.

Traditional spacecraft control systems contain a fault-protection subsystem that monitors the system's behavior and "safes" the spacecraft by pointing its solar panels toward the sun, reporting a possible malfunction to ground control, shutting down unnecessary systems, and waiting for instructions from Earth. Although such fault-protection mechanisms have had a long history of success, safing a spacecraft at critical junctures such as orbit insertion is obviously a recipe for disaster. Perhaps more important, scientists have come to expect exponentially more from these missions.

Safing a spacecraft or rover for minor faults that do not put it at risk and waiting hours for instructions can greatly reduce the amount of scientific data that can be obtained and sent back to Earth—indeed, such behaviors could preclude opportunistic science such as a comet fly-by.

Although it is a hostile environment for physical hardware, deep space is a relatively benign environment for mission control because extended periods of time are available to plan nonemergency maneuvers. A rover on the Martian surface, however, needs extensive autonomy to accomplish complex missions in which it may have only seconds to avoid toppling into a ditch.

Autonomous control poses some of NASA's trickiest problems and stimulates some of its most advanced research. Yet the agency has more mundane issues to deal with as well.

SOFTWARE QUALITY INITIATIVE

During a career that began with semiconductor engineering for Mariner missions to Mars and Venus in the 1960s, Tom Gavin, the senior member of the Jet Propulsion Laboratory's flight community, has seen software rise to prominence not only as an enabling technology but also as a major risk and cost factor. Gavin has made it his mission to improve JPL's software processes, better integrate software production with systems engineering, and encourage research that focuses on a bulging portfolio of flight projects.

In the Mars Exploration Rover, entry, descent, and landing are software-controlled. Gavin says, "People see the mechanical system of this vehicle, parachutes coming out, airbags coming out, and bouncing on the ground. But that is a software-driven system, and a lot of the project risk is in how robust that software is." Software accounts for a large share of project cost too, in coding, independent verification and validation, and testing—four testbeds for the Mars Exploration Rover alone, running 24 hours a day in three shifts, with deadlines dictated by the movements of the planets.

Missing a deadline that celestial mechanics impose could trigger a long, costly delay—22 months for a mission to Mars, 19 to Venus, or 13 to Jupiter. Keeping such a deadline can exact a different kind of penalty. Gavin notes that the software for PathFinder, which landed successfully on Mars in 1997, was largely undocumented.

In an effort he says is still in its infancy, Gavin has charged the JPL software community with developing a kind of discipline long established—and recently codified—on the hardware side. A few years ago, engineers captured 40 years of hardware experience in three books on flight project practices, design principles, and mission assurance. Now the lab is investing $4 million a year in a software quality initiative aimed at getting developers to codify and buy into a similar set of software practices. A survey currently under way will help define the baseline for this effort and for forward-looking research.

More effective matchmaking between practitioners' needs and applicable research is an integral part of the software quality initiative. Gavin is optimistic that the initiative will succeed by involving the practitioners in documenting their practices.

RESEARCH GOES DEEP

Meanwhile, exploratory research—at NASA facilities including Ames, Dryden, Glenn, Goddard, Johnson, Langley, and Marshall as well as JPL—aims at ensuring the reliability of software for future missions. The unique challenges of deep space have brought an infusion of funding to software reliability research at JPL, with close ties to affiliated groups at Ames. Researchers are applying advanced technology for both code synthesis and code verification of software that runs the gamut from artificial intelligence-based autonomy software—which uses methods such as rapid propositional deduction and adaptive neural nets to uncover faults or execute plans—to large systems with more conventional kinds of complexity.

JPL established a Laboratory for Reliable Software in 2003, with model-checking guru Gerard Holzmann as its founding member. Mission Data System (MDS) is a bigger enterprise within JPL's Interplanetary Network Directorate that has similar motivation.

In 1998, just as it was about to launch six independently designed missions in the span of six months, JPL became acutely aware of the need to make more effective use of its software engineering resources and to reuse software common to all missions. At that time, avionics engineer Robert Rasmussen championed MDS as a multimission architectural framework that could unify software and systems engineering, from the conception of a mission through development and flight.[1]

When NASA people use the term MDS, they could be referring to the architecture, the idea, the million lines of code that currently embody it, development of mission software for the Mars Science Laboratory, a set of development processes, or related research. In any case, MDS offers a multimission framework for building, testing, and reusing software that will fly in spacecraft, land in rovers, and operate here on Earth. This is a striking departure from established practice, in which highly compartmentalized development efforts have produced essentially one-off software systems for each space mission.

Between them, Holzmann's research and the MDS project don't begin to encompass all the resources being brought to bear on ensuring that NASA software will reliably do what needs to be done. But they do represent the range of approaches NASA is pursuing, and they illustrate a philosophical tension that both polarizes and helps to energize the whole endeavor.

Programming is a human effort, and people make mistakes.

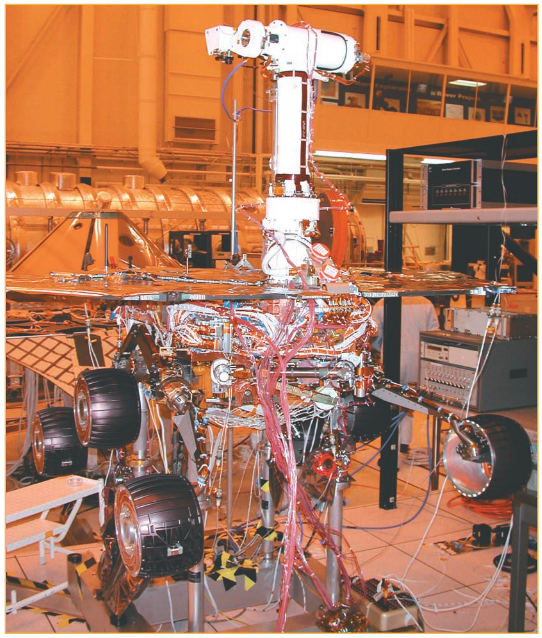
TO ERR IS HUMAN

The basic observations underlying Holzmann's work are that programming is a human effort and that people make mistakes. He cites estimates that around 50 software defects remain in 1,000 lines of newly written uncommented code, and around 10 remain in code that's been thoroughly tested. Extreme measures of additional testing can push the number of residual defects per 1,000 lines down toward one, but that still adds up to a lot of bugs in a large system, and few products ever reach that level.

Statistics for novice and experienced programmers are essentially the same. "An experienced programmer tends to make just as many mistakes," Holzmann says, "but they are much more complex, they're subtle, they require a lot of deep thought. If he makes a mistake, it is usually much more difficult to find."

Testers are human too, and following this line of reasoning leads to the conclusion that human-driven testing processes are, like programming practices, inherently flawed. For Holzmann, this alone makes automating alternative methods an imperative. With a combination of optimism and patience, Holzmann is determined to convince his colleagues of the power of automated model checking techniques as an alternative to the conventional approach to software testing.

Methods like model checking, static analysis, and runtime analysis offer the ability to do exhaustive



Mars Exploration Rover at the JPL spacecraft assembly facility. Courtesy NASA.

exploration of source code, testing even the most unlikely scenarios. Conventional testing, a time-consuming operation for which the allotment of time tends to get squeezed as the delivery or launch date approaches, may root out the most likely failure scenarios and more subtle vulnerabilities; however, it can never offer the possibility of finding all the bugs.

Furthermore, testing methods lose ground all the time. "We build systems more complex than we can understand and more complex than we can check," Holzmann asserts. "Conventional testing processes, which were designed for deterministic, sequentially operating systems, haven't really changed in 30 years. Yet systems now are multithreaded, which makes them nondeterministic because of process and thread scheduling—all the interleaving is different in every run, events occur at unpredictable moments—and the test methods are simply not designed to handle that."

In contrast, he says, researchers are beginning to win the battle of software analysis. Winning, in this context, means that formerly unassailable problems show signs of yielding to improved algorithms and the steady march of processing power. By Holzmann's calculation, a small, automated test he devised in 1980 would have kept a computer busy for seven days, if he'd had one powerful enough to run it at all. In 2000, with a thousand times more memory and processing speed readily available, the same test could be done in seven seconds. Today, it would take two seconds; in two more years, it should take just one.

By the late 1970s, some believed formal methods could be automated and eventually scaled up. By the late 1980s, Holzmann had developed Spin, probably the best known and most widely used model checker in the world.[2] In the late 1990s, he and his colleagues at Bell Labs used Spin and associated technology for the first time in the development of a real product when they verified call-processing code for an Internet Protocol-based telephone switch.

Although the chunk of code they checked was relatively small—10,000 lines out of roughly 25 million—it was functionally the heart of the system and a nightmare of potentially fatal interactions between concurrent processes. This application proved a few important claims for the technology, including that it could extract models from source code automatically. The creation of a "test harness" to extract models from C code was a one-time investment, reusable throughout the development process.

JPL researchers who were using Spin to verify code for the Cassini mission drew Holzmann into NASA's orbit in the 1990s. Within weeks of his move to California, he had helped to build a test harness for verifying code already flying on the Mars Exploration Rovers and was seeking ways to contribute to the Mars Science Laboratory. At JPL, Holzmann is working to build a core group of theorists, system builders, and "energizers" that will interact with a virtual team of a few hundred to show that these techniques work.

SPINNING IN SPACE

NASA researchers, including Ames' Klaus Havelund and Thomas Pressburger, were early and insightful experimenters with the Spin model checker. Together, they developed their own innovative verification and testing technology, Java PathFinder 1, which translated high-level source code to Promela, Spin's formal language.[3] Subsequently, for Java PathFinder 2, Ames' Willem Visser and Havelund developed a new model checker, in which a Java virtual machine interprets the source code.[4] Work is under way to integrate Java PathFinder 2's model-checking technology with Java PathExplorer, a separately developed tool for runtime analysis and monitoring.[5]

This effort began with Deep Space 1, a technology validation flight. Part of the mission called for the Remote Agent software to take control of the spacecraft for two days. Havelund and Ames' John Penix ran part of the Plan Runner, a critical Remote Agent code component, through a Spin cycle as a verification experiment. This check automatically found five bugs, concurrency errors that were corrected before flight.

Despite that check and 800 hours of preflight testing, Remote Agent hit a deadlock six hours after it was activated. A team assembled to analyze the system's 12,000 lines of Lisp found the bug within five hours—the same kind of bug that Havelund and Penix had found in their earlier work—in a module that had been reworked after their verification experiment. Interestingly, the team opted not to fix this particular bug because the chances of its occurring again were extremely small. The bug did not recur—whereas a bug fix had a greater chance of causing problems—again, showing how rare these bugs are to begin with and why they are so hard to find with conventional testing.

That episode highlighted both model checking's potential and its current limitations. "Particularly for concurrent software where there is a lot of interleaving," says senior research scientist Michael Lowry, "model checking can be much better than testing."

According to Lowry, the increasing number of threads in some mission software is not the only trend driving up the risk of concurrency errors. Smaller numbers of complicated threads can have the same effect. Beyond that, he says, "If you're interacting with an environment in complicated ways, the whole system becomes concurrent. Think about a rover on Mars. As you move away from time-based sequencing, you have lots of different interleavings between what the environment does and what your software will do. That's exactly where you're going to get a vehicle that's capable of dealing with a rich environment—both for science return and for its own survival—and that's a situation where concurrency occurs even if the software program has only one thread."

Yet another trend highlights the challenge of scaling model checking to deal with large programs. Numbers that Lowry cites as rough guides to this trend are the 30,000 lines of code developed in the 1980s for the Cassini mission, 120,000 lines of code for the mid-1990s development of the Mars PathFinder, and 428,000 lines of code for the Mars Exploration Rover. Others suggest that the software for the Mars Science Laboratory could grow to several million lines of code.

FINDING ALTERNATE PATHS

When Havelund and Penix began hand-translating Deep Space 1 code for Spin to check, they could analyze around 30 lines of code a day. Automating translation in Java PathFinder boosted analysis to around 1,500 lines of code per day, limited mainly by memory. At that point, however, the researchers felt constrained by Spin and Promela. That's when Visser and Havelund developed a model checker that uses a Java virtual machine. According to Visser, this model checker could "analyze all the behaviors, all the paths through the code, because we had control of scheduling, so we could schedule all the possible interleavings." Visser says that creating the first version took only three months, but they have been working for four years to perfect it.

The software for the Mars Science Laboratory could grow to several million lines of code.

The largest program they have analyzed so far is 8,000 lines of prototype rover software with six complicated threads, translated to Java from the original C++. Like Spin, Java PathFinder is expected to become more effective simply by riding the hardware curve and benefiting from improvements in automated translation. Penix is writing a C++ to Java translator so that Java PathFinder can be used on C++ code as well.

These researchers also aim to improve the effectiveness of automated software analysis by coming at it from another direction. Whereas Java PathFinder searches the space of paths through the code to find one that has an error in it, Java PathExplorer focuses on detecting errors in a single path. The key, Havelund says, is using more powerful temporal logic "which can, for example, reason about future and past time logic at the same time, reason about real time, and reason about values."

Java PathExplorer runs the program once, recording the system's state throughout that execution. Working with an execution trace in which all events of interest have been logged, the tool employs one set of algorithms for exact analysis to determine whether or not there is an error in that particular trace. It uses another set of algorithms to infer whether other possible permutations of the trace could contain errors.

Recent enhancements include strengthening the tool's temporal logic and extending its capability to high-level data races. An example of the latter would be a problem in which one thread updates x and y coordinates in a single operation, and another thread reads them separately, one at a time. Thus, if the first thread can update both coordinates between the reading of x and y, a "correct" operation of the program could yield the wrong coordinate pair.

Havelund and Visser plan to use the underlying Java PathFinder and Java PathExplorer technology to generate tests automatically. They also have combined these model checking and runtime analysis tools in a testing environment for K9, a planetary rover.

Mars Science Laboratory—artist's conception. Courtesy NASA.

COMPONENTS IN CONTEXT

In a complementary effort at Ames directed at improving model checking scalability, Dimitra Giannakopoulou and Corina Pasareanu focus on modular or compositional verification. Their approach decomposes the analysis of a program into analysis of its components that might be easier to check than the whole, and it does this in a way that guarantees the properties of the reconstructed whole. These researchers also are experimenting with the K9 rover software.

Given an abstract model of a component's behavior and a required property, Giannakopoulou and Pasareanu have developed techniques for formulating assumptions, which encode the contexts or environments in which the component will satisfy the property. When the component's environment is available, it is checked against the assumption to ensure correctness of the whole. "We split the component behavior into behavior that is controllable and uncontrollable by the environment," says Giannakopoulou, "and get as a result how controllable behavior affects the satisfaction of required properties."

These researchers have developed a framework based on the use of learning algorithms and model checking for formulating assumptions incrementally. When it's not clear what the component's environment will be, the fallback is a twist on runtime analysis—generate a monitor based on the assumption, instrument the program, monitor the environment against the assumption during deployment, and trigger recovery code when the assumption is violated.

Because a lack of design information blocks this approach at the model level, Giannakopoulou and Pasareanu are extending their approach to work directly at the software level. Where design models of a system are not available, they want to apply the same techniques to code—generating assumptions and performing the same kind of "modular reasoning" on the source code. It's an aspiration at this point, but they have some ideas about how to make it work and the collaborations needed to test their ideas.

DYNAMICS OF STATIC ANALYSIS

Source code is in fact the starting point for static analysis, another kind of tool being honed on NASA software. Though formal, this method differs sharply from model checking, and it produces different kinds of results. According to Guillaume Brat, an Ames-based researcher, static analysis can analyze code on the basis of program semantics without constructing a model or running the program. Brat and his collaborator, Arnaud Venet, have been working to optimize static analysis for NASA and to prove its practical potential in software coding, unit testing, and integration.

Static analysis does not help much with concurrency, deadlocks, and data races. It can, however, do a thorough job of rooting out errors in programming style—such as uninitialized variables and pointers, out-of-bounds array access, and invalid arithmetic operations—that could corrupt data or even crash a system at runtime. Almost by definition, such errors are common and plentiful, but static analysis has the potential to weed them out.

This type of analysis produces a kind of mathematical hologram—described as a computable approximation to the set of values arising dynamically at runtime when executing a program—that researchers can view from various angles and probe for details. The method's scientific basis is abstract interpretation, which expresses a program in terms of equations, and lattice theory, which offers techniques for solving the equations. The output essentially is three sets of results: Errors the tool is sure are errors, code that is surely correct, and apparent problems that might be errors or might be something else, such as dead code.

The objective is to compute numerical invariance at every program point. The analysis must guess what values an integer can take for any execution of the program. For every concrete operation in the program, such as addition, there is an abstract operation based not on integers but on a finite representation for those values such as an interval. An interval is an abstraction that represents the values of real variables in a program. For example, the interval 1-7 represents integers 1, 3, 5, 7.

Brat and Venet launched their project by applying the state-of-the-art PolySpace static analysis tool to real NASA code from the space station, Deep Space 1, and Mars PathFinder. The tool worked just well enough to set baseline measurements for scalability and precision—that is, the proportion of definite classifications to warnings of possible problems in the code. Then they created a specialized tool tailored for NASA software as they knew it. They have begun testing their new tool—the C Global Surveyor—on entry, descent, and landing software for the Mars Exploration Rover as well as on software for the rover itself. Along

Mars Science Laboratory—artist's conception. Courtesy NASA.

with other researchers, they have begun exploring ways in which MDS developers might make gains in terms of automated verification by accepting some restrictions on how they program in C++.

TO BE HUMAN IS TO ENGINEER

Kirk Reinholtz, one of the key collaborators working with Holzmann and other proponents of automated approaches, is an eloquent spokesman for a different point of view. Reinholtz, who describes himself as "a software engineer, born and raised," is the chief programmer on the Mars Science Laboratory project.

In explaining how his team is building one million-plus lines of Mission Data System code into a multipurpose system for the 2009 launch, Reinholtz shows a firm faith in the power of human skills, judgment, and processes: "The purpose of that software is essentially to turn everything we've learned over decades of doing embedded, extremely reliable software into more of an engineering discipline than an art."

According to Reinholtz, the genius of MDS lies in "hoisting" issues that experience shows can be real problems in actual missions to the architectural level, "where we have fairly mature processes to observe them, get plenty of eyes on it, do the verification, and so forth." What a contemporary software engineer might tend to hide, he says, MDS makes explicit.

Down to the level of specifying a vocabulary for discussing engineering goals, MDS aims to make it easier for programmers to do the right thing and harder for them to make mistakes.

The approaches that MDS and model checking typify—one more confident in experience and ingenuity, the other more wary of human fallibility—offer solutions that could contribute to the success of the Mars Science Laboratory and other missions. Both also may point toward a way out of what JPL principal software architect Nicolas Rouquette calls "the traditional divorce and death march," in which projects succeed because of the sacrifice of "heroes."

DESIGNING NEXT-GENERATION SYSTEMS

Traditional design methods—including those used at JPL in the past—involve a hierarchical decomposition of a system into subsystems, with each software engineering team providing its own customized solutions and iteratively integrating them at the system level.

Although subsystem decomposition works in well-understood domains such as enterprise systems, it works less well in an environment such as deep space where system resources such as power or memory are limited and the system must interact with an unpredictable physical environment.[1,5] These various subsystems must share these limited resources, but the assumptions that one subsystem design team makes might not hold across all subsystems. For example, it is virtually impossible to abstract an idealized camera in this environment because the physical camera draws power used by other resources, consumes CPU cycles to process data, and could be used as a navigation device to track stars or celestial bodies. In the end, says Daniel Dvorak, MDS deputy architect, using object-oriented methodologies in such an environment leads to a hierarchical subsystem decomposition that is difficult to verify, validate, or reuse.

Drawing on his vast experience with past JPL missions, Robert Rasmussen defined in MDS an abstract architecture for designing next-generation deep-space systems. Such systems interact with the physical world, reacting to and trying to control physical state, and mission controllers on the ground think in similar terms. Therefore, Rasmussen proposed a state-based control architecture that models the interactions between physical states and attempts to control state by applying goals and constraints on state rather than employing a time-based sequence of commands, thus allowing varying degrees of mission autonomy.

Unlike traditional object-oriented subsystem decomposition, which hides or encapsulates system interactions in local program variables, flags, counters, pointers, and if statements, MDS elevates state variables to the top of the architectural hierarchy where they can be seen, understood, and engineered.[1,6] As such, systems engineering becomes state analysis, with a precise and bounded vocabulary for describing all interactions between the elements the project comprises. This vocabulary includes terms such as state variables, estimators, controllers, state value histories, state effects models, measurements, time and state constraints, resources, commands, scenario fragments, and goals.

Just as important, this constrained vocabulary translates to the software engineering domain, providing strictly defined input/output values that either constrain or aid a system developer when implementing that portion of the system: Either the programmer has introduced unnecessary com-

MDS provides a domain-specific framework for a family of applications and their attendant flight, ground, and test platforms.

plexity or a potential error, or the initial systems analysis was wrong. Moreover, all hardware components, their attributes, and the modeled transformations between them can be captured in a state database and queried by mission planners and systems engineers; by software developers and their automated tools; by simulators testing software against undelivered hardware; and, perhaps more important, by onboard software in the case of unexpected system behavior.

Covering the entire engineering discipline and enforcing it from top to bottom, MDS uses a shared vocabulary based on state analysis for requirements capture. It applies this vocabulary to a software development process engineered down to the level of configuration management, at the same time leaving room for new technical approaches.

MDS combines a state-based systems architecture and component-based software architecture to provide a domain-specific framework for a family of applications—whether they are multiple generations of satellites, landers, or rovers—and their attendant flight, ground, and test platforms.[1] The state-based architecture offers a structured process for disciplined analysis that emphasizes model-based design for estimation and control, makes interactions explicit, and exposes complexity. The component architecture provides frameworks and adapter's guides, reusable building blocks in object-oriented design, guides for how to adapt it for concrete tasks, and examples of framework usage.

Flight software is largely embedded, whereas ground software has extensive resources in the form of servers, adequate power, and so forth. But whether it is one millisecond or one day, a communication delay closes the control loop from both the architectural and customer viewpoints. For example, mission controllers view the spacecraft as a point in a two-dimensional plot, whereas the spacecraft itself is an object in three-dimensional space. Then the question becomes, where is it pointing?

Both viewpoints use the same mathematical equations but derive different results due to their different degrees of freedom. In contrast, because MDS focuses on the similarities rather than the differences, developers can use the same mathematical and architectural framework to write both the flight navigation and ground control software. Unlike past missions, JPL will cost together the flight and ground software for the Mars Science Laboratory—a major cultural change.

STATE ANALYSIS

State analysis as embodied in MDS provides a uniform, methodical, and rigorous approach to discovering, characterizing, representing, and documenting a system's states, and modeling their behavior and the relationships among them. Knowledge of the system and its environment is represented over time in state variables, which include such things as

- *dynamics*—vehicle position and attitude, gimbal angles, wheel rotation;
- *environment*—ephemeris, light level, atmospheric profiles, terrain;
- *device status*—configuration, temperature, operating modes, failure modes;
- *parameters*—mass properties, scale factors, biases, alignments, noise levels;
- *resources*—power and energy, propellant, data storage, bandwidth;
- *data product collections*—science data, measurement sets;
- *data management and transport policies*—compression, deletion, transport priority; or
- *externally controlled factors*—spacelink schedule and configuration.

MDS reports, stores, and transports information about the system as histories of state, measurements, and commands.

Systems engineers use state analysis to capture mission objectives in detailed scenarios, keep track of system constraints and operating rules, describe the methods they will use to achieve objectives, and record information about hardware interfaces and operation. Throughout, the common framework elements (vocabulary) unify all aspects of the design process. For example, if the *goal* is to move a rover to a rock, the *state variable* to be controlled is the rover's position relative to the rock. *Measurements* provide evidence for that state—for example, wheel rotations, sun sensor, or stereo camera. For a stereo camera, *measurement models* indicate the distance to terrain features, light level, camera power (on/off), camera health, and so on.

Figure 1 shows the MDS goal-oriented state-based architecture. Given a model of how things work, estimators find "good" explanations for measurement (sensor) and command (actuator) data to estimate state. State variables hold state values, including the degree of uncertainty. To describe state evolution, state timelines combine current and past estimates with future predictions and plans. Together, time-based state information and models of state behavior supply the information needed to operate a system and assess performance.[1]

Operators express their intent in the form of goals declaring what should happen—not how. The operators and planners can elaborate the goals recursively, and even conditionally, into lower-level goals that are coordinated by a controller/scheduler that uses priority as the final arbiter to resolve conflicts. MDS keeps state estimations and state control completely separate to avoid the temptation to warp a state estimation to meet a control objective or the risk of having multiple interpretations for the same data.[1]

Finally, MDS represents actual hardware components in software as hardware adapters that facilitate the delineation of the abstract system model (including time) by translating raw input/output data and measurement and command models into abstract declarations about state. In addition, software-based hardware adapters can augment system hardware with supplemental behaviors such as sampling, I/O, sequencing and synchronization, time and metadata tagging, data buffering and routing, data format translation, error checking, and data preprocessing and compression. More important still, MDS isolates state frameworks from platform-specific interfaces and supports real, simulated, or abstract hardware in real or virtual time.

MISSION POSSIBLE?

Elevating state to a first-class entity and controlling state through goals that constrain it over time will guarantee an ambitious agenda for MDS. Because MDS makes both state and the models that describe it explicit, goals are self-checking by nature, prescribing possible inputs and outputs. Goals also provide hooks for model checkers, code instrumentation, and verification and validation tools to increase reliability.

A more intriguing possibility proposed by JPL's Rouquette would use automatic code generation for the vast majority of spacecraft system software. Rouquette views current model checking techniques as more suitable for niche problems, algorithms, and carefully chosen pieces of code where developers know what they're looking for—not for abstracted sections of code from large-scale systems in which complexity arises from the interactions between the different parts. Instead, Rouquette is banking on code generation and transformation techniques already used successfully in Deep Space 1.

Rouquette used the MathWorks' Stateflow toolbox and some homegrown tools to generate 90 percent of the code for the DS1 fault-protection subsystem from 60 or so state machines. Since building such models can take several months and they can be revised 50 or more times, automatically generating efficient C code with a small code footprint was a huge win—allowing talented engineers to build the system via models, as in civil engineering, rather than waiting for programmers to code, verify, and, more often than not, recode the particular system repeatedly. However, this solves only half the problem.

Over the past decade, code generators have indeed become more clever, progressing from "dumb" programming tools that do simple string manipulation to something more akin to a compiler that can parse a specification to gain a notion of valid input and then perform internal processing to produce some output. While these more sophisticated tools now do a very good job on the input side, says Rouquette, there is far less assurance on the output side, especially when producing output in languages whose meager semantics are difficult to prove formally and hard for model checkers to check.

With its state-based models, MDS offers a more formal expression of valid input and output. Smarter languages with higher levels of abstraction will facilitate model checking and make it possible to apply modern compiler technology and algorithms such as pattern matching, tree recognition, and algebraic formalisms for operations on trees to code generators themselves.

Rouquette's grand vision is to generate 95 percent of the Mars Science Laboratory code automatically, leaving only a small runtime that will be coded manually and will require extensive model checking and testing for validation. Space is a

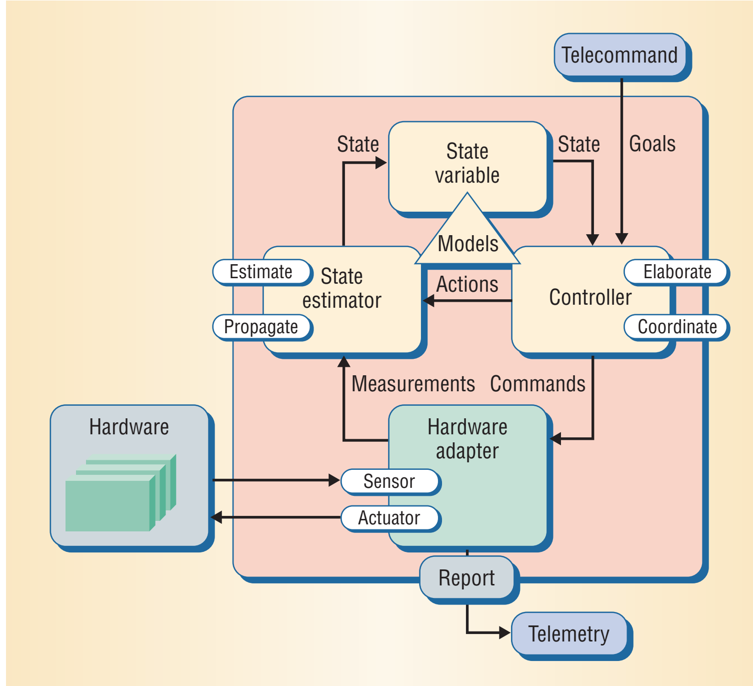


Figure 1. MDS architecture. A goal is a constraint on the value of a state variable over a time interval. State variables hold state values, including degree of uncertainty. Models express mission-specific relations among states, commands, and measurements. Estimators interpret measurement and command evidence to estimate state. Controllers issue commands, striving to achieve goals. Hardware proxies provide access to hardware buses, devices, and instruments.

largely unknown environment, says Rouquette, and the only way to ensure mission success is by having the many bright people involved use their knowledge, imagination, and time to engineer systems instead of writing code. “Analyzing rocks on Mars sets our expectations way too low, and there are many other nice places to go. But we have to make these things dirt cheap and a sure thing.”

By exploring new technologies and approaches to develop provably reliable software within tough constraints, NASA has a chance to deliver even more than the results of the scientific probes it launches. “We’re the engineers,” Tom Gavin says, “but we do this in the name of science.” Whatever progress the space agency makes in addressing its own software issues seems likely to advance the state of the art, contributing to computer science as well as software engineering. In addition, any successful spin-off that improves reliability while cutting development time and costs could, in principle, generate savings for US industry equal to the nation’s budget for space exploration.

Acknowledgments

We thank Erik Antonsson, Matthew Barry, Guillaume Brat, Daniel Dvorak, Martin Feather, Erann Gat, Tom Gavin, Dimitra Giannakopoulou, Klaus Havelund, Gerard Holzmann, Michael Lowry, Kenny Meyer, David Nichols, Tom Prince, Kirk Reinholtz, Nicolas Rouquette, and Willem Visser for their time. In addition, we owe a great debt to Lisa Townsend at JPL and Michael Mewhinney and John Bluck at Ames for arranging the interviews with these researchers.

References

1. D. Dvorak et al., “Software Architecture Themes in JPL’s Mission Data System,” *Proc. AIAA Space Technology Conf. and Exposition*, 1999; http://x2000.jpl.nasa.gov/nonflash/publications/aiaa99_mds_final.pdf.
2. G. Holzmann, *The Spin Model Checker: Primer and Reference Manual*, Addison-Wesley, 2003.
3. K. Havelund and T. Pressburger, “Model Checking Java Programs Using Java PathFinder,” *Int’l J. Software Tools for Technology Transfer*, vol. 2, no. 4, 2000, pp. 366-381.
4. W. Visser et al., “Model Checking Programs,” *J. Automated Software Engineering*, vol. 10, 2002, pp. 203-232; http://ase.arc.nasa.gov/jpf/papers.html.
5. K. Havelund and G. Rosu, “Monitoring Java Programs with Java PathExplorer,” *Proc. Workshop Runtime Verification, Electronic Notes in Theoretical Computer Science*, vol. 55, no. 2, 2001.
6. D. Dvorak, “Challenging Encapsulation in the Design of High-Risk Control Systems,” presentation, *Conf. Object-Oriented Programming, Systems, Languages, and Applications*, Seattle, 2002.

***Patrick Regan** is senior correspondent for science and technology at* NJN News *on New Jersey’s public television and radio network. Contact him at patrick.regan@ieee.org.*

***Scott Hamilton** is* Computer*’s senior acquisitions editor. Contact him at s.hamilton@computer.org.*

IT Employment Prospects in 2004: A Mixed Bag

Signs of general economic recovery and renewed IT sector growth are dampened for IT workers globally by improved productivity and locally by a global labor market.

Fred Niederman
Saint Louis University

A recent issue of *CIO* magazine featured a successful information technology employee from the United States, Mike Emmons, who skillfully adapted to several generations of new technology and apparently provided excellent value in his work, but was laid off along with his entire work group.[1] What's unusual is that Emmons and his colleagues received a significant severance package on the condition that they train their replacements, whom the company brought from India to work at substantially lower wages.

Clearly things have changed since 1999, when the cover stories for US business magazines described IT positions going unfilled and extensive congressional lobbying to increase quotas for overseas workers to fill them. For US workers, the change raises questions of industry direction and national policy. However, the change also signals the reality of a global marketplace. From this perspective, it raises two different but complementary questions:

- What are the prospects for the global IT workforce in the near and longer term?
- How will IT jobs be distributed among competing labor markets around the world?

The first question addresses total demand for IT workers. The second addresses the economics of business investments and how governments, universities, communities, and individuals can compete successfully for them.

A SHORT HISTORY: 1997-2003

Following the euphoric increase in the demand for IT professionals from 1997 to 2000, the recession of 2000-2001 marked new highs in unemployment. Earlier recessions had also triggered unemployment, but not to the same degree. By the end of 2002, unemployment among US mathematicians and computer scientists, a subset of IT professionals, rose to the highest levels since 1982, when the Bureau of Labor Statistics started collecting the data.[2]

Although issues relating to definition and measurement precision have prevented scientific confirmation, most industry observers agree that demand for IT workers in the late 1990s exceeded supply, certainly in the US. For example, a 1999 review of US labor literature[3] showed many signs of a shortage, including rising wages for IT workers and unfilled job openings.

This changed, of course, when the Internet investment bubble burst, and many Internet firms went out of business. At the same time, companies that had invested heavily in Y2K conversions began releasing employees hired specifically for that task. With the general downturn in the economy, companies canceled new IT projects or put them on hold, further slowing new hiring and adding to layoffs. These general tendencies all worked in concert to increase the availability of IT workers for a decreasing number of positions.

US workforce shrinkage

The US IT workforce shrank significantly between 2000 and 2003. The Information Technology

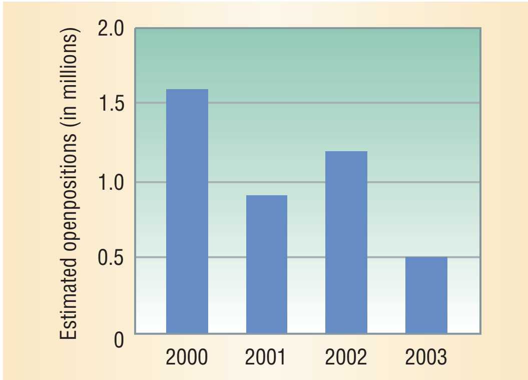


Figure 1. US IT workforce demand based on estimated open positions indicates a loss of more than 1 million positions between 2000 and 2003.

Association of America, a trade association representing US IT industry, estimated a reduction of 500,000 IT workers between 2001 and early 2002—from 10.4 to 9.9 million total.[4] Figure 1 shows more recent ITAA data, estimating US demand for IT workers in terms of open positions.[5] The estimated number of open positions peaked in 2000 at 1.6 million, dropped below 1 million in 2001, came back somewhat in 2002 to 1.2 million, then fell dramatically to less than half a million in 2003.

Another indication of declining IT work in the US and other countries—notably Britain, Germany, and Canada—is reduced recruiting from overseas. Programs like the H-1B visa program in the US, which had increased annual quotas to 195,000 foreign scientists, engineers, and programmers, are no longer filling their entire quota. The US recently returned to a cap of 65,000 such workers per year.

Likely global shrinkage

Job shrinkage in the US does not seem to correlate with equivalent or larger job growth in other countries. For example, Narendra M. Agrawal and Mohan Thite[6] cite reports from the National Association of Software and Service Companies in India indicating that the number of IT professionals increased from 6,800 in 1985-1986 to 522,000 in March 2002. We don't know if the March estimate represents growth, stability, or even shrinkage in India between 2000 and 2001.

Even if we assume an accelerating growth rate that added 100,000 of those workers in 2001, it would not offset the decreased number of US jobs. Nor is it likely that an increase in jobs throughout the European Union and Eastern Europe would account for the balance.

The evidence is not strong, but the best tentative conclusion is that the global IT workforce has contracted since 2000.

TOTAL DEMAND FOR IT WORK

Workforce contraction can result from both normal business cycles and from structural changes in the work.

Productivity issues

The nature of IT work, particularly in companies developing non-IT products, has been evolving steadily away from a structure of individual programmers banging out code in one language for use in one environment. Instead, software development has become a team activity that often extends existing systems written in multiple languages for diverse execution environments.

Modular development. This shift from building stand-alone systems to extending existing portfolios should generate new IT products more efficiently. It should also increase overall productivity because incremental module additions to an existing applications portfolio accrue benefits as they are installed, in contrast to approaches that require completing a monolithic system before the first benefits can ensue.

Adding new modules rather than developing massive new systems lets programmers work on smaller projects, which generally means less complexity and greater efficiency. It also decreases the need to maintain detailed documentation for coordination among many workers on a large project. The documentation for smaller projects might be crucial but should take less time to create.

A modular approach also increases opportunities for reuse and component-based systems that should decrease the programming time required to complete projects. It gives organizations more flexibility in system development. They can order new modules to fit with changes in needs, labor availability, and financial resources.

As development environments grow in sophistication, they offer more and better tools for code reuse and generation, automated testing, version control, and structured design methods. In addition to programmer-level productivity gains, companies can use organizational project environments like the Software Engineering Institute's Capability Maturity Model as tools for measuring and standardizing their development processes. If successfully implemented, these tools should result in more effective use of IT human resources through better coordination, documentation, and organizational learning.

The IT field has likely increased its productivity over the past several years. Even as the total IT professional workforce has declined in numbers, the amount of retail e-commerce has increased. As Figure 2 shows, the US Department of Commerce estimated that the dollar value of retail e-commerce in the US doubled from $6.25 billion during the second quarter of 2000 to $12.48 billion during the second quarter of 2003.[7] This suggests an improved capacity to handle not just overall value but probably also numbers of transactions, even as the overall number of employees has shrunk.

Shifts in the means of production for IT workers—changing from custom development to package integration—will tend to move development work from non-IT-producing companies to IT-producing companies. It will also shift the total amount of IT work required: Doubling the staff for a package creator may reduce staff requirements by a few percentage points at hundreds of non-IT-producing companies, creating a net productivity gain but potentially reducing the overall demand for IT workers.

In practice, the net outcome of such structural shifts is difficult to predict. At non-IT-producing companies, the workforce can decrease, but the company can also reallocate staff to other projects and achieve a net balance of workers as maintenance requirements increase with increased total IT activities and tools to support.

Similarly, shifting IT work from a non-IT-producing firm to a domestic outsourcer may accomplish the same level of work with fewer workers, assuming the outsourcer can create the same outputs more efficiently. However, the outsourcer often absorbs some or all of the non-IT-producing firm's employees. The firm might retain some employees to "interface" with the outsourcing company as communicators or monitors or to work on new application development or other projects.

Distributed complexity. Many scenarios arise when IT productivity increases. If the demand for new IT capabilities remains constant, fewer workers can meet it. If the demand decreases, even fewer workers are needed to meet it. More interesting is the case in which demand increases along with productivity. If productivity increases faster than demand, the number of workers could still decrease. Only when demand increases faster than productivity can the number of workers also increase.

Although this provides a useful mental framework, in practice many factors over and above the addition of new portfolio elements can stimulate the hiring of new IT workers. Increases in portfolio sizes correlate with a shift toward more distributed, multiplatform environments and more complexity in the organization's full expanse of computer-oriented capabilities. The labor needed to maintain the system should increase with these changes.

The relationship between complexity and labor could change if efforts to create "self-repairing" software and systems become commonplace. Barring significant changes in self-repair capabilities, however, aging system portfolios combined with increasingly diverse computing environments should continue to create new levels of technical complexity. Ultimately, the good news for global IT workers—though not necessarily IT consumers—is that complexity tends to increase faster than the number of elements added to a system.

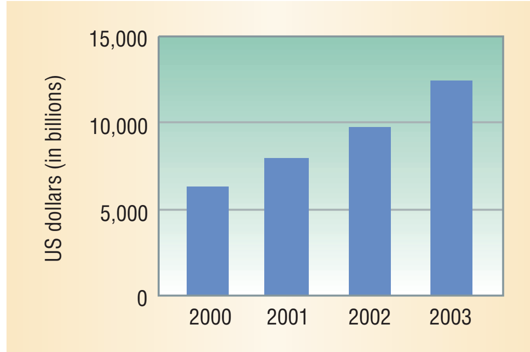


Figure 2. E-commerce in the United States doubled from 2000 to 2003, even as the workforce decreased.

Product and service innovations

New IT products and services also create new opportunities for growth. IT has automated many tasks, but substituting computer labor for human labor stimulates the conceptualization of additional tasks for automation and support. For example, bank ATMs have evolved to "credit at the pump" gas stations. Perhaps the next step will automate a "discussion" between each vehicle and a pumping system algorithm that optimizes the blend of gas and additives for higher performance and lower emissions. The savings in fuel could more than pay for the investment in distribution facilities as fuel reserves become more scarce.

Although I am not aware of any single "killer app" on the horizon, numerous "mini-killer apps" exist across a variety of niches from radio frequency identification and supply-chain technologies to Wi-Fi and endlessly morphing wireless applications. That millions of people have bought into cell phone technologies and personal digital assistants prepares the ground for enterprising individuals and organizations to consider new uses for these tools, which in turn will require employing workers to develop, test, implement, upgrade, and eventually maintain.

Workforce characteristics

Another factor that influences the need for new employees is the rate at which existing employees leave the field. At present, the IT field—at least in some sectors—includes a large number of individuals expected to retire within the next few years. This seems particularly true in the US public sector. A recent survey of US government technology workers found their average age was 45 years, and more than half were between ages 45 and 69.[8] Many state officials also report aging staffs, which could jeopardize IT service delivery if many of these

Table 1. Average annual IT salaries by country, June 2003.

Country	Average annual salary in local currency		Average annual salary in US dollars
United Kingdom	50,757	GBP	81,553
United States	80,286	USD	80,286
Europe	59,406	EUR	68,218
Australia	95,559	AUD	62,257
Canada	79,117	CAD	56,599
Singapore	75,319	SGD	43,058
South Africa	225,922	ZAR	30,055
Brazil	57,427	BRL	19,982
India	397,818	INR	8,593

Source: 2003 Information Technology Toolbox Salary Survey; http://crm.ittoolbox.com/Research/survey.asp. Used by permission.

experienced workers opt to retire when they become eligible.

One mitigating factor in assessments of the IT workforce size is the diffusion of computing from the sole domain of IT specialists to a wider array of workers. Increasing numbers of workers employed at the IT boundary—Web developers, business analysts, and specialized end users such as nursing informatics and geographic information systems specialists—use sophisticated IT components in their work without being classified technically as IT workers. This might also pertain to individuals in IT-producing firms who evolve from clearly technical workers to positions of customer support that blend technical and marketing activities.

Another influence on workforce size pertains to the trend toward shifting IT jobs from traditional full-time employment to contract work. Many kinds of contractual arrangements exist between employees and firms. Some arrangements resemble traditional work except in terms of pay and benefits; others are project-oriented, with significant employee movement. Current methods for estimating workforce size may not capture all of the nuances of the arrangements that contractors have with organizations. For example, a full-time employee might still be counted as an IT worker even if between assignments, whereas a contract employee might not be.

Growth in installed IT base

Assuming that global commerce and trade continue to grow, new applications—from customer resource management to ubiquitous computing—should continue to capture the imagination of organizations and customers. Given organizational creativity in developing new applications, the immense body of installed software requiring maintenance and updating, and the public appetite for new IT-enabled products, it is reasonably safe to predict continued growth of the installed IT base around the world.

While we may not see the booming growth of the 1990s Internet explosion or a Y2K threat, there is no indication of significantly shrinking demand for IT products and services.

GLOBAL COMPETITION FOR IT WORK

If the downturn in global IT employment largely represents a natural business cycle, then the total IT workforce size will increase as effects of the recent recession fade. If so, the issue for most IT workers and employers shifts to concern for how this work is distributed among them.

Wages

In classic economic theory, when supply increases and demand decreases, prices fall. In fact, organizations generally find it difficult to reduce the wages of full-time employees, though they might reduce benefits and halt or delay bonuses and raises.

The sharp growth in IT wages prior to the dot-com bubble burst leveled out afterward but still continued to grow at a slower rate until very recently, when they began to flatten and even to recede for some job titles. A recent survey published in *Software Development* suggests that in the US, both average and median manager salaries have increased slightly between 2002 and 2003, while salaries of other IT workers stayed about the same or declined very slightly during the same period.[9] The ITAA mirrored this finding in its May 2003 report, which concluded:

> While compensation for IT workers is not by and large growing, most workers are not seeing cuts in pay or benefits. Amidst an uncertain business environment, companies appear determined to maintain the status quo.

Table 1 shows survey results for worker salaries in different countries.[10] The differences are significant, indicating that annual IT salaries in the US and UK average approximately 9.5 times the average in India. This data does not indicate changes over time, but the difference would motivate an organization to at least consider hiring an Indian IT worker rather than one from the US or UK, if all else were constant.

Education pipeline

Just a few years ago, IT careers could be sold to students as an entrée into an exciting, profitable world where labor shortages meant that talented folks could almost name their price. Now, at least in the US, the number of students entering computer science and business management information system (MIS) programs has decreased significantly.[11]

Part of the decrease may reflect recent changes in immigration procedures that have made it more difficult for foreign students to pursue studies in the US. In November 2003, the Institute for International Education showed the number of students coming to the US was still increasing but at a much slower rate than in previous years.[12]

> The most popular fields of study for international students in the U.S. are business and management (20%) and engineering (17%). After two years of very large growth, the number of international students studying mathematics and computer sciences has decreased by 6%, although these students still make up 12% of the total.

It is logical, however, to expect enrollments in fields generally leading to IT careers to decline as a result of downsizing and industry layoffs. Although many students are still drawn to IT for its intrinsic rewards, it has become more difficult to attract students at the margin into IT. Careers must look much less stable and rewarding than they appeared to be a few years ago when dot-com millions seemed near at hand.

Selective hiring

Even as organizations reduce their workforces overall, they are still hiring individuals with selected skill sets. For example, IBM has been cutting jobs over the past year, but a CNET.News.com story quotes CEO Sam Palmisano indicating that the company is also looking to add positions:[13]

> As we look to 2004, more customers are expected to increase their investments in information technology. … We see the need for approximately 10,000 new positions in key skill areas, including high-value services, middleware technologies, Linux, and open standards-based hardware and software.

Overall, employers can be more selective in the education and experience they require for new hiring. A longitudinal study based on IT positions advertised in a variety of US locations from 1988 to 2001 found more than 10 percent increases in number of jobs for software engineers, with equivalent decreases for programmer/analysts and consultants.[14] The study showed no change in the percentage of permanent versus contract workers (consultants declined from about 5 percent to about 3 percent).

Even as organizations reduce their workforces overall, they are still hiring individuals with selected skill sets.

Potential shortages

What might happen if the business cycle turns once again toward strong growth in the IT sector?

If the growth is strong enough to absorb all the people who lost their jobs during the 2001 recession and its aftermath, the US could return to a shortage situation. First, the number of students in the computer science and information systems pipeline has decreased significantly. Second, even if the US decides to reinvigorate the H-1B program, it faces a number of difficulties. For example, the projections of job growth for software developers inside India are significant. Estimates as of March 2003 give India about 500,000 IT workers, and the projected need is for 2.2 million in 2008. Although Indian IT workers have historically prized the prospect of coming to the US,[6] their eagerness may be dampened by observing the effects of thousands of H-1B Indian workers essentially forced to return home during the recession.

English-speaking countries have an advantage in recruiting Indian workers relative to Japan and Germany, for example, where language poses a barrier. Moreover, the long history of Indian workers settling in the US offers an existing community into which they can quickly be absorbed.

About 150,000 newcomers are estimated to be ready to enter the job market in India.[6] If the estimates are correct, a worldwide shortage of IT workers could create significant opportunities for other countries, particularly in Eastern Europe, to replicate some of India's success in this domain.

Effects of outsourcing

Several factors over the past five years have led to increased and more effective use of what in the US is called "offshore outsourcing":

- improved communications technologies that allow smoother coordination of development work;
- accumulated experience and knowledge about what is needed to successfully manage fully outsourced or to integrate partially outsourced work;
- highly trained individuals in target countries, particularly in the area of coding;
- cost differential for employing workers in lower-cost-of-living countries; and

Even in optimistic scenarios in which the demand for IT work recovers from cyclical business forces, IT workers face increasing competition.

- well-organized enterprises in the target countries.

With regard to the last point, note that the first company in the world to receive Capability Maturity Model Level 5 certification was in India.[6]

According to the 2003 ITAA study,[5] some 12 percent of US IT-producing companies have already moved jobs offshore, but only 3 percent of non-IT producing companies have done so. This is likely to reflect a greater reluctance of companies focused on integrating packages and maintaining strategic applications to outsource relative to companies that may be creating discrete products, such as software packages, or providing help desk or related types of support. The percentages, however, are reasonably similar to Gartner group estimates that approximately 10 percent of IT work currently performed in US IT-producing companies will be outsourced to workers in other countries, along with about 5 percent of the IT work in non-IT companies.[15]

Effects on total workforce. Several factors might account for US outsourcing not reaching even higher percentages.

Although offshore salaries can be substantially lower, they do not represent the total cost comparison to domestic development. The full cost includes costs of vendor selection, transitioning the work, layoffs and retention, lost productivity and cultural issues, improving development processes, and managing the contract. One estimate puts these costs at from 15 to 55 percent above the contract.[16] Moreover, some costs occur up front and take time and energy to recover. In addition, cost advantages might not be easy to maintain. For example, salaries have increased an estimated 20 percent annually in recent years in India.[6]

Firms will likely be less inclined to outsource if they have any security concerns regarding their data or systems. Although it is not clear that IT is more secure in one location than another, crossing borders can add complexity, which can add to general security risks. On one hand, redundant work in multiple locations can reduce the risk from disruption in any single location; on the other hand, security is only as strong as the weakest element, and multiple locations offer multiple potential attack points.

Intellectual property can be a concern for some organizations. Individuals working in an outsourced organization might be tempted to form new companies using know-how from their parent organization to create new and often very effective competition.

Even in optimistic scenarios in which the demand for IT work and workers recovers from cyclical business forces, IT workers in the US, Canada, the EU, and even India will face increasing levels of competition. The number of qualified workers and indigenous IT enterprises continues to grow throughout many parts of the world. For example, in recent years Eastern Europe has shown significant economic growth and striking expansion of demand for telecommunications, wireless, and cable connections.[17] All the Eastern Europe countries are significantly increasing their capability to perform IT work.

Systemwide benefits. The *McKinsey Quarterly* published a report outlining four systemwide benefits that accrue to countries that outsource jobs to other countries:[18]

- cost savings that may lower prices and increase competitiveness in the marketplace;
- new revenues based on the growth in developing economies;
- repatriated earnings, which may allow companies to remain in business and return higher profits to their investors—many of whom can reside in the outsourcing country; and
- redeployed labor to the extent that IT workers can move into higher-value projects, develop entrepreneurial new products, or shift to other industries.

In India, for example, although an estimated two-thirds of revenue for IT products comes from outside the country, the remainder points to a large and potentially growing market for products and services that could come from inside its borders.

Much of the McKinsey analysis is based on broad economic data across industries. IT people in general could be affected more adversely in shifting to new industries as they tend to start from a higher salary base that is difficult to match in other areas.

Effects of insourcing

Offshore outsourcing has been increasing, but "insourcing" also remains significant in the US. An Associated Press news story reported that the US State Department issued 28,098 L-1 visas during the first half of fiscal 2003, "... an increase of nearly 7 percent from the same period in 2002."[19] The L-1 visa lets multinational firms transfer personnel for specialized tasks to the US from their station in other countries. In contrast, the H-1B visa is intended to supplement the US workforce in critical areas. An H-1B visa lets the holder stay in the US for five years based on organizational sponsorship. Agrawal and Thite[6] estimate that as many as 80 percent of the Indian IT workers who have come to the US on this program have become permanent residents or citizens.

Some educational institutions, notably Stevens Institute of Technology,[11] are recognizing the increasingly multicultural US workforce by including global IT courses and emphasizing cross-cultural communication in their curricula.

Professionals who move to the US and other Western countries represent a potential "brain drain" to their native countries, but workers who do return home usually have accumulated a valuable array of experience, training, and capital for investment. For example, a United Nations report found that returnees from the US started half of all the companies emerging from Taiwan's largest scientific park, the Hsinchu.[20] However, not all countries are as successful in attracting significant numbers of their most educated people to return.

From a US national perspective, insourcing jobs has many advantages over outsourcing. Insourced workers buy homes, cars, food, and otherwise stimulate the overall economy. Their potential for innovation tends to be high, which can in turn create new enterprises and jobs. Finally, insourcing decreases the risk of losing intellectual capital to potentially competing countries.[6]

What does the US or any country need to do to remain or become competitive in the world IT market? Success in developing a presence in this market is probably tied to disproportionate investment in science and technology by government and the private sector. One UN report,[16] for example, notes that both India and China invest in science and technology at unexpectedly high levels relative to their per capita income. Ireland went to great lengths to develop its public university system to support its success story in entering the world IT market.[21]

Such investment is risky business, though, in a world where large numbers of highly trained professionals make the return on such an investment less than certain.

Workplace issues

Recent IT research—mostly in the business MIS literature—focuses on applying emerging human resource management models to IT professionals. One intriguing set of studies addresses the work life of IT professionals. For example, in a study of the effects of prolonged stress on IT workers, Jo Ellen Moore[22] found evidence that burnout among IT professionals affects their decision to leave an employer and the IT field. Moore also found that men and women experience the effects differently. The implications are perhaps muted in the current economic climate, but if the labor market proves cyclical and we return to IT worker shortages, issues of retaining skilled staff could again prove troubling.

Both India and China invest in science and technology at high levels relative to their per capita income.

Employee turnover can be expensive not only in immediate financial terms but also in terms of lost intellectual capital and strategic power. Recent studies focused on IT workers have indicated a bifurcation in strategic approaches to retaining IT employees.[23] With the known half-life of IT skills, a segment of the industry and of non-IT-producing companies actually encourages a reasonably high rate of turnover.

The ultimate effect of this strategy on the individual firm remains unclear. In the short run, it may replace higher-paid experienced workers with lower-paid newcomers. It could have a reverse short-run effect if market prices are rising quickly. However, this policy is not likely to reward individual IT professionals for long years of difficult education to prepare for entering the field, and it may adversely affect the retention of IT workers in the labor market as a whole.

Soon Ang and Sandra Slaughter[24] have considered these labor strategies in a rich framework that contrasts the emphases for internal and external labor markets. Companies favoring internal labor markets tend to emphasize training and career paths to retain firm-specific knowledge; those favoring external labor markets tend to hire people with specific skills for specific projects.

Neither of these strategies is optimal for all firms in all situations, but consciously choosing a strategy and blending human resource practices that match it can be helpful to individual organizations. If the collective organizations that hire IT professionals want to make sure that sufficient employees are available in an economic upturn, they may need to develop institutions that can blunt the risk for individual workers during the downturns.

PROSPECTS FOR 2004

Prospects for IT workers for this year are difficult to reduce to a single generalization. The third-quarter annualized growth rate of the US gross domestic product was 8.2 percent, suggesting renewed demand for IT workers; but it is not clear whether this increase will offset ongoing productivity improvements that enable fewer workers to do more work.

All indications point to the continued migration of jobs from the US, Canada, and Europe to developing countries—especially India, but also China, Russia, and Eastern European countries. The migration has been strongest among programmers and IT service providers, such as help desk workers, but it could expand to designers and project management.

The IT job market also looks very different for workers in IT-producing companies such as Microsoft and Sun, outsourcing contract recipients such as EDS and Tata, and non-IT-producing companies such as McDonald's and Disney. The competitive advantages of IT workers in development companies typically emphasize mastery of tools and technologies for creating innovative products, whereas IT workers in outsourcing and non-IT companies need to know how to integrate and apply these same tools and technologies.

Clearly, macroeconomic forces will bring windfalls to some companies—perhaps those providing the outsourced services—even as they constrain others—perhaps IT-producing companies that can diminish their risks by developing products across various geographic locations.

Unfortunately for the individual IT worker, there are few ways to lower individual risks. The ITAA reports imply the need to accept continued loss of jobs that can be replicated in lower-cost countries. These reports also suggest the need for constantly upgrading skills in cutting-edge technology. Broadening skills from the purely technical may also be helpful in retaining work or finding new opportunities. Technical professionals with good business and communication skills can often find work supporting the interface between technology and business in areas such as data analysis and project management.

A recent book from Mercer Human Resource Consulting[25] concludes with a discussion of tactics for developing "self-managed careers." It emphasizes principles such as blending general with firm-specific skills, understanding the labor market dynamics within an organization, and taking a role in assisting others in managing their careers. But professionals must balance such general advice with real competition from newly minted IT workers whose skill set may be narrow but is probably targeted to emerging technologies. Together with their availability at often lower cost, these workers can represent an unbeatable advantage for any given job. In this example, firm-specific skills—where to go for technical or financial assistance within the firm—are not easily transferred nor necessarily recognized for the value they potentially represent.

Not everyone has all the requisites for creating new products or services and setting up their own enterprises, but individual IT workers should not lose sight of this possibility. I personally lived through major aviation engineering layoffs in Seattle, which ultimately forced many engineers to start new companies that resulted in a stronger, more balanced regional economy over a period of some years.

Individual IT workers might also need to be flexible in terms of physical location. Shifts in hot technologies can move job centers from place to place—for example, from e-commerce startups in Silicon Valley to government security specialties in Washington, D.C.

The IT profession offers numerous rewards, including intrinsically satisfying work—particularly when creating new systems or solving user problems. It also includes the opportunity for receiving average to excellent financial compensation. But as engineers in many fields have learned over the years, it is not immune from marketplace risks. ■

References

1. B. Worthen, "The Radicalization of Mike Emmons," *CIO Magazine*, 1 Sept. 2003; www.cio.com/archive/090103/people.html.
2. E. Frauenheim, "Taking the Measure of IT's Pain," *CNET News.com*, 8 July 2003; http://news.com.com/2008-1082-1023565.html.
3. P. Freeman and W. Aspray, *The Supply of Information Technology Workers in the United States*, Computing Research Association, 1999.
4. Information Technology Association of America, "Bouncing Back: Jobs, Skills and the Continuing Demand for IT Workers," ITAA white paper, May 2002.
5. Information Technology Association of America, "2003 Workforce Survey," ITAA tech. report, May 2003.
6. N.M. Agrawal and M. Thite, "Human Resource Issues, Challenges, and Strategies in the Indian Software Industry," *Int'l J. Human Resources Development and Management*, vol. 3, no. 3, 2003, pp. 249-264.
7. US Census Bureau, "Retail E-Commerce Sales in Third Quarter 2003 Were $13.3 Billion, Up 27.0 Percent from Third Quarter 2002," *Census Bureau Reports*; www.census.gov/mrts/www/current.html.
8. E. Chabrow, "Gap Analysis," *Government Enter-*

prise, Aug. 2003; www.governmententerprise.com/issue/002/itworkforce_01.html.
9. A.W. Morales, "Salary Survey 2003: What Are US Developers Worth?" *Software Development*, Nov. 2003; www.sdmagazine.com/documents/s=8939/sdm0311a/ sdm0311a.html.
10. R. Greenspan, "UK, US IT Pros Top the Salary Charts," *CyberAtlas*, 31 July 2003; www.cyberatlas. internet.com/markets/professional/print/0,,5971_2242531,00.html.
11. T. Hoffman, "Job Skills: Preparing Generation Z," *Computerworld*, 25 Aug. 2003; http://computerworld.com/careertopics/careers/story/0,10801,84295,00.html.
12. Institute for Int'l Education, "International Student Enrollment Growth Slows in 2002/2003, Large Gains from Leading Countries Offset Numerous Decreases," IIE press release, 3 Nov. 2003; http://opendoors.iienetwork.org/?p=36523.
13. E. Frauenheim, "IBM Meets Earnings Target, Talks of New Jobs," *CNET.News.com*, 15 Oct. 2003; http://news.com.com/2100-7342_3-5091980.html.
14. M.J. Gallivan, D.P. Truex, and L. Kvasny, "Changing Patterns in IT Skill Sets 1988-2003: A Content Analysis of Classified Advertising," to be published in *Data Base*, 2004.
15. M. LaMonica, "Panel: Sharpen Skills to Keep U.S. Tech Edge," *CNET.News.com*, 3 Oct. 2003; http://zdnet.com.com/2100-1104-5086181.html.
16. S. Overby, "The Hidden Costs of Offshore Outsourcing," *CIO Magazine*, 1 Sept. 2003; www.cio.com/archive/090103/money.html.
17. D. Halperin, "Worlds in Transition: IT in Eastern Europe," *TechNewsWorld*, 6 Sept. 2003; www.technewsworld.com/perl/story/31510.html.
18. V. Agrawal and D. Farrell, "Who Wins in Offshoring?" *McKinsey Quarterly*, no. 4, 2003; www.mckinseyquarterly.com/article_abstract.asp?ar=1363&L2=7&L3=10.
19. R. Konrad, "U.S. Tech Workers Training Their Replacements," Associated Press, 11 Aug. 2003; www.cnn.com/2003/TECH/biztech/08/11/training.replacements.ap/index.html.
20. A. Solimano, "Globalizing Talent and Human Capital: Implications for Developing Countries," United Nations Economic Commission for Latin America and the Caribbean, 2002; http://wbln0018.worldbank.org/eurvp/web.nsf/Pages/Paper+by+Solimano/.
21. E.M. Trauth, *The Culture of an Information Economy: Influences and Impacts in the Republic of Ireland*, Kluwer, 2000.
22. J.E. Moore, "One Road to Turnover: An Examination of Work Exhaustion in Technology Professionals," *MIS Quarterly*, Mar. 2000, pp. 141-168.
23. R. Agarwal and T.W. Ferratt, "Crafting an HR Strategy to Meet the Need for IT Workers," *Comm. ACM*, July 2001, pp. 58-64.
24. S. Ang and S. Slaughter, "Retaining Information Technology Professionals: The Effects of Internal Labor Market Strategies," to be published in *Data Base*, 2004.
25. H.R. Nalbantian et al., *Play to Your Strengths: Managing Your Internal Labor Markets for Lasting Competitive Advantage*, McGraw-Hill, 2004.

***Fred Niederman** is the Shaughnessy Associate Professor of Management Information Systems in the Cook School of Business at Saint Louis University. His research interests focus primarily on human resource issues pertaining to computer and MIS personnel and secondarily on global information systems and collaborative, group support systems. Niederman received a PhD in management information systems from the Carlson School of Business at the University of Minnesota. He is a member of the Association for Information Systems, the ACM, the Society of Information Management, and the Information Resource Management Association. Contact him at niederfa@slu.edu.*

DakNet: Rethinking Connectivity in Developing Nations

DakNet provides extraordinarily low-cost digital communication, letting remote villages leapfrog past the expense of traditional connectivity solutions and begin development of a full-coverage broadband wireless infrastructure.

Alex (Sandy) Pentland
MIT Media Laboratory

Richard Fletcher

Amir Hasson
First Mile Solutions

As a government representative enthusiastically talks about the new telephone for a village in remote rural India, a villager asks, "Who am I going to call? I don't know anybody who owns a telephone." Yet, despite this sensible observation, a phone is dutifully installed as part of the current government mandate to connect villages to neighboring towns. Although some villagers do use the phone occasionally, most still travel sometimes days to talk to family or to obtain the forms and other data that citizens in developed nations can call up on a computer in a matter of seconds.

In short, the goal of "broadband connectivity for everyone" has been shelved in favor of cutting back to the minimum possible standard telephone service in the mistaken belief that this is the cheapest way to provide connectivity. This compromise is particularly tragic given recent advances in wireless technology, which make running a copper line to an analog telephone far more expensive than broadband wireless Internet connectivity. Rather than backpedal on the goal of connecting everyone, society should be thinking, How can we establish the kernel of a user network that will grow seamlessly as the village's economics develop? In other words, what is the basis for a progressive, market-driven migration from government seed services—e-governance—to universal broadband connectivity that local users will pay for?

DakNet, an ad hoc network that uses wireless technology to provide asynchronous digital connectivity, is evidence that the marriage of wireless and asynchronous service may indeed be that kernel—the beginning of a road to universal broadband connectivity. Developed by MIT Media Lab researchers, DakNet has been successfully deployed in remote parts of both India and Cambodia at a cost two orders of magnitude less than that of traditional landline solutions. Villagers now get affordable Internet services—and they're using them. As one man in a small village outside of New Delhi remarked, "This is better than a telephone!"

THE WIRELESS CATALYST

Recent advances in wireless computer networking—particularly the IEEE 802 standards—have led to huge commercial success and low pricing for broadband networks. While these networks are viewed as mainly for offices or for hotspots in urban areas, they can provide broadband access to even the most remote areas at a low price. Today, wireless cell phone and wireless local loop (WLL) service costs roughly a third of copper or fiber landline service, while packet-based broadband computer networks cost roughly a ninth of the landline service—and they are far friendlier to data services and to lower-grade voice service such as voice messaging. These new technologies thus offer developing countries an opportunity to leapfrog over wireline and WLL telephony infrastructure to the forefront of broadband communications technology.

Wireless data networks based on the IEEE 802.11, or WiFi, standard are perhaps the most

Published by the IEEE Computer Society

promising of the wireless technologies. The forces driving the standardization and proliferation of WiFi in the developed world have resulted in features that can stimulate the communications market in the developing world. These features include ease of setup, use, and maintenance; relatively high bandwidth; and, most important, relatively low cost for both users and providers.

As one demonstration of the practicality of this new technology for rural connectivity, researchers from the Indian Institute of Technology at Kanpur, working with Media Lab Asia (www.medialabasia.org), have "unwired" a 100-square km area of the Gangetic Plain in central India. Figure 1 shows the corridor. This project provides broadband connectivity along a corridor with almost one million residents, at a projected one-time cost of under $40 per subscriber. Other experiments have shown the practicality of the technology in mountainous terrain and in city centers. Indeed, several cities in the US have begun to deploy free Internet connectivity using IEEE 802.11b.

Even with advances such as those demonstrated in the Digital Gangetic Plain project, the cost of real-time, circuit-switched communications is sufficiently high that it may not be the appropriate starting point for rural connectivity in developing nations. Market data for information and communication technology (ICT) services in rural India strongly implies that asynchronous service—voice messaging, e-mail, and so on—may be a more cost-effective starting point for rural connectivity projects.

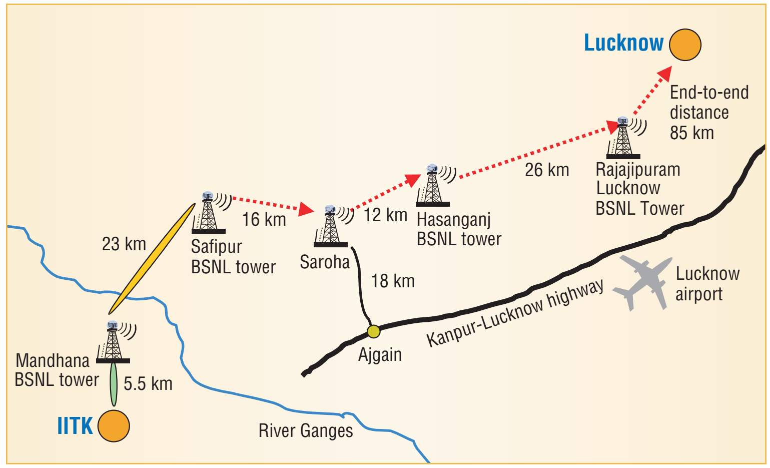


Figure 1. Digital Gangetic Plain project. Map shows the corridor of wireless technology in central India.

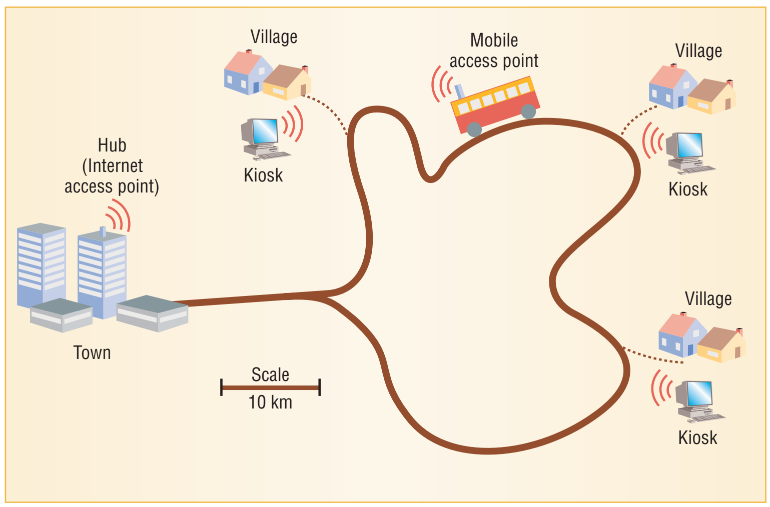


Figure 2. DakNet concepts. Physical transport, in this case a public bus, carries a mobile access point (MAP) between village kiosks and a hub with Internet access. Data automatically uploads and downloads when the bus is in range of a kiosk or the hub.

MOBILE AD HOC CONNECTIVITY

The DakNet wireless network takes advantage of the existing communications and transportation infrastructure to distribute digital connectivity to outlying villages lacking a digital communications infrastructure. DakNet, whose name derives from the Hindi word for "post" or "postal," combines a physical means of transportation with wireless data transfer to extend the Internet connectivity that a central uplink or hub, such as a cybercafe, VSAT system, or post office provides.

As Figure 2 shows, instead of trying to relay data over a long distance, which can be expensive and power hungry, DakNet transmits data over short point-to-point links between kiosks and portable storage devices, called mobile access points (MAPs). Mounted on and powered by a bus, a motorcycle, or even a bicycle with a small generator, a MAP physically transports data among public kiosks and private communications devices (as an intranet) and between kiosks and a hub (for non-real-time Internet access). Low-cost WiFi radio transceivers automatically transfer the data stored in the MAP at high bandwidth for each point-to-point connection.

DakNet operation thus has two steps:

- As the MAP-equipped vehicle comes within range of a village WiFi-enabled kiosk, it automatically senses the wireless connection and then uploads and downloads tens of megabytes of data.
- When a MAP-equipped vehicle comes within range of an Internet access point (the hub), it automatically synchronizes the data from all the rural kiosks, using the Internet.

The steps repeat for every vehicle carrying a MAP unit, thereby creating a low-cost wireless network and seamless communications infrastructure.

Some Common Myths about Rural Information and Communication Technology

***Myth:* The village telephone is the best model for poor communities.**
***Truth:* Giving everyone access to digital messaging—voice mail, digital documents, e-mail, and so on—is better than installing a community telephone.**

Rural information and communication technology (ICT) is typically introduced as a communications channel that the community shares. Whether through a public call office (PCO) or a public computer kiosk, users are introduced to ICT as shared utilities with a technically literate operator acting as an intermediary.

In this shared-use model, much ICT has relied on real-time communications, such as landline telephone, cellular phone, or satellite radio links. These real-time technologies can be useful for immediate interactivity and accessing highly time-sensitive information. Successful examples include India's PCOs and the GrameenPhone initiative (www.grameenphone.com/).

While successful at providing basic services, the strategy of deploying shared, real-time communications also has serious drawbacks. One is the large capital investment in a real-time infrastructure, which requires a high level of user adoption to recover costs. The average villager cannot even afford a personal communications device such as a telephone or computer, let alone a subscription fee for access to the communications infrastructure. Hence, to recover cost, users must share the communications infrastructure. This limits the all-important value added from network effects. A villager who finds no use for a phone is typical, and this is perhaps why so few of the world's poor have used a telephone.

The real-time aspect of telephony can also be a disadvantage: Both intended parties must be present at each terminal to capture the infrastructure's full value. If a caller wishes to contact someone who does not own (or is not present at) a telephone, the communication is asynchronous despite the real-time infrastructure. Some kind of additional messaging mechanism (be it a messenger or an answering machine) is required to deliver the caller's message to its destination.

As a consequence, real-time telephony can reinforce gaps among rural populations since it encourages users to communicate mainly with people who have private phone lines, typically those of higher economic status located in more urban areas. In the GrameenPhone initiative, women were chosen as the community operators to help reduce this effect, since it was socially acceptable for women to deliver messages to everyone in the village.

Until widespread private ownership of ICT devices becomes economically feasible for end users, it may be useful to consider non-real-time infrastructures and applications such as voice mail, e-mail, and electronic bulletin boards. Also known as store-and-forward or asynchronous modes of communication, these technologies can be significantly lower in cost and do not necessarily sacrifice the functionality required to deliver valuable user services. They might also be more practical and socially appropriate for users than a shared real-time communications infrastructure.

***Myth:* Poor people don't need computers.**
***Truth:* The poor not only need digital services, but they are willing and able to pay for them to offset the much higher costs of poor transportation, unfair pricing, and corruption.**

Some rural service providers (RSPs) have achieved profitability by offering lower-cost substitutes for a villager's existing information, communication, and transportation expenses. For instance, Drishtee (www.drishtee.com) provides an e-government platform that lets villagers interact with local government offices remotely from a kiosk in their village that is managed by a trained operator. A variety of services such as filing a complaint, applying for a loan, and requesting a driver's license are generating up to $2,000 per year per kiosk for Drishtee.[1] The significant demand for these services results from a sound value proposition: Save villagers time and money. According to a villager who filed a complaint using a Drishtee kiosk,

> A visit to Sirsa costs Rs 50 [for travel], plus I waste a day. I will happily give Rs 10, even Rs 30 at the telecenter [kiosk] if I can save this.

Drishtee's success suggests that the introduction of ICT in

Even a single vehicle passing by a village once per day is sufficient to provide daily information services. The connection quality is also high. Although DakNet does not provide real-time data transport, a significant amount of data can move at once—typically 20 Mbytes in each direction. Indeed, physically transporting data from village to village by this means generally provides a higher data throughput than is typical with other low-bandwidth technologies such as a telephone modem.

Seamless scalability

In addition to its tremendous cost reduction, a critical feature of DakNet is its ability to provide a seamless method of upgrading to always-on broadband connectivity. As a village increases its economic means, its inhabitants can use the same hardware, software, and user interface to enjoy real-time information access. The only change is the addition of fixed-location wireless antennas and towers—a change that is entirely transparent to end users because they need not learn any new skills or buy any new hardware or software. The addition of fixed transceivers would provide real-time connectivity, thus enabling new, more sophisticated services, such as voice over IP, which allows "normal" real-time telephony.

Thus, as the "Some Common Myths about Rural Information and Communication Technology" sidebar describes, asynchronous broadband wireless connectivity offers a practical stepping-stone and migration path to always-on, broadband infrastructure and end-user applications. Together with rural areas might not have anything to do with technology per se. Much rural ICT starts with a specific technology and then tests out a variety of information and communication services to see which get accepted (a push approach). A better strategy might be to start with a basic service—in Drishtee's case, aggregating demand and brokering information exchange between the villager and the government—and then see how technology can support and streamline that service. Drishtee determined that computers and available connectivity were enough to capture, send, and receive information electronically.

Like other RSPs, however, Drishtee is constrained by India's lack of a viable communications infrastructure. Many of the villages that Drishtee operates in lack working phone lines because of poor line maintenance and delayed installations. As a result, Drishtee has resorted to "sneaker net," an asynchronous approach to connectivity that involves transporting and swapping floppy disks from the village to the government center and back again. Despite this labor-intensive approach, sneaker net is successful because Drishtee's applications that generate the most revenue require only intermittent connectivity.

Myth: Connectivity must be real time.
Truth: Asynchronous ICT services are sufficient to meet most rural community needs.

The Sustainable Access for Rural India (SARI) project in Tamil Nadu, India—a joint endeavor by the MIT Media Lab, the Harvard Center for International Development, and the Indian Institute of Technology, Madras—recently collected data about the communications needs, habits, and costs in hundreds of rural Indian households to gauge the desire for and perceived affordability of household communications.[2] The study found that the current market for successful rural ICT services does not appear to rely on real-time connectivity, but rather on affordability and basic interactivity:

> [Rural ICT companies] should start their operations by first focusing on providing basic communication and information services rather than more sophisticated applications.

Another SARI analysis done by McKinsey Consulting[3] indicates that although the universe of potential applications is large, "in the short-term only e-mail, scan-mail, voice-over-e-mail and chat are likely to be revenue-generating applications."

The McKinsey report also found that most of SARI's applications do not require real-time connectivity. It estimates that 50 percent of all existing rural mail will convert to e-mail, and people often preferred voice messaging to a real-time voice channel. Both e-mail and voice messaging are non-real-time applications.

In addition to these non-real-time applications, providers can use asynchronous modes of communication to create local information repositories that community members can add to and query. For example, a villager can access information from a computer somewhere outside the community and store that information in a village repository so that others can use it. This approach is particularly viable because the cost of digital storage is decreasing faster than the cost of most communication technologies.

Moreover, users are apt to find the information in a local repository highly relevant, which further decreases their reliance on a real-time infrastructure and international bandwidth. Users could search and browse the Web in non-real time through applications developed for low-connectivity environments such as TEK (http://tek.sourceforge.net/).

References

1. Boston Consulting Group, "Drishtee Case Study," Mar. 2002; www.digitalpartners.org/drishtee.html.
2. C. Blattman, R. Jensen, and R. Roman, "Assessing the Need and Potential of Community Networking for Developing Countries: A Case Study from India," Feb. 2002; http://edevelopment.media.mit.edu/SARI/papers/CommunityNetworking.pdf.
3. A. Pentland, R. Fletcher, and A.A. Hasson, "A Road to Universal Broadband Connectivity;" http://thinkcycle.org/tc-filesystem/?folder_id=37675.

the development of two other key rural communication components—robust, low-cost terminals and local user-interface design and applications—DakNet makes it practical for individual households and private users to get connected.

Economics

A back-of-the-envelope calculation for DakNet suggests that a capital investment of $15 million could equip each of India's 50,000 rural buses with a $300 MAP and thereby provide mobile ad hoc connectivity to most of the 750 million people in rural India. This figure represents a cost that is orders of magnitude lower than other rural communication alternatives.

Costs for the interactive user devices that DakNet supports—including thin-client terminals, PDAs, and VoIP telephones—may also soon become far more affordable than traditional PCs or WLL equipment. PDA-like devices using an IEEE 802-like wireless protocol retail for $100, with a manufacturing cost of approximately $50 (www.cybiko.com). System-on-a-chip technology is lowering these costs even more, potentially enabling wireless PDAs at prices as low as $25 (www.mobilesolve.com).

DAKNET IN ACTION

Villages in India and northern Cambodia are actively using DakNet with good results. As Figure 3 shows, local entrepreneurs currently are using DakNet connections to make e-services like e-mail and voice mail available to residents in rural villages.

One of DakNet's earliest deployments was as

Figure 3. A local entrepreneur using a DakNet connection to sell e-services like voice mail and e-mail to residents in a rural Indian village.

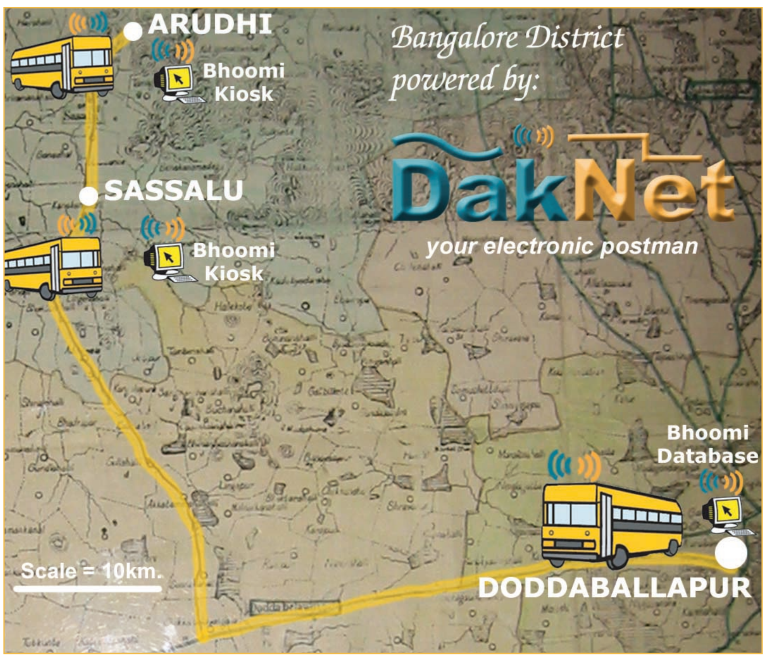


Figure 4. DakNet support of the Bhoomi e-governance project. Map shows route between Bhoomi database and village kiosks.

an affordable rural connectivity solution for the Bhoomi e-governance project. In September 2003, we also implemented DakNet in a remote province of Cambodia for 15 solar-powered village schools, telemedicine clinics, and a governor's office.

Bhoomi initiative in India

Bhoomi, an initiative to computerize land records, is recognized as the first national e-governance initiative in India. Pioneered by the State Government of Karnataka, Bhoomi has been successfully implemented at district headquarters across the state to completely replace the physical land records system.

As Figure 4 shows, DakNet makes Bhoomi's land records database available to villages up to 40 km away from Bhoomi's district headquarters, or "taluka," in Doddaballapur. In this deployment, we outfitted a public government bus with a DakNet MAP to transport land record requests from each village kiosk to the taluka server. The server processes requests and outputs land records. The bus then delivers the records to each village kiosk, where the kiosk manager prints them out and collects a payment of 15 rupees (US$0.32) per land record. The bus passes by the hub and stops at each village six times per day (three round-trips).

A "session" occurs each time the bus comes within range of a kiosk and the MAP transfers data. The average length of a session is 2 minutes and 34 seconds, during which the MAP transfers an average of 20.9 Mbytes unidirectionally (kiosk to MAP or MAP to kiosk) and up to twice that amount bidirectionally (from kiosk to MAP and MAP to kiosk). The average "goodput" (actual data throughput) for a session, during which the MAP and kiosk go in and out of connection because of mobility and obstructions, is 2.47 Mbps. These averages are based on repetitive testing in a sample group of villages that reflect the range of different antenna configurations. The team used both omnidirectional and directional antennas with differing gains according to the orientation of each kiosk with the road and the bus stop.

The total cost of the DakNet MAP equipment used on the bus is $580, which includes

- a custom embedded PC running Linux with 802.11b wireless card and 512 Mbytes of compact flash memory;
- a 100-mW amplifier, cabling, mounting equipment, and a 14-in omnidirectional antenna; and
- an uninterruptible power supply powered by the bus battery.

The average total cost of the equipment used to make a village kiosk or hub DakNet-ready was $185. Assuming that each bus can provide connectivity to approximately 10 villages, the average cost of enabling each village was $243 ($185 at each village plus $580 MAP cost for 10 villages).

Villagers along the bus route have enthusiastically received the DakNet-Bhoomi system. They are grateful to avoid making the long, expensive trip into the main city to obtain land records.

Figure 5. The Internet Motoman project in Cambodia. (a) The main hospital, with its VSAT connection to the Internet, acts as the hub. (b) Because the roads are so bad during rainy periods, MAP-enabled Honda motorcycles are used to connect schools to the hub. (c) For locations with particularly challenging terrain, there is even a MAP-equipped ox cart.

E-mail for Cambodian schools

CambodiaSchools.com operates 225 rural schools throughout Cambodia with funding from private donors and the World Bank. Our aim was to provide students with Internet access by providing asynchronous connectivity to the backbone or hub—a satellite dish in the provincial capital of Ban Lung, which has a 256-Kbyte-per-second link.

As in the Bhoomi project, we used public transportation, but, as Figure 5 shows, the terrain in northern Cambodia is so difficult that we had to place MAPs on Honda motorcycles instead of buses. For one particularly remote area, we even affixed the MAP to an ox cart. The results of the project, which we dubbed the Internet Village Motoman, were once again gratifying. For the first time, students in these Cambodian schools could send e-mail, request Web pages, and feel connected to the rest of the world.

DakNet's low deployment cost and its enthusiastic reception by rural users has motivated dozens of inquiries for further deployments. We are already working on versions for Nigeria, Jordan, and Colombia, as well as developing plans for offering a turnkey solution that will let users deploy DakNet themselves (www.firstmilesolutions.com). This should provide millions of people their first possibility for digital connectivity, and, as study after study has shown, increasing connectivity is the most reliable way to encourage economic growth.

The larger goal is to shift the policy focus of the government's universal-service-obligation funds from wireline village telephones to wireless ad hoc networking. The shift will probably require formal assessment of user satisfaction, resulting economic growth, and of course system reliability. If we can clear these bureaucratic hurdles, however, governments might be able to connect the world's poor to the Internet far sooner than anyone believed possible. ■

***Alex (Sandy) Pentland** is a Toshiba Professor of Media Arts and Sciences at MIT and cofounder of MIT Media Laboratory's Digital Nations consortium, the Media Lab Asia in India, the LINCOS project in Costa Rica, and the Center for Future Health. His work encompasses wearable computing, communications technology for developing countries, human-machine interfaces, artificial intelligence, and machine perception. Pentland is also a cofounder of the IEEE Computer Society's Wearable Information Systems Technical Committee, and he has won numerous awards in the arts, engineering, and sciences. Contact him at pentland@media.mit.edu.*

***Richard Fletcher** is a cofounder of First Mile Solutions. He received a PhD in low-cost wireless electronics and sensors from the MIT Media Lab and has three patents, with seven more pending. His research projects other than DakNet include low-cost radio frequency identification tags and satellite-linked environmental probes for Mt. Everest, Antarctica, and Costa Rica. Contact him at fletcher@firstmilesolutions.com.*

***Amir Hasson** is a cofounder of First Mile Solutions, specializing in IT and Web consulting and marketing. Hasson received an MS in the management of technology from the MIT Sloan School of Management and has spent the past year creating and deploying WiFi networks in developing countries. Contact him at amir@firstmilesolutions.com.*

Combining Optical Holograms with Interactive Computer Graphics

Merging optical holograms with 3D graphical elements can provide an acceptable tradeoff between quality and interactivity: The holographic data provides high-quality but static content, while additional graphical information can be generated, inserted, modified, and animated at interactive rates.

Oliver Bimber
Bauhaus University

Today, many applications for optical holograms exist, including interferometry, copy protection, data storage, and holographic optical elements. A hologram is a photometric emulsion that records interference patterns of coherent light. The recording itself stores the amplitude, wavelength, and phase information of light waves. In contrast to simple photographs, which can record only amplitude and wavelength information, holograms can reconstruct complete optical wavefronts. This results in the captured scenery having a three-dimensional appearance that can be observed from different perspectives.

Museum exhibits often use optical holograms because they can present 3D objects with almost no loss in visual quality. Displaying artifacts virtually removes the need to build physical replicas of the original objects. In addition, this holographic technology can be used to make medical, dental, archaeological, and other recordings—both for teaching and documentation.

Optical holograms are static, however, and lack interactivity. Multiplex holograms offer an apparent exception. Built from multiple vertical-strip holograms that contain recordings of the same scenery at different time intervals, they let observers perceive the recorded scene in motion while moving around the hologram or by spinning a cylindrically shaped version around its principal axis. Multiplex holograms are not, however, truly interactive.

Combining 3D computer graphics with stereoscopic presentation techniques provides an alternative that allows interactivity. Although state-of-the-art rendering methods and graphics hardware can produce realistic images at interactive rates, they do not approach the quality and realism of holographic recordings.

Autostereoscopic displays allow for glass-free observation of computer-generated scenes. These displays can present several perspective views at one time, thus supporting multiple users simultaneously. Resolution and rendering speed, however, decrease with the number of views generated. Holographic images, in contrast, can provide all depth cues—perspective, binocular disparity, motion parallax, convergence, and accommodation—and theoretically can be viewed simultaneously from an unlimited number of positions.

PRINTING AND RENDERING

The two main types of computer-generated holograms, *digital holography* and *electroholography*, have become the focus of several groups researching applications for them.

Digital holography[1] uses holographic printers to sequentially expose small fractions of the photo-

Holography and Autostereoscopy

Both of the two basic optical hologram types—*transmission* and *reflection*—are reconstructed by illuminating them with monochromatic light, as described in Figure A. This approach requires that the light hit the emulsion at the same angle as the reference laser beam used to record the hologram.

Transmission and reflection holograms

To view a transmission hologram, the light source and observer must be on *opposite* sides of the holographic plate. The light is transmitted through the plate before it reaches the observer's eyes. Those portions of the emulsion not recorded or illuminated remain transparent.

To view a reflection hologram, the light source and observer must be on the *same* side of the holographic plate. The light reflects from the plate toward the observer's eyes. As with transmission holograms, unrecorded or nonilluminated portions of the emulsion remain transparent.

These two hologram types have generated a spectrum of variations. Although some holograms can be reconstructed only with laser light, others can be viewed only under white light. Rainbow holograms, one of the most popular white-light transmission hologram types, diffract each wavelength of the light through a different angle. This lets viewers observe the recorded scene from different horizontal viewing positions but also makes the scene appear in different colors when observed from different points.

In contrast to rainbow holograms, white-light reflection holograms can provide full parallax and display the recorded scene in a consistent—but in most cases monochrome—color for different viewing positions.

It is possible to produce both transmission and reflection variations of color white-light holograms. Usually, this process requires recording the same content on several emulsion layers while exposing each layer to laser light with a different wavelength. When reconstructed, each object wave from each layer contributes its individual wavelength, merging together into a multicolor image.

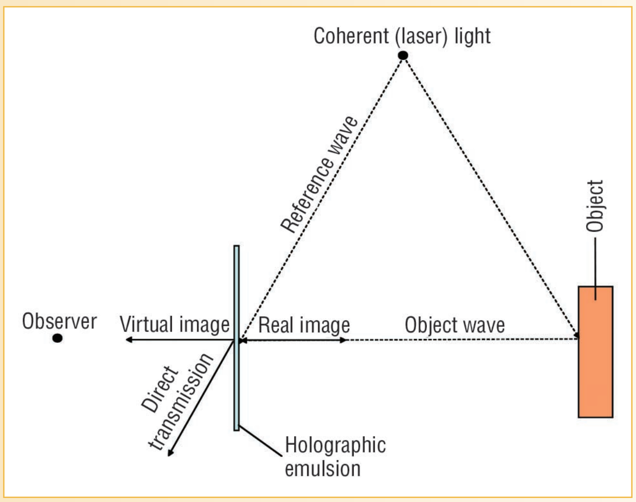


Figure A. Optical holographic recording and reconstruction. A laser beam can be split into two identical beams. One beam, the reference wave, illuminates the holographic emulsion directly while the other beam illuminates the object to be recorded. If the emulsion is illuminated with a copy of the reference wave, it interacts with the recorded interference fringes and reconstructs the object wave, which is visible to the observer.

Parallax displays

These CRT or LCD display screens are overlaid with an array of light-directing or light-blocking elements. Using these elements, the display directs emitted light to both eyes differently—allowing each eye to see individual portions of the displayed image. The observer's visual system interprets corresponding light rays as being emitted by the same spatial point. Dividing the screen space into left and right image portions allows for a glass-free separation of stereo pairs into two or more viewing zones.

Some displays control the parallax array mechanically or electronically, depending on the observer's viewpoint, to direct the viewing zones more precisely toward the eyes. Others generate many dense viewing zones—each showing a slightly different perspective of the rendered scene at the same time. Such displays support multiple users simultaneously but do not yet allow for high frame rates.

One parallax display type includes a barrier that applies an array of light-blocking elements—for example, a light-blocking film or liquid crystal barrier—in front of a screen. The light-blocking elements cover portions of the screen for one eye that are visible from the other eye.

Another parallax example, a lenticular-sheet display, utilizes the refraction of a lens array—consisting of small, cylindrical, prismlike or spherical lenses—to direct the light into the different viewing zones. Images generated with lenticular-sheet displays appear brighter than those displayed on barrier displays. Although prisms and cylinders provide a horizontal parallax only, spherical lenses support a full parallax.

metric emulsion with a computer-generated image. This process results in conventional holograms that display computer-generated content. This technique also can be used to construct large-scale, tiled holograms.[2] Although digital holograms can be multiplexed to display scenes in motion, they remain noninteractive.

Electroholography facilitates the computer-based generation and display of holograms in real time.[3,4] Holographic fringes can be computed by either

- rendering multiple perspective images, then combining them into a stereogram;[5] or
- simulating the optical interference and calculating the interference pattern.[6]

Once computed, the system dynamically visualizes the fringes with a holographic display. Since creating a hologram requires processing, transmitting, and storing a massive amount of data, today's computer technology still sets electroholography's limits. To overcome some of these performance issues, researchers have developed advanced reduction and compression methods that create truly interactive electroholograms. Unfortunately, most of these holograms are small, low resolution, and monochrome. However, recent advances in consumer graphics hardware may reveal potential acceleration possibilities that can overcome these limitations.[7]

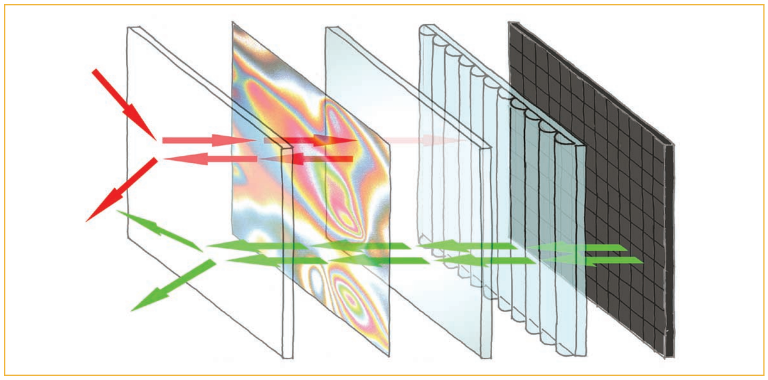


Figure 1. Optical functioning. The explosion model of the optical layers' stacked structure shows, from left to right: glass protection, holographic emulsion, mirror beam splitter for transmission holograms only, lenticular-lens sheet, and LCD array. Reflected light rays (red arrows) reconstruct the object wave on their return through the emulsion. Stereoscopic images (green arrows) pass through all layers until they merge with the hologram.

PARTIALLY RECONSTRUCTING OBJECT WAVES

Combining optical holograms with 2D or 3D graphical elements can provide an acceptable trade-off between quality and interactivity. While the holographic content provides high-quality but static content—as described in the "Holography and Autostereoscopy" sidebar—the combined technology can generate, insert, modify, and animate additional graphical information at interactive rates.

Technically, researchers can use optical combiners such as mirror beam splitters or semitransparent screens to visually overlay the output rendered on a screen over a holographic plate. However, the hologram's reconstructed light will interfere with the overlaid light of the rendered graphics, making an effective combination impossible.

To solve this problem, researchers can reconstruct the object wave only partially, leaving gaps at those places where they have inserted graphical elements. Doing this requires a point light source capable of selectively emitting light in different directions to create an incomplete reference wave. Conventional video projectors provide such light sources and are well suited to viewing white-light reflection or transmission holograms because today's high-intensity discharge lamps can produce a very bright light.

If we use autostereoscopic displays, such as parallax displays, to render 3D graphics registered to the hologram, both holographic and graphical content appear three-dimensional within the same space. This effect can also be achieved by using stereoscopic displays with special glasses that separate the stereo images.

Both reflection holograms, which lack an opaque backing layer, and transmission holograms remain transparent if not illuminated. Thus, they can serve as optical combiners themselves—leading to very compact displays. The illumination and rendering techniques work the same for both hologram types.

Figure 1 shows how a transmission hologram can be combined effectively with a flat-panel lenticular-lens sheet display—a variation of a parallax display that utilizes the refraction of a lens array to direct light into the different viewing zones.

Placing a transmission hologram in front of a mirror beam splitter illuminates it from the front and augments it with graphics from the back. Reflection holograms do not need this beam splitter.

A thin glass plate protects the emulsion from being damaged and keeps it flat to prevent optical distortion. The lenticular-lens sheet directs the light emitted from the LCD array through all layers toward the observer's eyes. The projected light is transmitted through the first two layers and partially reflected back—either by the beam splitter in combination with a transmission hologram or by a reflection hologram—to reconstruct the recorded content. The screen mostly absorbs the remaining portion of light transmitted through all layers. Figure 2 shows how the system computes the selective illumination on the holographic plate to reconstruct the portion of the hologram not occluded by graphics.

Assuming that information about the holographic content's depth and a description of the graphical content area are available, researchers can use conventional graphics hardware for ren-

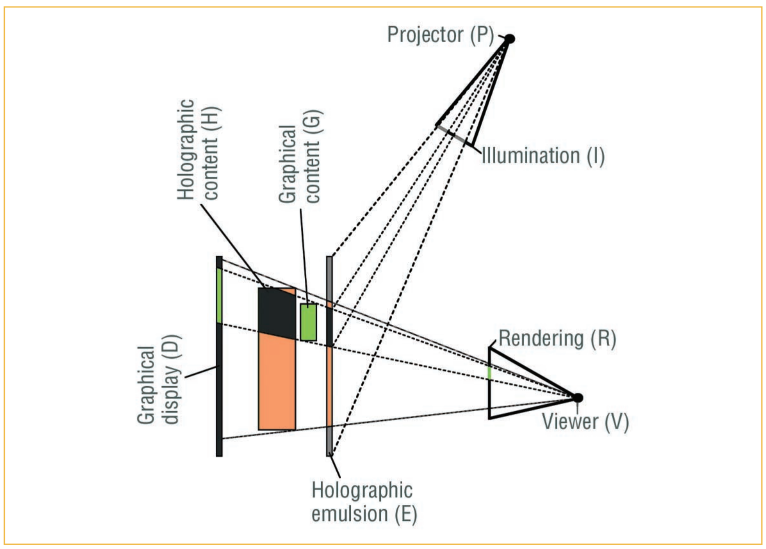


Figure 2. Conceptual sketch of the display constellation. Colored areas on the graphical display and holographic emulsion show which visible image areas constitute the hologram (red) and which constitute the graphics (green).

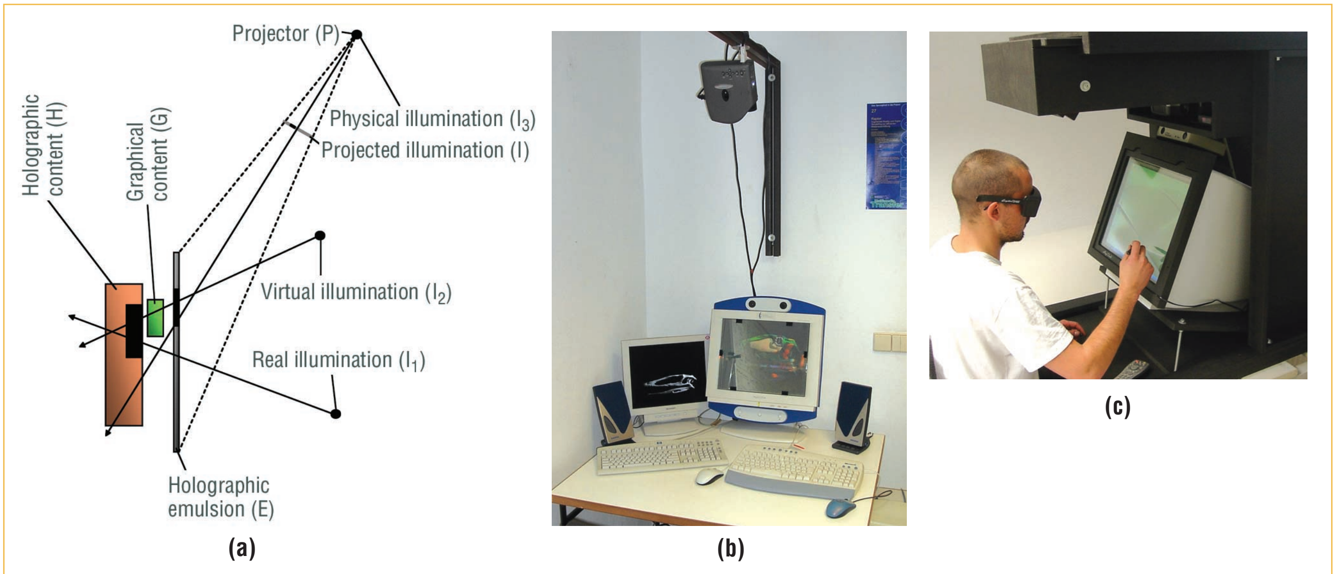


Figure 3. Light-interaction and proof-of-concept prototypes. (a) To simulate virtual shading and shadow effects on the holographic content, the recorded and physical illumination effects must first be neutralized. (b) Autostereoscopic prototype with parallax display and head-finder. (c) Active stereoscopic prototype with CRT monitor, infrared tracking, and touch screen.

dering and illumination. Fortunately, the depth information required to approximate the holographic content's surface can be very coarse. For this example, it consists of a mesh with 5,000 triangles that totals 44.4 Kbytes. This image does not come close to approaching the quality of its optical counterpart but can be rendered easily in real time.

The system geometrically aligns this content during an offline registered step. If the recorded hologram includes optical markers with the actual content, cameras can automatically perform this registration. In addition, the researchers must perform an offline calibration to determine the video projector's extrinsic and intrinsic parameters with respect to the holographic emulsion. If it is mechanically possible to mount the holographic emulsion close to the graphical display, these two planes can be considered identical for the rendering algorithm.

To partially reconstruct the object wave, the system first creates an intermediate texture image by rendering the holographic content from the viewer into the graphics card's depth buffer and filling the card's frame buffer entirely with the lighting source's predefined color values. In addition, the system renders the graphical scene description into depth and stencil buffers, then it clears the frame buffer's stenciled areas and copies the result into the memory block allocated for the intermediate texture image.

If the graphics card provides a render-to-texture option, the read-back operation from the frame buffer into texture memory can be bypassed. The system renders the final illumination image from the projector by drawing the holographic emulsion into the frame buffer and texturing its geometry with the intermediate texture image. The projector then beams the illumination image onto the holographic emulsion.

Next, the system generates a rendering image from the viewer over the off-axis of the graphical display by rendering the content's depth information into the depth buffer and the graphical content's scene description into the depth and frame buffers. The graphical display then shows the rendering image.

LIGHT INTERACTION

The reconstructed object wave's amplitude is proportional to the reference wave's intensity. Besides using an incomplete reference wave for reconstructing a fraction of the hologram, intensity variations of the projected light permit local modification of the recorded object wave's amplitude.

Practically, this means that to create the illumination image, the system uses shading and shadowing techniques to render the holographic content instead of rendering it with a uniform intensity. To do this, the shading effects caused by the real light sources used for illumination during hologram recording, as well as the physical lighting effects caused by the video projector on the holographic plate, must both be neutralized. Next, the influence of a synthetic illumination must be simulated. This can also be done with conventional graphics hardware, as Figure 3a shows. Three intensity images must be rendered.

For the first image, the system renders the holographic content's depth information from the desired relationship between the viewer and the emulsion, using a white diffuse material factor and graphical light sources that generate approximately the same shading and shadow effects as the real light sources used during the holographic recording process. This results in an intermediate texture. The system generates the first image by rendering the holographic emulsion from the perspective of the video projector and texturing it with the intermediate texture. This image simulates the intensity of the recorded object wave.

The system repeats the process to create the second image, this time using graphical light sources to shade the holographic content under the new, virtual-lighting situation. The ratio of the second image to

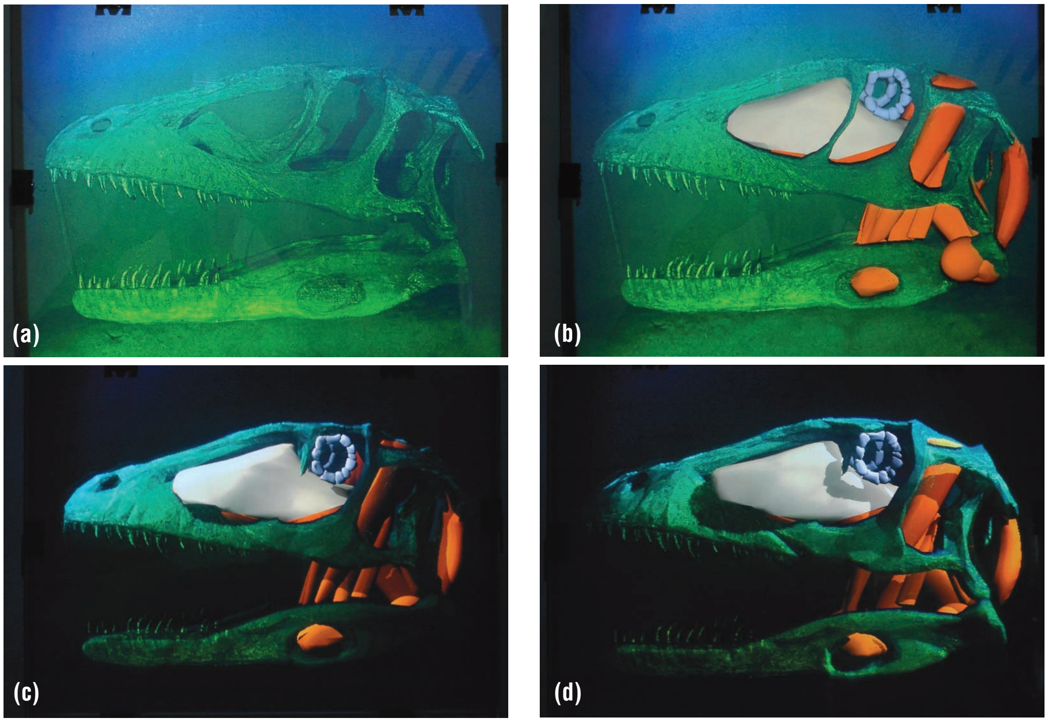


Figure 4. Rainbow hologram of a dinosaur skull combined with 3D graphical elements and synthetic shading effects. (a) The holographic plate illuminated with a uniform light. (b) The plate illuminated only at the portions not occluded by graphical elements such as muscles and other soft tissue. (c) The virtual light source located at the top-left corner, in front of the display. (d) The virtual light source placed at the top-right corner, in front of the display.

the first image represents the required intensity of the reference wave for the holographic plate's emulsion.

For the third image, the system renders the geometry of the holographic emulsion from the projector with a white diffuse material factor and a virtual point light source located at the projector's position. This intensity image represents the geometric relationship between the holographic plate and the video projector as a physical point light source. This third image contains form factor components, such as the square-distance attenuation and angular correlation of the projected light onto the holographic plate, and it neutralizes the physical effects of the projector itself.

The final illumination image can be computed in real time by dividing the second image by the first image and then by the third image via pixel shades. The projection of the resulting image onto the holographic emulsion will neutralize the physical and recorded illumination effects as much as possible and create new shadings and shadows based on the virtual illumination. Again, the system must stencil out the appearance of the graphical content in the final image before displaying it.

During all illumination and rendering steps, the system uses hardware-accelerated shadow-mapping techniques to simulate real and virtual shadow effects on the holographic content's depth information and the graphical scene description. Finally, the system can cast synthetic shadows correctly from all holographic and graphical elements onto all other elements.

This technique's capabilities are clearly limited. It produces acceptable results if the recorded scenery has been illuminated well while making the hologram. However, it cannot neutralize recorded shadows and extreme shading differences. Further, it cannot cancel out recorded color, reflections, and higher-order optical effects.

PROVING THE CONCEPT

The autostereoscopic prototype in Figure 3b and the stereoscopic desktop prototype in Figure 3c were built to validate the proposed techniques. These prototypes consist entirely of off-the-shelf components, including either a lenticular-lens-sheet display with integrated head-finder for wireless user tracking or a conventional CRT screen with active stereo glasses, wireless infrared tracking, and a touch screen for interaction. Both prototypes use digital light projectors. A single PC with a dual-output graphics card renders the graphical content on the screen and illuminates the holographic plate on the video projector.

In both cases, the screen additionally holds the front layers—glass protection, holographic emulsion, and optional mirror beam splitter. The remaining two layers shown in Figure 1—the lenticular-lens sheet and LCD array—already form part of the autostereoscopic display. The display shown in Figure 3c does not need them because the system uses shutter glasses to separate the images for the left and right eye.

A rainbow transmission hologram and a reflection hologram of a dinosaur skull were recorded with a 527.5-nm green laser. Figure 4a shows a photograph of the entire reconstructed hologram, illuminated with a projected uniform light. The system generates illumination and stereoscopic images so that the graphical and holographic content can

Figure 5. Display variations. (a) Desktop display that can be used in a light-box fashion, with a special input device allowing interaction and providing haptic feedback of holographic and graphical content. (b) Wall-mounted display in a museum environment, with a ceiling-mounted video projector replacing conventional spotlights. An integrated head-finder alerts the display when observers stand in front of it.

be merged within the same space, as Figure 4b shows. Projecting an intensity image that contains new shading and shadow effects instead of a uniform illumination lets the system neutralize most of the diffuse shading recorded in the hologram and produced by the projector. The system can then consistently illuminate the holographic and graphical content, creating matching shading and shadow effects, under novel lighting.

Figures 4c and 4d show the synthetic shading effects caused by a virtual light source. In addition, the image shows virtual shadows cast correctly between hologram and graphical elements. Note that none of the photographs have been retouched. When capturing these images, the system rendered the graphics monoscopically. Although Figure 4 shows the results with a monochrome transmission hologram, reflection and color holograms can achieve the same effects.

Given that hardware-accelerated consumer graphics cards support all these rendering techniques, including shadow mapping and shading, using this method can easily achieve interactive frame rates.

INDUSTRY, SCIENCE, AND EDUCATION

Optical holograms can store a massive amount of information on a thin holographic emulsion. This technology can record and reconstruct a 3D scene with almost no loss in quality. For example, the monochrome hologram shown in Figure 4 comprises a theoretical information content of approximately 96.42 Gbytes. Higher-quality holograms can store a fringe pattern that exceeds one terabyte. Reaching the same quality and performance with computer graphics would require rendering the entire data set in real time on a 1.08 terapixel display—equivalent to 9 million points per square millimeter on a 40×30 cm panel. For a color hologram of the same size, the information content multiplies. Moore's law—which asserts that computing power doubles every 18 months—must be applied many times for graphical or electroholographic rendering techniques and displays to reach this quality at interactive frame rates.

A combination of interactive computer graphics and high-quality optical holograms represents an alternative that can be realized today with off-the-shelf consumer hardware. Several commercial online services already offer uncomplicated and inexpensive ways to create color holograms from a set of images or video clips. With this technology, users can create holograms with almost any content—even outdoor scenes.

Archaeologists, for example, already use optical holograms to archive and investigate ancient artifacts.[8,9] Scientists can use hologram copies to perform their research without having access to the original artifacts or settling for inaccurate replicas. They can combine these holograms with interactive computer graphics to integrate real-time simulation data or perform experiments that require direct user interaction, such as packing reconstructed soft tissue into a fossilized dinosaur skull hologram. In addition, specialized interaction devices can simulate haptic feedback of holographic and graphical content while scientists are performing these interactive tasks. An entire collection of artifacts will fit into a single album of holographic recordings, while a light-box-like display such as that used for viewing x-rays can be used for visualization and interaction, as Figure 5a shows.

This approach has the potential for wide application in other industries. In the automotive industry, for example, complex computer models of cars and components often lack realism or interactivity. Instead of attempting to achieve high visual quality and interactive frame rates for the entire model, designers could decompose the model into sets of interactive and static elements. The system could record physical counterparts of static elements in a hologram with maximum realism and release computational resources to render the interactive elements with a higher quality and increased frame rate. Multiplexing the holographic content also lets users observe and interact with the entire model from multiple perspectives.

Augmenting optical holograms in museums with animated multimedia content lets exhibitors communicate information about the artifact with more excitement and effectiveness than text labels offer. Such displays can also respond to user interaction. Because wall-mounted variations like the one in Figure 5b require little space, museums can display a larger number of artifacts.

Clearly, holograms or other replicas cannot substitute for original artifacts because viewing those originals is the main reason patrons visit a museum. If, however, a unique artifact is unavailable or too fragile to be displayed, a hologram offers an enticing alternative by showing the artifact as a high-quality, 3D image that, combined with computer graphics, lets users experience it interactively.

We can use this concept to develop a palette of display variations. For example, with only minor changes to the presented rendering techniques, arbitrarily curved holograms such as the cylindrical shapes used for multiplex holograms can be supported instead of only simple planar plates. Even without graphical augmentations, projector-based illumination alone has several potential applications. In combination with optical or digital holograms, it can be used to create visual effects. Certain portions of a hologram, for example, can be made temporarily invisible while others can be highlighted.

Emerging large-scale autostereoscopic displays and existing stereoscopic projection screens will let designers scale up this proposed concept. Not only the display, but also the holograms themselves, can be composed from multiple smaller tiles to reach large dimensions and high resolutions.

The foundations of future 3D displays may have been laid in the late 1940s by Dennis Gabor, who, 20 years later, received the Nobel Prize in physics for his invention of holography. Large interactive electroholographic displays with high resolution and full color would provide the ultimate displays—realizing what today is only possible in science fiction. Technological advances will pave the road toward making this fantasy a reality. Intermediate solutions, such as merging optical holograms with 3D graphical elements, represent the first steps along this path. ■

Acknowledgments

Thanks to Tim Frieb and Holowood Holografiezentrum Bamberg for their technical support and to Matthias Hanzlik for the illustrations in Figure 5.

References

1. M. Yamaguchi, N. Ohyama, and T. Honda, "Holographic 3-D Printer," *Proc. SPIE*, vol. 1212, Springer, 1990, p. 84.
2. M.A. Klug, "Scalable Digital Holographic Displays," *Proc. Image Processing, Image Quality, Image Capture Systems Conf.* (IS&T PICS 2001), Soc. for Imaging Science and Technology, 2001, pp. 26-32.
3. J.S. Kollin, S.A. Benton, and M.L. Jepsen, "Real-Time Display of 3-D Computed Holograms by Scanning the Image of an Acousto-Optic Modulator," *Proc. SPIE*, vol. 1136, *Holographic Optics II: Principles and Applications*, Springer, 1989, pp. 178-185.
4. M. Lucente, "Interactive Three-Dimensional Holographic Displays: Seeing the Future in Depth," *Siggraph Computer Graphics*, special issue on Current, New, and Emerging Display Systems, ACM Press, 1997, pp. 63-67.
5. M. Lucente and A. Tinsley, "Rendering Interactive Images," *Proc. Siggraph 95*, ACM Press, 1995, pp. 387-394.
6. M. Lucente, "Interactive Computation of Holograms Using a Look-Up Table," *J. Electronic Imaging*, vol. 2, no. 1, 1993, pp. 28-34.
7. C. Petz and M. Magnor, "Fast Hologram Synthesis for 3D Geometry Models Using Graphics Hardware," *Proc. SPIE 03, Practical Holography XVII and Holographic Materials IX*, vol. 5005, Springer, 2003, pp. 266-275.
8. F. Dreesen and G. von Bally, "Color Holography in a Single Layer for Documentation and Analysis of Cultural Heritage," *Optics within Life Sciences* (OWLS IV), Springer, 1997, pp. 79-82.
9. F. Dreesen, H. Deleré, and G. von Bally, "High-Resolution Color Holography for Archaeological and Medical Applications," *Optics within Life Sciences* (OWLS V), Springer, 2000, pp. 349-352.

Oliver Bimber *is a junior professor for augmented reality at Bauhaus University, Weimar, Germany. His research interests include display technologies, computer graphics, and human-computer interaction. Bimber received a PhD in engineering from Darmstadt University of Technology, Germany. Contact him at bimber@ieee.org.*

CALLS FOR IEEE CS PUBLICATIONS

Over the past ten years, Internet service providers have built out their networks to cope with what they perceive as steadily increasing user demands. Because of that rapid development, network measurement has tended to have lower priority than immediate network operations, deployment, and upgrade concerns.

For a 2004 special issue on measuring performance, *IEEE Internet Computing* invites researchers and practitioners to submit original works on Internet measurement, especially studies that involve open source or freely available tools and data from wide area networks. Suitable topics include Internet topology, routing system behavior, understanding protocol behavior, monitoring/management of large-scale backbones, and quality of service monitoring.

Submissions are due **2 Feb.** The complete call for papers is available at www.computer.org/internet/call4ppr.htm#v8n5.

OTHER CALLS

MEMOCODE 2004, 2nd ACM/IEEE Conf. on Formal Methods & Programming Models for Codesign, 23-25 June, San Diego, Calif. Panel and tutorial proposals due **12 Feb.** www.irisa.fr/manifestations/2003/MEMOCODE/

NATW 2004, IEEE 13th North Atlantic Test Workshop, 13-14 May, Essex Junction, Vt. Papers due **7 Feb.** www.ee.duke.edu/NATW/

ICALT 2004, 4th IEEE Int'l Conf. on Advanced Learning Technologies, 30 Aug.-1 Sept., Joensuu, Finland. Papers due **13 Feb.** http://lttf.ieee.org/icalt2004

SRDS 2004, 23rd Symp. on Reliable Distributed Systems, 18-20 Oct., Florianópolis, Brazil. Submissions due **2 Apr.** www.srds2004.ufsc.br/

ISSRE 2004, 15th Int'l Symp. on Software Reliability Eng., 2-5 Nov., Saint-Malo, France. Abstracts due **2 Apr.,** submissions due **18 Apr.** www.issre.org/2004/

ATS 2004, 13th Asian Test Symp., 15-17 Nov., Kenting, Taiwan. Papers due **15 Apr.** http://ats04.ee.nthu.edu.tw/~ats04/

CALENDAR

FEBRUARY 2004

14-15 Feb: SLIP 2004, 6th Int'l Workshop on System-Level Interconnect Prediction, Paris. www.sliponline.org

14-18 Feb: HPCA-10, 10th Int'l Symp. on High-Performance Computer Architecture, Madrid, Spain. www.ac.uma.es/hpca10/

16-20 Feb: DATE 2004, Design, Automation & Test in Europe, Paris. www.date-conference.com/

19-20 Feb: ISVLSI 2004, Int'l Symp. on VLSI, Lafayette, LA. www.cacs.louisiana.edu/isvlsi/

23-25 Feb: Preclinical Development Forum, "Designing & Selecting Drug Candidates for Clinical Success," Cambridge, Mass. www.lifesciencesinfo.com/pdft

24-27 Feb: BAST 2004, 13th CRC-IEEE BAST Workshop, Santa Rosa, Calif. http://crc.stanford.edu/BAST/BAST.html

MARCH 2004

3-5 Mar: PRDC 2004, 10th Pacific Rim Int'l Symp. on Dependable Computing, Papeete, Tahiti. www.laas.fr/PRDC10

7-11 Mar: IEEE Infocom 2004, Hong Kong. www.ieee-infocom.org/ 2004/

10-12 Mar: ISPASS 2004, IEEE Int'l Symp. on Performance Analysis of Systems & Software, Austin, Texas. www.ispass.org/ispass2004/

14-17 Mar: PerCom 2004, IEEE Int'l Conf. on Pervasive Computing & Comm., Kissimmee, Fla. www.percom.org

20-24 Mar: CGO 2004, Int'l Symp. on Code Generation & Optimization, San Jose, Calif. www.cgo.org

23-25 Mar: WMTE 2004, 2nd IEEE Int'l Workshop on Wireless & Mobile Technologies in Education, Taoyuan, Taiwan. http://lttf.ieee.org/wmte2003/

23-26 Mar: ICDCS 2004, 24th Int'l Conf. on Distributed Computing Systems, Tokyo. www.cis.ohio-state.edu/icdcs04/

25-26 Mar: HASE 2004, IEEE 8th Int'l Symp. on High-Assurance Systems Eng., Tampa, Fla. http://hasrc.csee.wvu.edu/hase04/

27-31 Mar: VR 2004, IEEE Virtual Reality 2004 Conf. (with Haptics 2004), Chicago. www.VR2004.org

28-29 Mar: RIDE WS-ECEG 2004, 14th Int'l Workshop on Research Issues on Data Eng., Boston. http://europa.nvc.cs.vt.edu/ride04/

Submission Instructions

The Call and Calendar section lists conferences, symposia, and workshops that the IEEE Computer Society sponsors or cooperates in presenting. Complete instructions for submitting conference or call listings are available at www.computer.org/conferences/submission.htm.

A more complete listing of upcoming computer-related conferences is available at www.computer.org/conferences/.

28-30 Mar: SSIAI 2004, IEEE Southwest Symp. on Image Analysis & Interpretation, Lake Tahoe, Nev. www.ee.ttu.edu/Conferences/SSIAI2004

29-31 Mar: EEE 2004, IEEE Int'l Conf. on e-Technology, e-Commerce, & e-Service, Taiwan. http://bikmrdc.lm.fju.edu.tw/eee04

29-31 Mar: AINA 2004, 18th Int'l Conf. on Advanced Information Networking & Applications, Fukuoka, Japan. www.takilab.k.dendai.ac.jp/conf/aina/2004/

30 Mar.-2 Apr: ICDE 2004, 20th Int'l Conf. on Data Eng., Boston. www.cse.uconn.edu/icde04/

APRIL 2004

4-7 Apr: ITSW 2004, 11th IEEE Int'l Test Synthesis Workshop, Santa Barbara, Calif. www.tttc-itsw.org

5-7 Apr: ITCC 2004, 5th Int'l Conf. on IT, Las Vegas. www.itcc.info

14-16 Apr: COOL CHIPS VII, Int'l Symp. on Low-Power & High-Speed Chips, Yokohama, Japan. www.coolchips.org/

14-16 Apr: ICECCS 2004, 9th IEEE Int'l Conf. on Eng. Complex Computer Systems, Florence, Italy. www.dsi.unifi.it/iceccs04/

Call for Articles for *Computer*

Computer seeks articles for a special issue on sensor networks, to appear in August 2004. Guest editors are David Culler from the University of California, Berkeley; Mani Srivastava from the University of California, Los Angeles; and Deborah Estrin from the University of California, Los Angeles.

Distributed systems of embedded smart sensors and actuators promise unprecedented capabilities for the instrumentation and monitoring of the physical world. Sensors can monitor many types of information: temperature, vibration, air pressure, chemicals, even voice and video data. Proponents envision a host of novel applications for sensor networks, from earthquake structural failure analysis to rainforest habitat research. The small and many characteristics of sensor networks have fostered a wave of research innovation in disciplines that have previously been focused on computing and networking at a much larger scale. This interest is driven by the vision that, like the Internet, large-scale distributed networks of sensors will pervade the world but at a physical, rather than virtual, level.

Computer's special issue will focus on all aspects of the field: hardware, architectures, wireless communication, networking, middleware, application development, applications, and experience. There are opportunities for short and long papers, providing a forum for reporting on both early and mature research. *Computer* also is soliciting descriptions of sensor network technology in existing products and services.

Topics of particular interest include sensor network architectures; sensor node hardware; networking; low-power protocols and services; distributed algorithms; data query, dissemination, routing, and fusion; novel applications and services; application development tools; and deployment experiences.

The deadline for papers is **2 February 2004**. Submission guidelines are available at www.computer.org/computer/author.htm. **Submit manuscripts at http://cs-ieee.manuscriptcentral.com/.**

Send inquiries to the guest editors at culler@eecs.berkeley.edu, mbs@ee.ucla.edu, and destrin@cs.ucla.edu.

Computer seeks articles for a special issue on Internet data centers, to appear in November 2004. Guest editors are Krishna Kant from Intel and Prasant Mohapatra from the University of California, Davis.

Internet data centers form the backbone of most Internet-based services, including e-commerce, IP-based telecom services, hosting services, and the like. As the reach of the Internet widens and more business-critical services are offered, the demands on IDCs grow along multiple dimensions, including responsiveness, service differentiation, security, and availability. Many other forces are likely to affect how the data centers of the future are designed, provisioned, and operated.

Computer's special issue will focus on research issues in identifying and implementing new strategies for optimizing IDCs: application services, protocol enhancements, performance evaluations, provisions for adequate security, protection and isolation, and ensuring an adequate quality of service. *Computer* is soliciting a small number of high-quality papers from academia and industry that highlight various problems and solutions and provide a vision for future work in this area.

Topics of particular interest include system architecture and converged data centers; symmetric multiprocessors versus clustered systems; scalability, reliability, and fault tolerance; performance evaluation and workload characterization; operations, control, and autonomic management; power management issues; exploitation of new hardware/software technologies; and issues of security, protection, and isolation.

The deadline for papers is **1 April 2004**. Submission guidelines are available at www.computer.org/computer/author.htm. **Submit manuscripts at http://cs-ieee.manuscriptcentral.com/.**

Send inquiries to the guest editors at krishna.kant@intel.com and prasant@cs.ucdavis.edu.

18-21 Apr: DDECS 2004, 7th IEEE Workshop on Design & Diagnostics of Electronics Circuits & Systems Workshop, Tatranská Lomnica, Slovakia. www.ui.savba.sk/DDECS2004

19-22 Apr: CCGRID 2004, 4th IEEE/ACM Int'l Symp. on Cluster Computing & the Grid, Chicago. www-fp.mcs.anl.gov/ccgrid2004

19-22 Apr: CLAG 2004, 1st Int'l Workshop on Collaborative Learning Applications of Grid Technology (with CCGRID), Chicago. http://research.ac.upc.es/clag/clag2004cfp.htm

19-23 Apr: ASYNC 2004, 10th Int'l Symp. on Asynchronous Circuits & Systems, Hersonissos, Crete. www.async04.gr

25-27 Apr: EDP 2004, 10th IEEE/DATC Electronics Design Processes Workshop, Monterey, Calif. www.eda.org/edps/edp04/

26-30 Apr: IPDPS 2004, 18th Int'l Parallel & Distributed Processing Symp., Santa Fe, N.M. www.ipdps.org

MAY 2004

9-12 May: Security and Privacy, Oakland, Calif. http://www.ieee-security.org/TC/SP-Index.html

10-13 May: ISEE 2004, Int'l Symp. on Electronics & the Environment, Scottsdale, Ariz. www.iseesummit.org

13-14 May: NATW 2004, IEEE 13th North Atlantic Test Workshop, Essex Junction, Vt. www.ee.duke.edu/ NATW/

16-19 May: PADS 2004, 18th Workshop on Parallel & Distributed Simulation, Kufstein, Austria. www.pads-workshop.org/pads2004

17-18 May: ICAC 2004, Int'l Conf. on Autonomic Computing (with WWW 2004), New York. www.autonomic-conference.org

19-22 May: ISMVL 2004, 34th Int'l Symp. on Multiple-Valued Logic, Toronto. www.eecg.utoronto.ca/~ismvl2004

23-28 May: ICSE 2004, 26th Int'l Conf. on Software Eng., Edinburgh, UK. http://conferences.iee.org/icse2004/

24-27 May: ECBS 2004, 11th IEEE Int'l Conf. & Workshop on the Eng. of Computer-Based Systems, Brno, Czech Republic. www.fit.vutbr.cz/events/ECBS2004/

JUNE 2004

2-4 June: PBG 2004, Symp. on Point-Based Graphics, Zurich, Switzerland. www.graphics.ethz.ch/PBG/

7 June: CLADE 2004, Workshop on Challenges of Large Applications in Distributed Environments, Honolulu. www.caip.rutgers.edu/clade2004/

7-9 June: POLICY 2004, IEEE 5th Int'l Workshop on Policies for Distributed Systems & Networks, Yorktown Heights, N.Y. www.research.ibm.com/policy2004/index.html

23-25 June: IMSTW 2004, 10th IEEE Int'l Mixed Signals Test Workshop, Portland, Ore. www.ece.pdx.edu/imstw04/

23-25 June: MEMOCODE 2004, 2nd ACM/IEEE Conf. on Formal Methods & Programming Models for Codesign, San Diego, Calif. www.irisa.fr/manifestations/2003/MEMOCODE/

24-25 June: CBMS 2004, 17th IEEE Symp. on Computer-Based Medical Systems, Bethesda, Md. www.cvial.ttu.edu/Conferences/cbms2004/cbms2004.html

24-26 June: IWPC 2004, 12th Int'l Workshop on Program Comprehension, Bari, Italy. http://iwpc2004.di.uniba.it

27-30 June: ICME 2004, Int'l Conf. on Multimedia & Expo, Taipei. www.icme2004.org/

28 June-1 July: DSN 2004, Int'l Conf. on Dependable Systems & Networks, Florence, Italy. www.dsn.org

JULY 2004

6-9 July: ICWS 2004, IEEE Int'l Conf. on Web Services, San Diego, Calif. http://conferences.computer.org/icws/

6-9 July: CEC 2004, IEEE Conf. on E-Commerce, San Diego, Calif. http://tab.computer.org/tfec/cec04

7-9 July: ICPADS 2004, 10th Int'l Conf. on Parallel & Distributed Systems, Newport Beach, Calif. www.cacs.louisiana.edu/icpads2004/

AUGUST 2004

9-10 Aug: MTDT 2004, IEEE Int'l Workshop on Memory Technology, Design, and Testing, San Jose, Calif. Contact Rochit Rajsuman, f.rajsuman@advantest-ard.com.

19-20 Aug: ISESE 2004, Int'l Symp. on Experimental Software Eng., Redondo Beach, Calif. www.isese.org

30 Aug.-1 Sept: ICALT 2004, 4th IEEE Int'l Conf. on Advanced Learning Technologies, Joensuu, Finland. http://lttf.ieee.org/icalt2004

SEPTEMBER 2004

6-10 Sept: RE 2004, 12th IEEE Int'l Requirements Eng. Conf., Kyoto, Japan. www.re04.org

20-24 Sept: WI-IAT 2004, IEEE/ WIC/ ACM Int'l Joint Conf. on Web Intelligence, Beijing. http://wi-consortium.org

20-25 Sept: ASE 2004, 19th IEEE Int'l Conf. on Automated Software Eng., Linz, Austria. www.ase-conference.org

Paving the Way to the Future: New Programs for 2004

Since 1946, the IEEE Computer Society has striven to keep pace with the latest developments in the computer engineering field, adding new professional development resources in response to changing times. Two new publications, expanded digital library access, an online technical bookshelf, and enhanced distance learning courses are just a few of the enhancements to membership being offered this year.

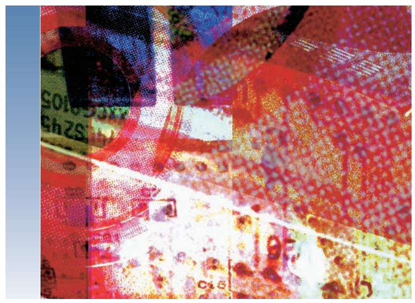
PUBLICATIONS

This month marks the debut of two new IEEE Computer Society-sponsored transactions. In partnership with the Association for Computing Machinery, the Computer Society is launching *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, a periodical dedicated to understanding and advancing the computational management of biological information. A second new transaction, *IEEE Transactions on Dependable and Secure Computing*, established in response to the demand for high-quality research results in secure systems, is also available to members for the first time this year.

IEEE/ACM Transactions on Computational Biology and Bioinformatics

Mapping out patterns found in large-scale molecular biological data—particularly DNA sequences—can lead to powerful insights into biological systems. Notable recent successes in the field have encouraged research biologists to further explore areas in which computational power can be applied to such large volumes of data.

Editor: Bob Ward, Computer, 10662 Los Vaqueros Circle, PO Box 3014, Los Alamitos, CA 90720-1314; bnward@computer.org



Consequently, participation in conferences on computational biology and bioinformatics has grown rapidly in recent years. To serve this growing audience, the Computer Society and its partners have developed *IEEE/ACM Transactions on Computational Biology and Bioinformatics*. Articles in this publication will address methodological, algorithmic, and statistical research issues in biology. The IEEE Neural Networks Council, the IEEE Control Systems Society, and the IEEE Engineering in Medicine and Biology Society join the Computer Society and the ACM in publishing this journal.

Computer Society members can subscribe to *IEEE/ACM Transactions on Computational Biology and Bioinformatics* by visiting www.computer.org/subscribe/. The professional member price is $35 per year.

IEEE Transactions on Dependable and Secure Computing

The Computer Society is producing a new journal intended to help engineers meet challenges in the creation of stable, secure systems. *IEEE Transactions on Dependable and Secure Computing* will provide an international forum for original results in research, design, and development of secure computing methodologies. The journal will serve as a companion pub-

Nominations Open for IEEE Division V Director-Elect

IEEE Computer Society members are invited to submit nominations for candidates to serve as 2005 IEEE Division V director-elect and 2006-2007 Division V director.

Division directors represent the members of IEEE societies on the IEEE Board of Directors and the Technical Activities Board; Division V and VIII directors represent the Computer Society membership. Elections for Division V director are typically held in even-numbered years, and Division VIII elections are held in odd-numbered years. The elected representative then serves one year in the director-elect role before assuming a two-year division director term.

IBM's Gene Hoffnagle currently serves as IEEE Division V director for 2004-2005. Jim Issak of Southern New Hampshire University serves as IEEE Division VIII director for 2003-2004. Computer Society past president Steve Diamond was elected in November 2003 to serve as IEEE Division VIII director-elect for 2004.

Submit nominations by **23 January** to Stephen L. Diamond, Chair, Nominations Committee, IEEE Computer Society, 1730 Massachusetts Ave. NW, Washington, DC 20036-1992.

IEEE Computer Society Publications Seek Editors in Chief for 2005-2006 Terms

The IEEE Computer Society seeks applicants for the position of editor in chief for the following periodicals for terms starting 1 January 2005: *IEEE Intelligent Systems, IEEE Transactions on Pattern Analysis and Machine Intelligence,* and *IEEE Transactions on Knowledge and Data Engineering*.

Qualifications and Requirements

Candidates for any Computer Society editor-in-chief position should possess a good understanding of industry, academic, and government aspects of the specific publication's field. In addition, candidates must demonstrate the managerial skills necessary to process manuscripts through the editorial cycle in a timely fashion. An editor in chief must be able to attract respected experts to his or her editorial board. Major responsibilities include

- actively soliciting high-quality manuscripts from potential authors and, with support from publication staff, helping these authors to get their manuscripts published;
- identifying and appointing editorial board members, with the concurrence of the Publications Board;
- selecting competent manuscript reviewers, with the help of editorial board members, and managing timely reviews of manuscripts;
- directing editorial board members to seek special-issue proposals and manuscripts in specific areas;
- providing a clear, broad focus through promotion of personal vision and guidance where appropriate; and
- resolving conflicts or problems as necessary.

Applicants should possess recognized expertise in the computer science and engineering community, have editorial experience, and be able to lead an active editorial board and work effectively with technical and publishing professionals. Applicants must have clear employer support.

Search Procedure

Prospective candidates are asked to provide, by **15 March**, a complete curriculum vitae, a brief plan for the publication's future, and a letter of support from their institution or employer. Materials should be sent in Portable Document Format (PDF) to the search committee chair for the appropriate publication. Contact information for each search committee coordinator follows.

IEEE Transactions on Knowledge and Data Engineering. Contact Alicia Stickley, Transactions Production Manager, IEEE Computer Society, 10662 Los Vaqueros Circle, P. O. Box 3014, Los Alamitos, CA 90720-1264, +1 714 821 8380 (voice), +1 714 821 4010 (fax); astickley@computer.org.

IEEE Transactions on Pattern Analysis and Machine Intelligence. Contact Alicia Stickley, Transactions Production Manager, IEEE Computer Society, 10662 Los Vaqueros Circle, P. O. Box 3014, Los Alamitos, CA 90720-1264, +1 714 821 8380 (voice), +1 714 821 4010 (fax); astickley@computer.org.

IEEE Intelligent Systems. Contact Dennis Taylor, Lead Editor, IEEE Computer Society, 10662 Los Vaqueros Circle, P. O. Box 3014, Los Alamitos, CA 90720-1264, +1 714 821 8380 (voice), +1 714 821 4010 (fax); dtaylor@computer.org.

lication to *IEEE Security and Privacy* magazine, launched last year.

Traditionally, outlets that explore topics in reliable computing have focused on established areas such as e-commerce, benchmarking, and system architectures. In addition to these subjects, papers published in *IEEE Transactions on Dependable and Secure Computing* will address the development of secure systems for emerging technologies. For instance, the minute size of components used in nanoscale computing systems presents a special concern regarding materials reliability. Burgeoning wireless services also present new issues in security and dependability.

Computer Society members can subscribe to *IEEE Transactions on Dependable and Secure Computing* by visiting www.computer.org/subscribe/. The professional member price is $31 per year.

DISTANCE LEARNING CAMPUS

The Distance Learning Campus offers valuable learning opportunities exclusively to members of the IEEE Computer Society. Members may take any of 100 Web-based training courses, which are selected based on survey data and member profile records. These courses cover a variety of computing programs and subjects that include XML, AMA management skills, Java, Cisco, Windows security, Oracle, Unix, and Visual C++. Twice a year, the Distance Learning Campus course catalog is reevaluated and updated with new selections.

Offered in partnership with KnowledgeNet, courses in the Distance Learning catalog offer presentations with voiceovers, 3D graphics, flash animations, on-screen text, and *visual sentences* that turn complex concepts into easy-to-understand images. Users can take pre-assessment tests, track ongoing progress, and study at a self-directed pace.

More than 15,000 members used the Distance Learning Campus in 2003. Visit the campus at www.computer.org/distancelearning/.

IEEE COMPUTER SOCIETY ONLINE BOOKSHELF

The IEEE Computer Society has announced a new member benefit for 2004. The Computer Society Online Bookshelf, a subset of 100 unabridged books from Books24x7's ITPro collection, is now available for free to all Society members. Books24x7 offers content from more than 90 publishers including Oxford Press, Wiley, Microsoft, and MIT Press.

Book topics run the gamut of computer technology issues: networks and protocols, programming languages, certification and compliance, business and culture, desktop applications, operating systems, ReferencePoint suites, software engineering, and Web development. Titles on the Computer Society Online Bookshelf are available for viewing through a customizable Referenceware online interface, which allows users to organize favorite titles and bookmarks.

All members will have free access to a rotating set of 100 books that a committee of Computer Society volunteers has selected as being of most value to their peers. For $89, members can access a 500-book collection. For $299, members can access Books24x7's full 2,500-book collection.

Books24x7, a subsidiary of SkillSoft, provides Web-based digital technical and business reference content to more than a million subscribers. Books24x7's Referenceware platform enables users to search, browse, read, and interact with the content of several targeted professional reference libraries.

Complete title lists, as well as registration information, for the Computer Society's Books 24x7 collections are available at www.computer.org/bookshelf/.

CONFERENCES

Each year, the IEEE Computer Society sponsors a spectrum of technical meetings in all subfields of the computing profession, ranging from small workshops to large symposia. Society technical councils, task forces, and technical committees sponsor or partcipate in more than 130 conferences each year. Conferences of special note in the next six months are previewed below.

Proceedings from many conferences are available through the Computer Society Digital Library. Subscribers enjoy full access to an online library of articles, proceedings, and magazine features, available at www.computer.org/publications/dlib/. Others can search the collection and purchase individual items.

26-30 January: SAINT 2004

Gathering this year in Tokyo, researchers studying the Internet and its applications will have an opportunity to learn about the latest developments in IPv6 technology, high-performance grid computing, and Internet security, among other topics. Panels at SAINT are set to address the future of Internet architecture, ubiquitous services, and networking.

SAINT, cosponsored by the Information Processing Society of Japan, is the flagship conference of the IEEE Computer Society Technical Committee on the Internet. Fees for advance registration, before 6 January, are $533 for members, $670 for nonmembers, and $370 for students. Late and onsite registration is $650 for members, $820 for nonmembers, and $460 for students. Visit www.saint2004.org for more conference information.

7-11 March: IEEE Infocom 2004

Since 1981, the IEEE Infocom conference series has provided a premier venue for the presentation and exchange of new ideas and research findings in the computer networking field. For 2004, conference organizers have solicited papers on a broad variety of topics that include multimedia protocols, optical networks, and scheduling and buffer management. Two related events, the 2004 Workshop on High-Speed Networking and OpenArch 2004, will run concurrently with the Infocom conference, set this year in Hong Kong. Infocom 2004's early registration period ends 6 February.

For more information regarding Infocom, including registration fees as they become available, visit www.ieee-infocom.org/2004/.

23-26 March: ICDCS 2004

First presented by the IEEE Computer Society Technical Committee on Distributed Processing (TCDP) in 1979, ICDCS is the oldest conference series in the field of distributed computing systems. ICDCS provides a forum for engineers and scientists in academia, industry, and government to discuss the latest research findings on topics including agents and mobile code, middleware, and ubiquitous computing. ICDCS 2004 will take place in Tokyo.

Advance registration fees are $500 for members, $690 for nonmembers, and $310 for students. After 31 January, fees are $620 for members, $820 for nonmembers, and $370 for students. Visit www.cis.ohio-state.edu/icdcs04/ for program highlights and more information.

27-31 March: IEEE VR 2004

Leading experts in virtual reality and closely related fields such as augmented reality, mixed reality, and 3D interaction will share insights and discoveries at the 2004 IEEE Virtual Reality Conference. The VR conference series is the cornerstone of the Computer Society Technical Committee on Visualization and Graphics' annual event calendar. Tutorials and workshops share the bill with an extensive exhibition of the latest tools and products available to the VR community. Located this year in Chicago, the conference and exhibition will take place in conjunction with the 2004 Haptics Symposium.

IEEE Computer Society members enjoy discounted rates on VR 2004 conference registration fees. Visit www.vr2004.org for more details.

30 March - 02 April: ICDE 2004

The 20th International Conference on Data Engineering offers an opportunity for the presentation of research results on advanced data-intensive applications and data and knowledge engineering issues. Conference participants will share research solutions and identify new directions for future research and development. Set for Boston in 2004, ICDE is sponsored by the IEEE Computer Society Technical Committee on Data Engineering.

Seminars on the ICDE 2004 schedule include XML query processing, data mining for intrusion detection, and metadata management, among others. Visit the ICDE 2004 Web site at www.cse.uconn.edu/icde04/ for information on conference registration.

14-16 April: Cool Chips VII

Since 1998, the annual Cool Chips conference has presented the research behind leading-edge technologies related to all areas of microprocessors and their applications. This year, in Yokohama, Japan, the Cool Chips VII symposium will focus on low-power, high-performance processors for multimedia, as well as novel architectures and schemes for single-core, multicore, and embedded systems. Cool Chips VII organizers are also soliciting original works on software solutions, including binary translations, compiler issues, and low-power techniques.

Visit www.coolchips.org for more conference information as it becomes available.

19-22 April: CCGrid 2004

In its fourth year, the 2004 International Symposium on Cluster Computing and the Grid, sponsored by the IEEE Computer Society and the IEEE Task Force on Cluster Computing in conjunction with the ACM, addresses the role of grid computing in the new century. Several workshops, addressing topics ranging from grid education, to advanced networks, to P2P computing, are scheduled to run concurrently with CCGrid 2004, set this year in Chicago.

Visit www.ccgrid.org/ccgrid2004/ for registration and program information as it becomes available.

26-30 April: IPDPS 2004

In its 18th year, the International Parallel and Distributed Processing Symposium continues to provide an international forum for researchers in all facets of parallel computing. Scheduled workshop topics at IPDPS 2004 include advances in parallel and distributed computing models, communication architecture for clusters, and grid benchmarking.

Beginning this year, IPDPS participants will have the opportunity to organize informal birds-of-a-feather sessions. Convening this year in Santa Fe, New Mexico, IPDPS 2004 is presented by the Computer Society's Technical Committee on Parallel Processing. See www.ipdps.org/ipdps2004/ for more information regarding IPDPS 2004.

9-12 May: Security and Privacy

For more than 20 years, the IEEE Symposium on Security and Privacy

Mentor Program Seeks Women Undergraduates by 16 February

The CRA Committee on the Status of Women in Computing Research (CRA-W) is seeking undergraduates and potential mentors for its annual Distributed Mentor Project. Operating under a $1.6 million grant from the National Science Foundation, the Distributed Mentor Project has aimed, since 1994, to increase the number of women in computer science and engineering graduate school programs. Outstanding women undergraduates are paired with female mentors who shepherd the students through a summer of research hosted at the mentor's home institution.

The intention of the mentoring and research program is to more actively engage the undergraduates in the work environment they would encounter in a graduate school or private research setting. The geographically distributed nature of the project helps to address the deficit of female computer engineering faculty at many institutions, where students often lack women as role models. The student participants commonly form close ties with their mentors, which translates into a much higher likelihood of moving on to postgraduate studies.

Student participants receive a $600 weekly stipend for up to 10 weeks of study at the host institution, plus a travel allowance of up to $500. Mentors can receive $2,500 for mentoring one student, or $4,000 for mentoring two or more.

Participants create dedicated Web sites as part of their summer research activities. Web sites from past summers are available at www.cra.org/Activities/craw/dmp/awards/2003/2003.php.

To apply as an undergraduate or potential mentor, visit the CRA-W Distributed Mentor Project Web site at www.cra.org/Activities/craw/dmp/index.php. The deadline for applications to the summer 2004 program is **16 February**.

The CRA was established to foster leadership development among groups that are underrepresented in the computing profession. The IEEE Computer Society is one of the CRA's six Affiliated Professional Society Members.

has brought together researchers and practitioners to share developments in computer security and electronic privacy. Workshop and tutorial topics for the 2004 conference are scheduled to include intrusion detection, language-based security, biometrics, and peer-to-peer security. Again this year, the symposium will feature a session of five-minute talks, where attendees can present preliminary research results or summaries of works published elsewhere.

This year's symposium takes place in Berkeley, California. See www.ieee-security.org/TC/SP-Index.html for more conference information.

23-28 May:
ICSE 2004

For 26 years, the International Conference on Software Engineering has provided researchers, practitioners, and educators an opportunity to present and discuss trends in the software engineering field. ICSE 2004, in Edinburgh, Scotland, will include a doctoral symposium for PhD students in addition to keynote speeches and other presentations. Topics to be addressed at ICSE 2004 include software engineering principles, theories, techniques, and tools, as well as other related areas, such as programming languages, distributed systems, databases, and networks.

Several software engineering-related conferences and symposia are scheduled to run concurrently with ICSE, including EWSA 2004, the first European Workshop on Software Architecture.

Visit the conference Web site at http://conferences.iee.org/icse2004/ for more details on ICSE 2004.

24-27 May:
ECBS 2004

Participants in the 11th Annual IEEE International Conference and Workshop on the Engineering of Computer-Based Systems will meet in Brno, Czech Republic, to advance research developments in the engineering of computer-based systems. Topics to be addressed include system modeling, architectures, communications, reliability, system integration, and project management. An industrial track will provide a forum for companies to present short papers on results of industrial research and development.

ECBS 2004 is sponsored by the IEEE Computer Society Technical Committee on the Engineering of Computer-Based Systems. Visit www.fit.vutbr.cz/events/ECBS2004/ for the latest information on conference registration.

IEEE Computer Society members enjoy a minimum 25-percent discount on registration fees for any conference we sponsor. Visit www.computer.org/conferences/ for a full list of conferences. ■

KUWAIT UNIVERSITY. The Department of Information Science at the College for Women in Kuwait University invites applications for faculty positions at the ranks of Assistant, Associate, and Professor starting in February 2004 in the following areas: Information Science, Information Systems and Technology, Computer Science or Computer Engineering, Knowledge Management and Data Mining, etc. The medium of instruction is English. Responsibilities include teaching undergraduate courses, conducting independent scholarly research, and administrative duties. Both male and female candidates may apply. The ideal candidate must have a Ph.D. from a reputed western university in the area of specialization or related fields. The candidate must also demonstrate evidence of quality teaching and research and have full command of English. To be considered for an Associate or Professor level, the candidate must have a strong publication record in refereed international journals. The College for Women is part of Kuwait University which is one of the leading public institutions of higher learning in the Gulf region. The university emphasizes quality in teaching, research, and service. Kuwait University offers a generous benefit package that includes competitive tax-free salary that is determined according to rank and experience, annual air tickets for faculty member and his/her family (spouse and up to three children under the age of 20), tuition allowance for children schooling, a one-time settling-in allowance, housing allowance, free national health care, paid mid-year holidays and summer vacation, and end-of-contract gratuity. The University also offers an excellent academic environment and financial research support. Interested candidates should submit a current CV with a cover letter indicating their areas of interest. It is preferred that candidates send their CV by e-mail at jobs@cfw.kuniv.edu. Applications are accepted until the positions are filled. All communication (application and inquiries) should be addressed to: Dean, College for Women, Kuwait University, P.O. Box 5969, Safat 13060, Kuwait. Phone: +(965) 251-4232, Fax: +(965) 255-1157, e-mail: jobs@cfw.kuniv.edu.

UNIVERSITY OF CENTRAL FLORIDA, Orlando, Florida, The School of Computer Science in the College of Engineering & Computer Science. UCF Computer Science is looking for talented faculty in all areas of specialization. We expect to fill over 10 faculty lines in the next few years, and seek outstanding candidates for tenured and tenure-track faculty positions at all levels for appointments in 2004. We are particularly interested in candidates with research interests in bioinformatics, compilers, computer architecture, computer graphics, computer vision, database and information systems, mobile systems, multimedia computing, networking, novel forms of computing, operating systems, parallel and distributed computing, and security/cryptography. We offer competitive salary and start-up packages to outstanding individuals, and our faculty enjoy generous benefits. New faculty have graduate student support and significantly reduced teaching loads. Special senior-level positions are available to exceptionally qualified individuals, and we welcome candidacies from coordinated groups of faculty. In the past five years, nine distinguished faculty have joined UCF Computer Science, and we seek to add to the growing strength of our program. Applicants should have a Ph.D. and a strong commitment to the academic process, including teaching, scholarly publications, and sponsored research. Candidates at the Associate or Full Professor level should have demonstrated leadership in their fields at the national/international level. Those applying for Assistant Professorships should have a track record of high quality publications, and be recognized for their potential. UCF Computer Science is the oldest Ph.D. granting CS program in the state of Florida. We have a rapidly growing educational and research program with over $2.8MM in annual expenditures from grants and contracts, with over 250 graduate students and 860 undergraduate CS majors. In addition, we offer a B.S. degree in Information Technology (IT), with over 400 undergraduate majors. We also encourage faculty candidates for the IT program to respond to this ad. The University is strongly committed to continue the buildup of strength in CS, with significant special monetary support dedicated to maintain and expand the excellence of the School of Computer Science. Our Federal research sponsors include NSF, ARO, ONR, NASA, U.S. Army STRICOM, the Department of Transportation, and other agencies of the Department of Defense. We enjoy the support of numerous industrial sponsors including established companies such as Adaptec, ATI, Boeing, Harris, Honeywell, Imagesoft, Intel, Lockheed Martin, Lucent, Oracle, Schwartz Electro-Optics, as well as hi-tech start-ups such as Aximetric, Giganet Technologies, and Millenium Technologies. UCF has over 42,000 students and is among the nation's fastest growing universities. We are located in Orlando, FL at the center of the I-4 High Tech Corridor with a thriving industrial base in telecommunications, computer systems, semiconductors, defense and space, lasers, simulation and software, and the world renowned entertainment industry. The Orlando metropolitan area enjoys an exceptional climate with rapid access to the Atlantic seashore and the Gulf of Mexico. Because of the presence of major theme parks such as Disney World, Sea World and Universal Studios, we benefit from a major airport with numerous direct international and national connections. Orlando is also a major center for national and international technical conferences. Please send applications electronically (PDF format is preferred) to cs-positions@cs.ucf.edu. Applications should include a cover letter, a detailed CV, a Research Statement, a Teaching Statement, and five names of references. Alternatively, though not preferred, hardcopy applications may be sent to: Chair, Computer Science Search Committee, School of Computer Science, University of Central Florida, Orlando, FL 32816-2362. Applications received by January 31, 2004 will be given priority. The University of Central Florida is an Equal Opportunity/Affirmative Action employer. Women and minorities are particularly encouraged to apply. As an agency of the State of Florida, UCF makes all application materials and selection procedures available for public review.

THE UNIVERSITY OF TEXAS AT ARLINGTON, Computer Science and Engineering Department, Faculty Openings for Fall 2004. The University of Texas at Arlington (UTA), Computer Science and Engineering (CSE) Department - CSE@UTA invites applications for multiple tenure-track faculty positions at all levels. However, preference will be given to positions at assistant or associate professor levels. All areas of computer science will be considered, including: computer security; bio-informatics, software engineering; pervasive computing; multimedia and video processing; intelligent systems; networks and telecommunications; database and data mining; and applied theory. UTA, part of The University of Texas System, is located in the heart of the rapidly growing Dallas/Fort Worth area, one of the nation's largest high-technology regions, with a flourishing industrial base and excellent opportunities for industry/university collaboration. We at CSE@UTA are committed to excellence in research, teaching, and service. We are in the

SUBMISSION DETAILS: Rates are $275.00 per column inch ($300 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.

Fourth year of our "Top 25 Initiative" plan to reach a national top 25 ranking within 10 years. The initiative is strongly supported by all CSE@UTA stakeholders including the UTA administration, faculty, students and alumni, and industry partners. Since 2000, we have added 8 new tenure track faculty and 5 new full-time non-tenure track faculty to our roster of 38 full-time faculty. The number of our PhD students has more than doubled since 2000 and the actively funded research awards exceeded $5.8 million in 2002-03 academic year. Applicants must have an earned doctorate in computer science, computer engineering, or closely related fields and a commitment to teaching and scholarly research. Applicants are expected to have an excellent record of professional accomplishments, commensurate with their level of experience. The faculty opening is anticipated for September 2004. Screening of applications will begin immediately and will continue until all positions are filled. Interested persons should submit a letter of application, a resume, and reference letters online at: http://www.cse. uta.edu/application/. Please note that we do not accept hard-copy submissions. For additional information, please contact: Dr. David Kung, Chair of Search Committee, Department of Computer Science and Engineering, The University of Texas at Arlington, Phone: 817-272-3605, FAX: 817-272-3070, Email: search@cse.uta.edu; http://www.cse.uta. edu. The University of Texas at Arlington is an Equal Opportunity/Affirmative Action Employer.

NORTHWESTERN UNIVERSITY. Applications are invited for tenure-track faculty positions starting September 1, 2004 in the Department of Electrical and Computer Engineering of the Robert R. McCormick School of Engineering and Applied Science at Northwestern University. We have needs in the following areas: Solid State and Photonics, with an emphasis on nanotechnology; Systems, with an emphasis on networks and communications; Computer Engineering, with an emphasis on computer architecture. The candidates must have a Ph.D. or equivalent in Electrical and Computer Engineering or Computer Science, and will be expected to teach three to four quarter courses per year, perform research, and supervise graduate students. We are primarily looking for candidates at the Assistant Professor level (except in the Solid State and Photonics area, where we are searching for a senior faculty member); however, outstanding candidates at the Associate and Full Professor levels will also be considered. The ECE department has 31 full-time faculty members, 250 undergraduate and 120 graduate students, and many state-of-the-art laboratory facilities. The department offers B.S., M.S., and Ph.D. degrees in both Electrical Engineering and Computer Engineering. For more information about the department, please check: http://www.ece.northwestern.edu. Northwestern University is an Affirmative Action/Equal Opportunity Employer. Applications from women and minorities are especially encouraged. Hiring is contingent upon eligibility to work in the United States. The deadline for applications is February 1, 2004; however, the search will continue until the positions are filled. Please send a resume and names of at least five references to: Professor Prith Banerjee, Chairman, Department of Electrical and Computer Engineering, Northwestern University, 2145 Sheridan Road, Evanston, IL 60208-3118. EMAIL: banerjee@ece.northwestern.edu. Phone: (847) 491-3641.

INDIANA UNIVERSITY, Computer Science Department, Faculty Position. The Indiana University Computer Science Department anticipates filling a tenure track, assistant professor position beginning in the fall of 2004. Though applications from all areas of computer science are welcome, we are strongly interested in candidates with research interests in embedded systems, algo-

Rensselaer

Why not Change the World?

TENURE-TRACK FACULTY POSITION
Computer Science

The Division of Engineering and Science (DOES), Rensselaer at Hartford, invites applications for a tenure-track or tenured position for the 2005 academic year beginning in Fall 2004. Rank will be commensurate with qualifications. The candidate must have a Ph.D. in computer science or closely-related area and have demonstrated sufficient promise or achievement in both research and teaching. Candidates from the general area of User Interface Design are especially encouraged but we welcome strong candidates in any area. The candidate will teach graduate-level courses, supervise Ph.D. students, and be expected to develop an externally funded research program.

Rensselaer at Hartford, (www.rh.edu), a branch campus of Rensselaer Polytechnic Institute, focuses on education for the working professional. Over 1000 part-time students are seeking MBA or Master of Science degrees in management, computer science, information technology, and engineering. Rensselaer at Hartford's campus is located in the center of Connecticut, well positioned within a corridor of many of America's largest corporations and entrepreneurial organizations. The region offers a rich and diverse cross section of industries and financial institutions. Hartford is conveniently located within a short distance to many cultural, recreational, and educational opportunities. This is an exciting opportunity for a strong, dynamic individual to be a leader in the growth of a campus from an almost entirely education orientation to a mixture of education and research, without ever losing focus on the high-end working professional.

We offer competitive salaries and an excellent benefits package that includes health, dental, and life insurance, a retirement plan, tuition assistance, and much more. Find out more details at: www.hr.rpi.edu.

To apply, send a CV and list of references with a cover letter summarizing your qualifications to: **Alan Eckbreth, VP and Dean, Rensselaer at Hartford, 275 Windsor Street, Hartford, Connecticut 06120; or e-mail: hr@rh.edu**

Applications material received by February 29, 2004 will receive full consideration.

Rensselaer Polytechnic Institute, an equal opportunity-affirmative action employer, particularly encourages applications from women and minority candidates.

INFORMATICS iCORE CIRCLE OF RESEARCH EXCELLENCE

iCORE Research Chair
Distributed High Performance Computing

The Alberta Informatics Circle of Research Excellence (iCORE), in conjunction with Alberta universities, is seeking a Chair to lead a research program in the distributed systems, high performance computing, or grid computing areas. The Chair is expected to be an internationally recognized, exceptional researcher who will develop a research team in one or more of these areas.

Western Canada, through the WestGrid project (http://www.westgrid.ca), has invested $44 million in high performance computing research infrastructure that spans Calgary, Edmonton and Vancouver. These grid-enabled HPC resources are connected by a leading-edge dedicated optical network.

iCORE Chairs can be held at any of the three research universities in Alberta: University of Alberta, University of Calgary and the University of Lethbridge.

UNIVERSITY OF CALGARY University of Lethbridge UNIVERSITY OF ALBERTA

Successful applicants will be appointed with tenure and have substantial research funding for an initial period of five years, renewable once for a second five years.

Salaries and research funding associated with iCORE Chairs are highly competitive.

If you are interested, or know of someone who may be interested, please contact:

Lynn Sutherland, Vice President, Programs
(403) 210-5335 sutherland@icore .ca

www.icore.ca

rithms and programming languages. The CS department, which is part of the College of Arts and Sciences, is working closely with our new School of Informatics which is also seeking to fill CS related positions. A Ph.D. in Computer Science is required for all CS faculty positions. Applicants must have demonstrated potential for excellence and productivity in research. In addition, a strong contribution to the educational mission of the department is expected. The department occupies a spacious limestone building with extensive state-of-the-art computing facilities. The attractive wooded campus of Indiana University is located in Bloomington, chosen as one of the most cultural and livable small cities in the US, and only one hour from the Indianapolis airport. To learn more about the department please visit our web site at www.cs.indiana.edu. Please send a detailed CV and a list of references to: Faculty Search, Computer Science Department, Indiana University, Lindley Hall 215, Bloomington, IN 47405-7104. email: search@cs.indiana.edu. Indiana University is an Equal Opportunity/Affirmative Action Employer. The Computer Science Department strongly encourages applications from women and minorities.

THE UNIVERSITY OF CALIFORNIA, BERKELEY. THE UNIVERSITY OF CALIFORNIA, BERKELEY invites applications for tenure-track positions in ELECTRICAL ENGINEERING AND COMPUTER SCIENCES at the Assistant Professor, Associate Professor, or Full Professor level, beginning in Fall Semester 2004, subject to budgetary approval. Several faculty searches have been approved. We are also considering the possibility of joint searches with other UC Berkeley departments. Applicants should have received (or be about to receive) a doctoral degree in Computer Science, Electrical Engineering, Computer Engineering, or a related field. A principal requirement is demonstrated excellence in research. In addition, potential for excellence in teaching and leadership are important requirements. Successful applicants will be expected to establish a quality research program and to teach both graduate and undergraduate courses in their general area of specialty. Interested persons should send an application consisting of a resume, a one- to two-page statement of their future research and teaching interests/plans, a select subset of publications, and the names of three references whom you have asked to send in recommendations. Review of completed applications will begin December 15, 2003. We will not consider applications received after March 1, 2004. Recommendation writers should send letters directly to the same address where applications are sent, to arrive before January 1, 2004 if possible. Reference letters will NOT be requested directly by the department. Computer Science applications should be sent to: CS Faculty Search Committee, c/o Debra Zaller, Computer Science Academic Personnel, 381 Soda Hall # 1776, UC Berkeley, CA 94720-1776. Electrical Engineering applications should be sent to: EE Faculty Search Committee, c/o Jean Richter, Electrical Engineering Academic Personnel, 231 Cory Hall # 1770, UC Berkeley, CA 94720-1770, The University of California is an Equal Opportunity, Affirmative Action Employer.

UNIVERSITY OF CALIFORNIA, LOS ANGELES, DEPARTMENT OF COMPUTER SCIENCE. The Department of Computer Science in the Henry Samueli School of Engineering and Applied Science at the University of California, Los Angeles, invites applications for tenure-track positions in all areas of Computer Science and Computer Engineering. Applications are also strongly encouraged from distinguished candidates at senior levels. The UCLA Campus administration is committed to significant growth in the faculty of the Computer Science Department, with a projection of 2-3 new faculty members per year for the next two years. Quality is our key criterion for applicant selection. Applicants should have a strong commitment to both research and teaching and an outstanding record of research for their level. We seek applicants in any mainstream area of Computer Science and Computer Engineering. To apply, please visit http://www.cs.ucla.edu/recruit. Faculty applications received by January 15 will be given full consideration. The University of California is an Equal Opportunity/Affirmative Action Employer.

UNIVERSITY OF WASHINGTON, Tenure-Track, Research, and Teaching Faculty. The University of Washington's Department of Computer Science & Engineering has one or more open positions in a wide variety of technical areas in both Computer Science and Computer Engineering, and at all professional levels. A moderate teaching load allows time for quality research and close involvement with students. Our recent move into a beautiful new building, the Paul G. Allen Center for Computer Science & Engineering, allows many opportunities for new projects and initiatives. Information about the department can be found on the web at http://www.cs.washington.edu. We welcome applicants in all CSE research areas (but especially candidates whose research interests include hardware). We expect candidates to have a strong commitment both to research and to teaching. The department is primarily seeking individuals at the Assistant Professor rank; however, under unusual circumstances and commensurate with the qualifications of the individuals, appointments may be made at the rank of Associate Professor or Professor. We are also seeking non-tenured Research Assistant Professors and Lecturers. Applicants for the tenure-track and research positions must have earned a doctorate by the date of appointment; those for the lecturer position must have earned at least a Master's degree. Please apply online at http://www.cs.washington.edu/news/jobs.html, or if web access is impossible, send a letter of application, a resume, statement of research and teaching interests, and the names of four references to: Faculty Recruiting Committee, Computer Science & Engineering, University of Washington, Box 352350, Seattle, WA 98195-2350. Applications received by January 30, 2004 will be given priority consideration. The University of Washington is a recipient of a new National Science Foundation ADVANCE Institutional Transformation Award to increase the participation of women in academic science and engineering careers. We are building a culturally diverse faculty and encourage applications from women and minority candidates. The University of Washington is an affirmative action, equal opportunity employer.

DUKE UNIVERSITY, Department of Computer Science. Experimental Systems Faculty Position: We invite applications and nominations for a tenure-track or tenured faculty position at any rank in the Department of Computer Science at Duke University, to start September 2004. Preference will be given to applicants in the various areas of experimental systems and architecture. We are seeking to build upon our strong, highly collaborative group in experimental systems. We are broadly interested in all areas of experimental systems with emphasis on databases, computer architecture, networking, security, and distributed systems. Computational Biology Faculty Position: We invite applications and nominations for a tenure-track assistant professor position in the Department of Computer Science at Duke University, effective from September 2004. We seek applicants with outstanding records in the general area of computational biology and bioinformatics. The areas of particular interest include but are not limited to sequence analysis, functional genomics, transcriptomics, structural genomics, machine learning and data mining, and computational simulation and mathematical modeling. For more information about the faculty, facilities and other resources, please refer to www.cs.duke.edu. Applications should be submitted online at http://www.cs.duke.edu/csnews/facsearch. Applications should include a curriculum vitae, a list of publications, and copies of the most important publications. A Ph.D. in computer science or related area is required.

BIOINFORMATICS FACULTY
COMPUTER SCIENCES DEPARTMENT
The City College of the City University of New York

The City College invites applications for a tenure track faculty position in Bioinformatics at the Assistant Professor level, starting with the Fall semester of 2004. We seek applicants with an active research profile in biologically, chemically or medically oriented computer science, design an active collaboration with workers in biomedical engineering, and who can develop ties with the local pharmaceutical and medical industries. Applicants must have a PhD in Computer Science or an allied filed, a demonstrated record of publications and a commitment to research excellence.

City College is the oldest college of the City University system; Computer Science resides within the School of Engineering, which also houses the Department of Biomedical Engineering. The Computer Science Department offers the BS and MS in Computer Sciences; the PhD is offered as well, in collaboration with the Graduate School of the City University of New York. The BE in Computer Engineering is offered jointly with the Department of Electrical Engineering.

The Department's active research reflects its position in the School of Engineering, and encompasses diverse areas of Computer Science and Engineering. Our primary research concentrations are in computational geometry and vision, speech recognition, scientific computation, data systems and information retrieval, distributed computing information security and assurance, combinatorial mathematics, web-based human-computer interaction information management and E-commerce, multimedia networks and digital libraries, computational methods for image and speech processing, remote sensing, and computational algebra. Further information can be obtained from the Department's web site as http://www-cs.engr.ccny.cuny.edu.

Information on the City College Department of Biomedical Engineering can be obtained at http://www.ccny.cuny.edu/nycbe; information on the City University PhD program in Computer Science can be obtained at http://web.gc.cuny.edu/Computerscience.

the City College of New York

Applications with vitae and names of three (3) references, must be postmarked by February 15, 2004, to **Professor Douglas Troeger, Chair, Computer Sciences Department, R8/206, The City College of New York, Convent Avenue at 138th St., NY, NY 10031.**

The City College/CUNY is an EEO/AA/IRCA/ADA Employer.

IOWA STATE UNIVERSITY
OF SCIENCE AND TECHNOLOGY

Department of Electrical and Computer Engineering Iowa State University of Science and Technology invites applications for The Palmer Chair in Electrical and Computer Engineering: Candidates in any traditional or emerging area of Electrical and Computer Engineering will be considered. Applicants should have credentials appropriate for a distinguished position at the rank of full professor. Applicants must have a) an earned doctorate in electrical or computer engineering or related area, b) effective communication skills and experience working closely with undergraduate and graduate students, c) strong record in research and scholarship, d) national and international acclaim in their areas of expertise, e) demonstrated ability to interact productively with industrial community and/or governmental agencies, and f) demonstrated ability to provide leadership and mentoring to build a strong research and teaching program in their areas of expertise. The responsibilities include: 1) teaching at the undergraduate and graduate levels, 2) establishing and directing leading edge research program and publishing **highly cited** research results in leading journals and conferences, 3) guiding graduate students, 4) seeking, establishing, and maintaining external research funding, 5) providing leadership in professional and outreach activities. Full information can be found at the department website www.ece.iastate.edu. Application deadline is March 31, 2004 or until position is filled. Applicants should send a letter of application and a resume including at least three references to Arun K. Somani, Chair, Electrical and Computer Engineering Department, 2215 Coover Hall, Iowa State University, Ames, Iowa 50011-3060. ISU is an EO/AA employer.

D. E. Shaw Research and Development

Research on Algorithms and Architectures for Computational Biochemistry

Extraordinarily gifted computer scientists, systems architects, electrical engineers and systems software professionals are sought to join a rapidly growing New York–based research group pursuing an ambitious, long-term project aimed at achieving major scientific advances in the field of biochemistry and fundamentally transforming the process of drug discovery. This research effort is being financed by the D. E. Shaw group, an investment and technology development firm with approximately $6 billion in aggregate capital, and operates under the direct scientific leadership of its founder, Dr. David E. Shaw.

Among the group's current research activities is a project aimed at developing a massively parallel special-purpose supercomputer and innovative mathematical and computational techniques to direct unprecedented computational power toward the solution of key scientific and technical problems in the fields of molecular simulation and molecular design. Successful candidates will be working closely with a number of the world's leading computational chemists and biologists, and will have the opportunity not only to participate in an exciting entrepreneurial venture with considerable economic potential, but to make fundamental contributions within the fields of biology, chemistry and medicine.

Serious candidates will have an exceptionally distinguished history of academic and/or industrial accomplishment in computer science, electrical engineering, applied mathematics, or a related area. Particularly relevant areas of expertise might include parallel computation, high-speed interconnection networks, scientific computing, numerical analysis, optimization, the analysis of algorithms, operating systems, digital systems simulation, reconfigurable computing, and ASIC design, but specific knowledge of any of these areas is less critical than exceptional intellectual ability and a demonstrated track record of achievement. We are prepared to reward exceptionally well-qualified individuals with above-market compensation.

Please send your curriculum vitae (including list of publications, thesis topic, and advisor, if applicable) to Research.Development7@deshaw.com.

D. E. Shaw Research and Development, L.L.C. does not discriminate in employment matters on the basis of race, color, religion, gender, national origin, age, military service eligibility, veteran status, sexual orientation, marital status, disability, or any other protected class.

D E Shaw & Co

Assistant Professor applicants should arrange for at least four letters of reference to be sent preferably via email (facsearch@cs.duke.edu) to the Faculty Search Chair. Senior candidates should provide the names and contact information of three potential references. To guarantee full consideration, applications and letters of reference should be received no later than January 15, 2004. facsearch@cs.duke.edu. Faculty Search Chair, Department of Computer Science, Duke University, Durham, NC 27708-0129. Duke University is an affirmative action, equal opportunity employer.

UNIVERSITY OF CENTRAL ARKANSAS. The Computer Science Department at the University of Central Arkansas (UCA) invites applications for a tenure-track faculty position, with rank based on qualifications, to be started in the fall semester of 2004. Applicants should have a PhD degree in an area of computer science/engineering. A strong commitment to excellence in teaching, and research involving students is expected. The department is located in a new building equipped with state-of-the-art equipment and facilities, and it enjoys strong support from local corporations. The department offers a Bachelor's degree for about 200 majors, and will launch its new Master's program, in spring, 2004. For more information about the university and the department, please visit http://www.cs.uca.edu. UCA is committed to excellence in undergraduate and graduate education, and fosters student-centered learning, technology enhanced pedagogies, and research experiences for all students. UCA is located in Conway, about 25 miles away from Little Rock, and has convenient access to numerous recreational sites. To apply, send a curriculum vita, statements of teaching and research interests, and contact information for at least three references to: Chair, CS Search Committee, Lewis Science Center Room 105, College of Natural Sciences and Mathematics, University of Central Arkansas, Conway, AR 72035. Electronic submissions may be sent to CSsearch@mail.uca.edu. Review will begin immediately and continue until the position is filled. UCA is an Equal Opportunity Affirmative Action Employer.

UTAH STATE UNIVERSITY, Department of Computer Science. Assistant Professor starting Fall 2004 specializing in applied computer science (Bio-informatics, Software systems, Multi-agent systems). See http://personnel.usu.edu/ for details. Qualified applicants should send a letter of interest, current curriculum vita (statements of research experience and interests, proposals written and funded, publications, and teaching experience), and the names of three references that the committee may contact, to: search@cs.usu.edu or Faculty Search Committee, Dept. of Computer Science, Utah State University, Logan, UT 84322-4205.

UNIVERSITY OF CALIFORNIA, SANTA BARBARA, Faculty Position: Department of Computer Science. The University of California Santa Barbara invites applications for faculty positions in Computer Science and Computer Engineering. The Department of Computer Science currently has 25 full-time faculty and approximately 200 graduate students (including approximately 120 Ph.D. students) involved in various research areas including digital libraries and databases, parallel and distributed systems, programming languages, networking, computer and network security, theory of computation, algorithms, computer vision, human-computer interaction, bioinformatics, and computational science and engineering. The Department of Computer Science is part of the expanding College of Engineering which encompasses over 130 faculty in various

Seattle University

Tenure-track faculty
(rank open)

The Department of Computer Science and Software Engineering invites applications for one or more tenure-track faculty positions to begin in September of 2004. Applicants are required to have a Ph.D. in either Computer Science, Software Engineering, Computer Security or a closely allied field. Candidates should be capable of teaching a broad range of computer science or software engineering courses. A specialty in computer security or information assurance will be a strong plus for candidates because of the department's intention of strengthening our offerings and program in that area. Seattle University offers Bachelor of Arts and Bachelor of Science degrees in Computer Science and a Master of Software Engineering degree (the country's first MSE degree, now in its 25th year).

Applications must include a curriculum vita, statements of teaching philosophy and research plans and a separate statement addressing how you could contribute to the Seattle University mission. We also require three letters of reference. Please send your application to Faculty Search Committee Chair, Department of Computer Science and Software Engineering, Seattle University, 900 Broadway, Seattle, WA 98122-4340 or by email to **CSSE_Search@seattleu.edu**. Information about Seattle University, a statement of its mission, and an expanded job announcement can be found at **www.seattleu.edu/home/about_seattle_university/** and **www.seattleu.edu/scieng/comsci/**. The department search committee will begin reviewing applications on January 5, 2004.

Seattle University is a Jesuit Catholic university enrolling more than 6,000 undergraduate and graduate students in eight colleges and schools. The university is an equal opportunity, affirmative action employer.

University of Massachusetts Boston

Department of Computer Science,
www.cs.umb.edu

Assistant Professor

The Computer Science Department at the University of Massachusetts Boston invites applications for Fall 2004 for one faculty position at the Assistant Professor level. We offer a BS, an MS with an emphasis on software engineering, and a Ph.D. in Computer Science. We seek to strengthen our research program significantly. Current faculty interests include databases, data mining, biodiversity informatics, natural language processing, computer and human vision, system modeling, algorithms, and theoretical computer science.

Strong candidates will be considered from any area of Computer Science but preference will be given to a candidate who does research in Software Engineering and is interested in assuming a lead role in teaching our graduate year-long required sequence in this discipline. Evidence of significant research potential and a PhD in Computer Science or a related area are required. We offer a competitive salary and a generous start-up package. Send cover letter, curriculum vitae, statements about research and teaching, and the names and email addresses of three references to:

search@cs.umb.edu.

Our campus overlooks Boston harbor; our faculty and students enjoy professional life in a center of academia and the software industry. For more information, visit us at http://www.cs.umb.edu.

Review of applications has begun and will continue until the position is filled.

UMass Boston is an Affirmative Action, Equal Opportunity, Title IX employer and strongly encourages women, members of all ethnic groups, and people with disabilities to apply.

University of Massachusetts Boston
www.umb.edu

engineering disciplines. UCSB is a major research institution, elected member of the Association of American Universities, as well as an integral part of the nine-campus University of California system. Graduate degrees in Computer Science are offered at the M.S. and Ph.D. levels. The Department and the College feature a highly collaborative and stimulating working environment. Department faculty lead several NSF ITR projects, and have recently been awarded two NSF IGERT grants for interdisciplinary graduate education and research, in computational science and engineering and in interactive digital multimedia. The total amount of 2002-03 extramural funding in the department was approximately 6.8 million dollars. Applicants should hold a doctoral degree in Computer Science or a related field; appointments are scheduled to begin in July 2004. Primary consideration will be given to candidates who apply by January 15, 2004, however, positions will remain open until filled. Send resume, research and teaching statements, and names and addresses of at least four referees to: Recruitment Committee, Department of Computer Science, University of California, Santa Barbara, CA 93106-5110. To apply by electronic mail, send the application to: recruit@cs.ucsb.edu. Additional information about the Computer Science Department may be found at http://www.cs.ucsb.edu. The department is especially interested in candidates who can contribute to the diversity and excellence of the academic community through research, teaching, and service. An EO/AA employer.

UNIVERSITY OF NOTRE DAME, Department of Computer Science & Engineering. The Department of Computer Science and Engineering (http://www.cse.nd.edu) invites faculty applications. Rank and area of speciality are open. Our faculty are actively engaged in high-quality research in a variety of areas, supported by NSF, DARPA, SRC, Air Force, HP/Compaq, and other organizations. The blend of small class size, low teaching load, and a "PhD only" graduate program allows us to emphasize true excellence in both teaching and research. Faculty must be committed to both quali8ty teaching and a strong externally-funded research program. Notre Dame is ranked 19th among national universities in the 2003 U.S. News and World Report. Notre Dame's heritage and values are unique among top-ranked national universities, resulting in a distinctive character of campus life. Screening of applications will begin immediately and continue until the positions are filled. Applicants should send cover letter, cv, a statement of research and teaching interests, and names and addresses of at least three references either to: facultysearch@cse.nd.

ARIZONA STATE UNIVERSITY

Ira A Fulton School of Engineering

Senior Faculty Position in Information Assurance and Security

The Ira A. Fulton School of Engineering at Arizona State University seeks to fill a position in the area of Information Assurance and Security at the rank of full professor or associate professor. The expected starting date for this position is August 16, 2004. The appointment of the successful candidate may be in the Department of Computer Science and Engineering, the Department of Industry Engineering, or both. Nominations and applications for this position are invited.

Applicants for this position are required to have a Ph.D. in computer science, industrial engineering, information systems, systems engineering or a related field. The applicant must have ability and /or accomplishment in establishing or leading research and teaching activities in the area of Information Assurance and Security (IAS), addressing prevention, detection and/or reaction mechanisms to assure trustworthy computers and networks and Quality of Service. Excellence in research and scholarly activity, teaching, and service in the area of computer science and engineering or industrial engineering appropriate to rank is required. Evidence of scientific, academic and organizational leadership, educational innovation, and demonstrative effectiveness in establishing industry partnerships are desired.

ASU is a major research university widely recognized as a rapidly emerging educational institution in the US. The main campus is located in the city of Tempe, in the metropolitan Phoenix area. This year the Fulton School of Engineering received a $50 million gift, which will provide funding for scholarships, fellowships, and research programs. Both the Department of Computer Science and Engineering and the Department of Industrial Engineering provide a stimulating and fast-growing environment of research and teaching, with ample opportunities for partnerships with high-technology industry and with emphasis on quality, leading-edge graduate and undergraduate education.

For more information about the Arizona State University, you are invited to visit the website http://www.asu.edu

Initial closing date for applications is February 10, 2004; if not filled, applications will be reviewed on a bi-weekly basis thereafter until the search is closed.

Application packages must include a cover letter, curriculum vitae, research and teaching statements, copies of the most important publications, and the names and addresses of four references. These packages must be sent by post to IAS Faculty Search Committee Chair, P.O. Box 878809, Fulton School of Engineering, Tempe, AZ 85287-8809.

Arizona State University is an Affirmative Action/Equal Opportunity Employer.

University of Kentucky

Department of Computer Science

The Department of Computer Science invites applications for a tenure-track position to begin in January or August 2004. The new faculty member will play an important role in the new Visualization and Vision Science Facility, a research unit with over 9,000 sq. ft., and over 5M in funding from state and federal agencies. The goal of the Facility is to advance state-of-the-art visualization technologies through basic and applied research programs. To that end, we are particularly interested in candidates with specialization in computer vision, computer graphics, geometric modeling, distributed multimedia, scientific computation, or human computer interaction. Applicants with credentials in other fields that will support the goals of the Facility will also be considered.

The Department of Computer Science offers BS, MS, and PhD degrees. Our faculty are actively involved in research in artificial intelligence, computer vision, geometric modeling, networking, cryptography, numerical analysis, operating systems and theory. The Department is funded by a number of external grants including DARPA projects, several NSF ITR awards, and an NSF Research Infrastructure award.

The Department is experiencing a period of dynamic growth, particularly in the areas of multimedia, computer networking, and distributed systems. In the previous six years, the Department has hired nine new faculty members, three of which are NSF CAREER awardees. New members of the computer science faculty will have access to significant laboratory space, generous start-up research funds, and will actively participate in the research activities of the Visualization and Vision Science Facility. A new building for the Department and the Facility is in the planning stage.

The University of Kentucky is an equal opportunity employer and especially encourages applications from women and minority candidates.

Completed applications consisting of a curriculum vita, statements of teaching and research interests, and the names of at least three references should be submitted to:

Chair, CS Search Committee
c/o Ms. Diane Mier,
773 Anderson Hall
University of Kentucky
Lexington, KY 40506.

Application review will begin on September 1, 2003 and continue until the positions are filled.

edu, or to Faculty Search Committee, Department of Computer Science and Engineering, 384 Fitzpatrick Hall, University of Notre Dame, Notre Dame, IN 46556.

UNIVERSITY OF CALIFORNIA, RIVERSIDE, MARLAN AND ROSEMARY BOURNS COLLEGE OF ENGINEERING, Faculty Positions in Computer Science & Engineering. The University of California, Riverside invites applications for tenure-track or tenured faculty positions in the Department of Computer Science and Engineering for the 2004-2005 academic year. Applicants must have a Ph.D. in Computer Science or in a closely related field. Areas of particular interest include: software engineering/programming languages, computational science, and graphics and visualization, but applications are welcome in all areas of Computer Science and Engineering. Senior candidates must have an exceptional research and teaching record. Junior candidates must show outstanding research and teaching potential. Salary will be competitive, and commensurate with appointment rank and qualifications. UC Riverside is the fastest-growing member of the ten-campus University of California System, widely regarded as the most distinguished system of public higher education in the United States. The CSE Department has now grown rapidly to 24 full-time faculties, and expects to grow to about 40 over the next five years. The Campus- and College-wide expansion programs include a new state-of-the-art building soon to house the CSE and EE Departments. Annual research expenditures in the College are close to \$352K/year per faculty. The CSE Department offers the B.S., M.S., and Ph.D. degrees. More information is available at http://www.cs.ucr.edu. The city of Riverside, located about 60 miles away from Los Angeles, offers easy access to mountains, Pacific Ocean beaches, cultural activities, shopping, and other attractions. Applications and inquiries should be sent to: Chair, Faculty Search Committee, Department of Computer Science and Engineering, University of California, Riverside, CA 92521-0304. (909) 787-5639 or by e-mail to: search@cs.ucr.edu. Applications must include curriculum vitae, list of publications, a statement of research and teaching objectives, and names and addresses of four references (junior candidates may have reference letters mailed directly to the department). Formal review of applications begins January 20, 2004. Early applications are encouraged, however, late applications will be considered until the positions are filled. The University of California, Riverside is an Equal-Opportunity/Affirmative-Action Employer.

ASSISTANT, ASSOCIATE OR FULL PROFESSOR
DEPARTMENT OF COMPUTER SCIENCE

Dept of Computer Science of the University of Texas at El Paso invites applications for two positions at all ranks for Fall 2004. We seek applicants in high-performance computing (e.g., computer systems design, performance evaluation, parallel/distributed computing, programming environments & compilers) & theory motivated by applications (e.g., interval computations, soft computing) & security. In exceptional cases, we will consider applicants in other areas of computer science. We offer BS & MS degrees in Computer Science & have a joint Ph.D. program with the Dept of Electrical & Computer Engineering. Approval of a distinct Ph.D. program in Computer Science is expected for fall 2004. Candidates must hold a Ph.D. in Computer Science or a closely related field. UTEP recently received an NSF ADVANCE award to further the recruitment, retention & advancement of female faculty in NSF-related disciplines. Applications from dual-career couples are encouraged. UTEP is a Carnegie doctoral-intensive university with an enrollment of 18,500 students, situated where the Rocky Mountains meet the Rio Grande. The campus reflects the beauty of the surrounding high desert. El Paso is a highly livable, bi-cultural community of 700,000 people that offers affordable homes & is a major meeting point for the United States & Latin America.

Send curriculum vitae, a statement of teaching & research interests & contact information for at least 4 professional references to: Faculty Recruiting 2004, Dept of Computer Science, UTEP, El Paso, TX, 79968-0518. Much more information about the department, including e-mail contacts for inquiries & electronic submission of applications is available at http://www.cs.utep.edu.

The University of Texas at El Paso does not discriminate on the basis of race, color, national origin, sex, religion, age, disability, veteran's status or sexual orientation in employment or the provision of services.

OUR TIME UTEP IS NOW

CALIFORNIA UNIVERSITY OF PENNSYLVANIA MATHEMATICS AND COMPUTER SCIENCE DEPARTMENT. Full-time tenure track position in Information Technology to begin Fall 2004. Duties will include teaching undergraduate and graduate courses in information technology and related programs, program and curriculum development, providing assistance in national accreditation of the Information Technology program, advising undergraduate and/or graduate students, participating in departmental/college/university/system-wide committees, and engaging in scholarly and professional activities including grant writing. For a complete description and application requirements see the Web site www.cup.edu. AA/EEO.

FLORIDA ATLANTIC UNIVERSITY, Department of Computer Science and Engineering, Faculty Recruitment Information. The Department of Computer Science and Engineering seeks applications for several tenure-track faculty positions at the Assistant and Associate Professor levels. The positions require a doctorate in computer science, computer engineering, or a closely related field. Applicants must show evidence of demonstrated teaching ability and research potential. The appointments will begin from August 2004. Review of applications will begin from February 15, 2004, and will continue until the positions are filled. Salary, fringe benefits, and teaching load are competitive. Florida Atlantic University, a member of Florida's State University System, is a multi-campus institution with its main campus located in Boca Raton, on the Atlantic coast, midway between West Palm Beach and Fort Lauderdale. The University has over 25,000 students and offers a variety of degree programs at all levels. The Department currently has 26 regular faculty, with other visiting and research faculty normally in residence. It offers bachelor's, master's and doctoral programs in computer science and in computer engineering. The Department also offers our undergraduate computer science program at our Davie campus (25 miles south of Boca Raton) and our Port St. Lucie campus (80 miles north of Boca Raton). Over 1600 undergraduate, 300 master's, and 50 doctoral students are enrolled in the Department. The Department has several well-equipped laboratories. It interacts closely with many high-tech companies located in the area (including Motorola, IBM, Siemens, Citrix, and others), which have helped provide state-of-the-art facilities. We have an active research program, with both federal and industrial sponsors. More information about the Department can be accessed through the Web at http://www.cse.fau.edu. Applicants should send a hardcopy resume and the names, phone numbers, and email addresses of at least three professional references, along with a cover letter specifying teaching and research interests and rank (Assistant, Associate) sought, to Chair, Department of Computer Science and Engineering, Florida Atlantic University, 777 Glades Road, Boca Raton, Florida 33431. Florida Atlantic University is an equal opportunity/equal access institution.

NEW JERSEY INSTITUTE OF TECHNOLOGY, Faculty Positions: Computer Science Department. The Computer Science Dept. at New Jersey Institute of Technology invites applications for tenure track / tenured faculty positions, beginning spring/fall 2004 in bioinformatics, networking/security, web technologies & software engineering. System-building experience will be a plus. Applicants must have PhD in computer science or related area. Senior applicants must have an outstanding record. Junior applicants should have shown research potential & commitment to excellence in teaching. Salaries are competitive. Dept. research interests incl. algorithms, computational biology & bioinformatics, computer vision, databases, parallel processing, simulation & modeling, software engineering & computer networking. The dept. has 27 faculty members & 1,400 students. The dept. offers BS, MS & PhD degrees. NJIT is located in Newark's University Heights, a multi-institutional campus shared with Rutgers University, the University of Medicine & Dentistry of New Jersey & Science Park. NJIT is close to several other universities, numerous pharmaceutical, telecommunications & financial companies & research labs thus offering excellent opportunities for collaboration, consulting & industry sponsored research. NJIT is minutes from New York City & close to the Jersey Shore which offer a wide range of cultural & leisure activities. Applicants can apply by sending their CV & a list of at least 3 references to: New Jersey Institute of Technology, attn: Personnel Box CS-AAP, University Heights, Newark, NJ 07102-1982 by mail, or by email to: hr@njit.edu. NJIT is an equal opportunity, affirmative action, equal access employer & especially encourages applications from minorities, women & persons with disabilities.

STATE UNIVERSITY OF NEW YORK AT BINGHAMTON, Department of Computer Science, The Thomas J. Watson School of Engineering and Applied Science, http://www.cs.binghamton.edu. Applications are invited for two or more anticipated tenure-track positions at the Assistant/Associate Professor level beginning in Fall 2004. Salary and startup packages are competitive. We are especially interested in candidates with specialization in Software Engineering, Programming Languages/Compilers, Information Security, Web-based Systems, Data Mining and areas related to systems development. Applicants must have a Ph.D. in Computer Science or a closely related discipline by the time of appointment. Strong evidence of research capabilities and commitment to teaching are essential. We offer a significantly reduced teaching load for junior tenure track faculty for at least the first three years. Binghamton is one of the four Ph.D. granting University Centers within the SUNY system and is nationally recognized for its academic excellence. The Department has well-established Ph.D. and M.S. programs, an accredited B.S. program and is on a successful and aggressive recruitment plan. Local high technology companies such as IBM, Lockheed-Martin, BAE and Universal Instruments provide opportunities for collaboration. Binghamton borders the scenic Finger Lakes region of New York. Send a resume and the names of three references to Professor Kanad Ghose, Department of Computer Science, State University of New York at Binghamton, P.O. Box 6000, Binghamton, New York 13902-6000. For candidates applying for Fall 04, first consideration will be given to applications that are received by February 27, 2004. Applications will be considered until the positions are filled. Binghamton University is an equal opportunity/affirmative action employer.

UNIVERSITY OF ALABAMA AT BIRMINGHAM (UAB), Department of Computer and Information Sciences, Assistant or Associate Professor Position, http://www.cis.uab.edu/. The University of Alabama Birmingham (UAB) Department of Computer and Information Sciences is seeking highly qualified candidates for a tenure track position at the Assistant or Associate Professor rank, beginning as early as Fall 2004. Scalable Data Mining, Knowledge Discovery, and Distributed Database Systems are of particular interest. Highly qualified candidates with a Ph. D. in Computer Science or a closely related field are encouraged to apply. A strong research background, including advanced knowledge, demonstrated research results, refereed publications, and potential for research funding in one or more of the focus areas are required. Commitment to excellence in teaching and service and interest in multidisciplinary collaboration with the biotechnology and medical research programs of the UAB campus are also very important. Send four references, a complete CV, a one-page research plan, and a one-page teaching plan via email to facapp@cis.uab.edu or via regular mail to: Faculty Search Committee, Department of Computer and Information Sciences, University of Alabama at Birmingham, 1300 University Blvd, Birmingham, AL 35294-1170. Interviewing for the position will begin as soon as qualified candidates are identified, and will continue until the position is filled. The University of Alabama at Birmingham is an equal opportunity/affirmative action employer.

UNIVERSITY OF FLORIDA, Department of Electrical and Computer Engineering. The Department of Electrical and Computer Engineering invites applications for two tenure-track or tenured faculty positions in the general area of Computer Engineering. Strong candidates in all areas of Computer Engineering are encouraged to apply for tenure-track positions at all ranks. The department is particularly interested in candidates with strong backgrounds and interests in one or more of the following areas: distributed computing, computer architecture, hardware/software systems design, high-performance computing, nanocomputing, adaptive computing, mobile computing, biologically-inspired computing and embedded systems. Applicants for tenured or senior positions must have distinguished

NUI MAYNOOTH
Ollscoil na hÉireann Má Nuad

DEPARTMENT OF COMPUTER SCIENCE

Website: www.cs.may.ie

Associate Professor Post

NUI Maynooth invites applications for the position of Associate Professor in Computer Science.

Applicants should have a strong track-record in research and teaching. Applications are welcome in all areas of research. The person appointed will play an active role in all aspects of the Department's activities, pursuing high-quality research, providing research leadership, and supervising and teaching students at both undergraduate and postgraduate level.

Salary Scale (new entrants):
€68,495 - €91,555 p.a. (8 points)

Prior to application, further details of the post may be obtained by writing to the **Personnel Officer, National University of Ireland, Maynooth, Maynooth, Co. Kildare, Ireland. Confidential Fax: +353-1-7083940 Email: personnel@may.ie**

Applications (including a full CV, the names, email and postal addresses, telephone and fax numbers of three referees, and a personal statement) should be forwarded to the **Personnel Officer**, to arrive no later than **27 February, 2004.**

records. Candidates should be committed to excellence in teaching and research. All applicants should hold a PhD degree. Salary and support are competitive and commensurate with experience and background. Successful candidates will have the opportunity to work with highly qualified faculty, excellent students and well-equipped laboratories from both the Department of Electrical and Computer Engineering and the Department of Computer and Information Science and Engineering of the University of Florida. More information is available at http://www.ece.ufl.edu and http://www.cise.ufl.edu. Candidates should send their resume, research and teaching plan, cover letter stating the desired position, and names and addresses of at least three references to Dr. José Fortes, Professor and BellSouth Eminent Scholar, C/O Computer Engineering Faculty Search Committee, Dept. of Electrical and Computer Engineering, P.O. Box 116200, 339 Larsen Hall, University of Florida, Gainesville, Florida 32611-6200. Email applications are encouraged. They should be sent in pdf format to cesearch@acis.ufl.edu. The committee will begin reviewing applications on January 28th, 2004 and will continue to receive applications until the positions are filled. The University of Florida is an Affirmative Action Employer and women and minorities are encouraged to apply. According to Florida law, applications and meetings regarding applications are open to the public on request.

CALIFORNIA STATE UNIVERSITY, FULLERTON. The Department of Computer Science at CSU Fullerton invites applications for tenure-track position(s) for the academic year starting August 2004. The department's instructional programs lead to bachelors and masters degrees. CSUF is an urban university located in Orange County, surrounded by high technology industry and a diverse community. Faculty members regularly teach all levels of computer science. The Department values teaching, research, and publishing. A Ph.D. in Computer Science or a related field is required. All candidates must have a publication record in computer science or a related field. Candidates with specialized interest in computer networks, pervasive computing, data security, and software engineering are preferred, but interest in other areas such as graphics and e-commerce will also be considered. This is a tenure-track position. Salary is competitive and commensurate to rank and experience. Review of applications will commence immediately and will continue until the positions are filled. Please send a letter of application, curriculum vitae, and list of three references with contact information to: Dr. Dimitri Michalopoulos, Chair, Department of Computer Science, CS-552, California State University, Fullerton, Fullerton CA 92834-6870 or email csoffice@ecs.fullerton.edu. Cal State Fullerton is an Equal Opportunity/Title IX/ ADA Employer.

POHANG UNIVERSITY OF SCIENCE AND TECHNOLOGY (POSTECH), Faculty Positions. The Department of Computer Science and Engineering (CSE), Pohang University of Science and Technology, has several faculty openings at all ranks beginning in March 2004. All areas of computer science and engineering are welcome. A Ph.D. or its equivalent is required. Successful candidates are expected to conduct outstanding research and be committed to quality teaching both at undergraduate and graduate levels. Further information about the CSE program and positions is available at www.postech.ac.kr/department/cse. Applicants will be reviewed as received and will continue until filled. Applicants should send a curriculum vitae, three letters of reference, and any supporting documents to Search Committee Chair, Department of Computer Science and Engineering, Pohang University of Science and Technology, San 31 HyoJa Dong, Pohang 790-784, Republic of Korea.

CALIFORNIA UNIVERSITY OF PENNSYLVANIA MATHEMATICS AND COMPUTER SCIENCE DEPARTMENT. Full-time tenure track position in Computer Science to begin Fall 2004. Duties will include teaching undergraduate and graduate courses in computer science and related programs, program and curriculum development, providing assistance in national accreditation of the Computer Science programs, advising undergraduate and/or graduate students, participating in departmental/college/university/system-wide committees, and engaging in scholarly and professional activities including grant writing. For a complete description and application requirements see the Web site www.cup.edu. AA/EEO.

MORAVIAN COLLEGE. Moravian College invites applications for a tenurable-track position at the assistant professor level in computer science to begin Fall Term 2004. Applicants should have a Ph.D. in computer science and a broad range of expertise and interests. The successful candidate will have strong interpersonal skills and a primary commitment to undergraduate teaching, scholarship, and curriculum development for computer science in a liberal arts context. Particularly desirable areas of expertise include graphics and visualization, theoretical or applied artificial intelligence, and scientific computing, but all areas of computer science will be considered. Teaching responsibilities will include undergraduate courses for the computer science major as well as courses for non-majors. Moravian College faculty are encouraged to participate in interdisciplinary teaching. Applications should be received by February 20. Send resume, transcripts, a statement of teaching philosophy, and three letters of recommendation to Alicia Sevilla, Chair, Department of Mathematics and Computer Science, Moravian College, 1200 Main Street, Bethlehem, PA 18018-6650.

VIRGINIA TECH. The Bradley Department of Electrical and Computer Engineering invites applications for a tenure-track full professor position in computer engineering. Named professor opportunities exist for an exceptionally qualified candidate. Some areas of interest are: software engineering, embedded systems, and high-performance computing. The ECE Department has outstanding opportunities for faculty to develop highly visible research programs. All candidates are expected to conduct research and teach undergraduate and graduate courses. Positions are also available in other areas, including microelectronics and power electronics. For a detailed information, please visit the department website at www.ece.vt.edu. An equal opportunity/affirmative action employer.

DATABASE ADMINISTRATOR. Implement, monitor and revise existing database. BS in Comp Sci, related field or equiv req. Send ad w/resume to: La Tanya Odom, SunAmerica Inc., 21650 Oxnard St., Woodland Hills, CA 91367.

LOUISIANA STATE UNIVERSITY, ASSISTANT/ASSOCIATE/FULL PROFESSOR, (Computer Engineering/Tenure-track/One or more positions), Electrical and Computer Engineering. The Department of Electrical and Computer Engineering at Louisiana State University invites applications for one or more tenure-track positions in Computer Engineering at all levels available August 2004 or until positions are filled. Though applicants from all areas of Computer Engineering will be considered, of particular interest are applicants with research interests in computer architecture (including microarchitecture, memory systems and special-purpose architectures), design automation (including system-on-chip design and behavioral synthesis), and networking (including traffic engineering and network security). Required Qualifications: Ph.D. or equivalent degree in Electrical or Computer Engineering or related field; potential for excellence in teaching and research. The positions involve teaching graduate and undergraduate courses in electrical or computer engineering and research in areas of individual interests. Salary is competitive and commensurate with qualifications and experience. Release time and resources are provided in order to enhance the development of a quality research program. Opportunities for summer support are available. In addition to the resources in the department, the faculty has access to a 1024-processor cluster. Application deadline is March 1, 2004, or until candidate is selected. Please send your resume (including e-mail address), names of at least three references, and a statement of teaching and research interest, in electronic form to kemin@ece.lsu.edu (preferred) or mail your application package to: Dr. Kemin Zhou, Interim Chair, Electrical and Computer Engineering, Louisiana State University, Ref: Log# 0518, Baton Rouge, LA 70803. LSU IS AN EQUAL OPPORTUNITY/EQUAL ACCESS EMPLOYER.

UNIVERSITY OF PITTSBURGH, Department of Computer Science. The Department of Computer Science at the University of Pittsburgh is initiating a search for a tenure track position at the Assistant Professor level effective September 2004, pending budgetary approval. Outstanding candidates are sought in the area of Computer Systems, with particular considerations given to candidates working on Embedded Systems, Computer Architectures, Distributed Systems and System Security. Responsibilities include research, supervision of graduate student research (PhD and MS), and graduate and undergraduate teaching. Candidates should have a PhD in Computer Science and demonstrate exceptional research potential and teaching ability. Candidates should send a curriculum vita, a statement of research and teaching interests, and names and addresses of at least three references to Professor Rami Melhem, Chair, Department of Computer Science, University of Pittsburgh, Pittsburgh, PA 15260. In order to expedite the review process, candidates should also complete the web-based form at http://www.cs.pitt.edu/recruiting. Please direct your inquiries to faculty-search@cs.pitt.edu. Applications must be received by February 14, 2004 to ensure full consideration. The Department provides a stimulating environment for research and teaching that results in strong graduate and undergraduate programs. The Department already has a strong group in the Computer Systems area and is a partner in the Computer Engineering program, which is an interdisciplinary program between the Computer Science and the Electrical Engineering departments. Departmental resources include extensive computing facilities of over 400 workstations and personal computers with multimedia capabilities and specialized networks and devices. Faculty members also have network access to additional high performance computing platforms provided by the general computing facilities of the University as well as by the Pittsburgh Supercomputing Center (of which the University of Pittsburgh is a founding member). For further information about the Department please see http://www.cs.pitt.edu. The University of Pittsburgh is an Affirmative Action, Equal Opportunity Employer. Women and members of minority groups under-represented in academia are especially encouraged to apply.

MONMOUTH UNIVERSITY, Computer Science Department. Applications are invited for a tenure track position, at the Assistant or Associate Professor level, starting September, 2004. A Ph.D. in Computer Science, or related discipline, is required. Tenure obligations include a strong commitment to teaching across the curriculum in the undergraduate and graduate programs, scholarly contributions, and service to the University at all levels. Preference will be given to those with teaching experience and/or prior practice in the design and development of software systems, distributed computing, client-server architecture, and operating system design. Documentation should emphasize experience in teaching at all levels, and cite references to scholarly activities. The position also requires participation in departmental course and curriculum development efforts. Questions should be directed to frederic@monmouth.edu. Applicants should send a cover letter and include a statement of teaching and research experience and interests, a resume, and the names of three professional references to: Chair, Computer Science Search Committee c/o Francis C. Lutz, Dean-School of Science, Technology and Engineering Monmouth University, West Long Branch, NJ 07764. Interviews start after January 15, 2004, and will continue until the position is filled. See http://www.monmouth.edu for information about the university, school, and department. Monmouth University is an Equal Opportunity Affirmative Action Employer.

GEORGETOWN UNIVERSITY, Principal Technologist. Reporting to the Vice President for Information Services and Chief Information Officer, the Principal Technologist (PT) provides institutional leadership in the strategic application of information technology to academic, research and administrative computing. The PT supports and offers guidance to the CIO and other leaders in the use of technology across the University. Responsibilities include: fostering the innovative and effective application of advanced technologies to basic research and the development of scholarly information systems; supporting Academic and Administrative leaders in the pursuit of research grants and contracts; representing the University in external technology sharing and standards groups; co-chairing the University Information Services Technology and Operations Committee and guiding the Committee through the production of an annual technology master plan and architecture. Minimum Qualifications include a Bachelor's Degree or equivalent with relevant course work in Computer Science, Information Systems or related field. Graduate degree desirable. Experience also required in the in the application of computing technology in support of academics and application of computing technologies and concepts to scientific basic research. Applicants must have excellent written and verbal skills, proven management skills, demonstrated capacity in presentation to large groups on complex subject matter and the ability to develop technical papers, proposals and project plans. Please apply online at www.georgetown.edu/hr/jobs. Job # 2003 0566.

MEMORIAL UNIVERSITY OF NEWFOUNDLAND, TIER II CANADA RESEARCH CHAIR IN ASIC DESIGN. The Faculty of Engineering and Applied Science invites applications for a Tier II Canada Research Chair in the area of Application-Specific Integrated Circuit (ASIC) Design. The appointment will be tenure-track at the Assistant or Associate Professor level. Candidates should demonstrate exceptional potential to build an internationally reputed research program in ASIC design. The Chair will have opportunities to collaborate with an anticipated Chair in Photonics. Information on the application procedure, qualifications, and expected activities of the Chair may be found at www.engr.mun.ca/Jobs/asic.html. Applications will be reviewed beginning on February 20, 2004. Memorial University of Newfoundland is committed to employment equity and encourages applications from qualified women and men, visible minorities, aboriginal people and persons with disabilities. There is no restriction with regard to nationality or residence.

WASHINGTON STATE UNIVERSITY, Director, School of Electrical Engineering and Computer Science. The School of Electrical Engineering and Computer Science at Washington State University invites applications and nominations for the position of Director of the School, to be filled August 16, 2004. A Ph.D. degree in electrical engineering, computer engineering, computer science or a related discipline is required. Candidates must possess a national and/or international reputation for scholarly activities, a commitment to excellence in undergraduate and graduate education, a successful record of obtaining external funding, proven fiscal proficiency, and outstanding communication and leadership skills. The initial appointment is for a four-year period with the possibility of reappointment. The position is a 12-month position with tenure, and salary is negotiable, commensurate with qualifications and experience. The School of Electrical Engineering and Computer Science is the largest of five engineering departments in the College of Engineering and Architecture at Washington State University. Programs are offered at all four University campuses: the main campus at Pullman and three urban campuses at Tri-Cities, Spokane, and Vancouver. The School has 40 full-time faculty, and student enrollment consists of 860 undergraduates and 140 graduates. The School awards baccalaureate through doctoral degrees in electrical engineering, computer engineering and computer science. The total budget for the School exceeds $8 million per year. The School has three endowed chairs and four distinguished professorships in electrical engineering, computer engineering and computer science. The School participates in two NSF UCRC centers, one for Design of Analog-Digital Integrated Circuits and the other for Power Engineering. The School maintains active research programs in many leading edge technologies and has excellent facilities. Additional information may be found at the School's website http://www.eecs.wsu.edu. Applications and nominations should be sent to: EECS Director Search Committee, School of Electrical Engineering and Computer Science, Washington State University, Pullman, WA 99164-2752. Screening of applications will begin February 15, 2004. WSU is an EEO/AA Employer. Protected group members are encouraged to apply.

RENSSELAER POLYTECHNIC INSTITUTE, Computer Science. The Department of Computer Science at Rensselaer Polytechnic Institute invites applications for one or more expected tenure-track positions at the Assistant Professor level. Strong candidates at all professional levels and areas of computer science will be considered, however, the Department has special interest in candidates working in the areas of architecture, bioinformatics, cryptography, data mining and visualization, human-computer interaction, and security. Applicants should hold a Ph.D. in Computer Science or in a closely allied field, have substantial research accomplishments for the professorial level sought, and demonstrate a strong commitment to teaching. Rensselaer, under the direction of its president, plans to double its research program in the next five years (see http://www.rpi.edu/web/President/Plan/index.html). Major research initiatives in information technology and biotechnology are integral components of this plan. The faculty of the Department of Computer Science are strongly encouraged to participate in collaborative research at the forefront of both of these initiatives. The department currently has 22 full-time faculty members and offers BS, MS and Ph.D. degrees. It has excellent computing facilities that support a vigorous growing research program (approximately $2 million per year) that currently includes 6 active NSF CAREER awards. Applicants should submit a vita with a list of publications, a statement describing current and planned research, and a statement describing teaching philosophy to the address listed below. Candidates should also arrange to have at least three letters of recommendation sent to the same address. To ensure full consideration, all application materials must be reviewed by the committee by March 15, 2004. Faculty Search Committee, Attn: Jacky Carley, Department of Computer Science, Rensselaer Polytechnic Institute, 110 8th Street, Troy, NY 12180-3590. Rensselaer Polytechnic Institute is an Equal Opportunity/Affirmative Action Employer. Women and minorities are strongly encouraged to apply.

PORTLAND STATE UNIVERSITY, Positions in Computer Science. Portland State University invites applications for two tenure- track, nine-month faculty positions at all levels to begin Fall 2004. Applicants for senior positions are expected to have a nationally recognized research program and to have demonstrated the ability to generate external funding; applicants for junior positions must show potential for future external support. The department is looking for faculty in the areas of networking, software engineering, computer security and databases, but excellent applicants in all areas will be considered. The minimum qualifications for the position include an earned Ph.D. The faculty member will teach undergraduate and graduate classes; develop program, course and laboratory resources in Computer Science; maintain scholarly activity in funded research and publications; provide professionally related public service; advise students, and support University activities through committee service. The department currently has nineteen faculty and offers a ABET accredited B.S., an M.S., and a Ph.D. in Computer Science. Our teaching loads give faculty time to maintain funded research programs and to collaborate with local industry. The department currently serves approximately 400 undergraduates and 120 graduate students. Further information about the department is available at http://www.cs.pdx.edu. Portland State University is located in downtown Portland, Oregon. Portland's "Silicon Forest" is one of the major software/hardware development centers in the country. Within the Portland metropolitan area are over 200 hardware and software organizations, including the world's largest campus of Intel, the Open Source Development Lab, the world headquarters of Tektronix, In Focus, and Mentor Graphics, major R&D sites of Oracle and Informix, a major division of IBM, along with many other companies ranging from major corporations to software start-ups. The University is approximately one hour from magnificent beaches and from year-round skiing. Review of applications will begin immediately and will continue until the positions are filled. Please send a statement of research and teaching interests, a vita, and the names of three references to: Faculty Search Committee, Department of Computer Science, Portland State University, PO Box 751, Portland, OR 97207-0751 or e-mail recruitieee@cs.pdx.edu. Portland State University is an Affirmative Action, Equal Opportunity institution and, in keeping with the President's diversity initiative, welcomes applications from diverse candidates and candidates who support diversity.

SYSTEMS ANALYST. Plan & develop new computer systems. MSc in Computer Engr or related field req. Send ad w/resume to: Celetron, 2125-B Madera Rd, Simi Valley, CA 93065.

UNIVERSITY OF WESTERN AUSTRALIA, Lecturer/Senior Lecturer (REF:291), Electronics (VLSI and Digital Systems), School of Electrical, Electronic and Computer Engineering. SALARY RANGE: Lecturer (Level B) $55,479 – $65,883 p.a. Senior Lecturer (Level C) $67,962 – $78,366 p.a. Applications are invited for the above tenurable position. Applicants must play an active role in digital systems research, must have a strong commitment to supervision of postgraduate and undergraduate students and work with one of the major research groups in the School. Applicants must have a PhD and should be able to demonstrate their potential as a lecturer and researcher. The School has very active research programmes relating to the following areas: computer systems, networking, microelectronics, photonics, communications, biomedical engineering, signal processing, robotics, control systems and power engineering. Information on current research programmes in the School can be found at http://www.ee.uwa.edu.au/research. Applicants for the position should have appropriate engineering experience in university and/or industry in areas relevant to digital design and implementation. Duties include development and delivery of lectures and laboratory courses in the relevant topics of the School's undergraduate courses in the area of VLSI and digital systems design and implementation. Candidates with industrial experience are encouraged to apply. Applicants must address the selection criteria and applicants with teaching experience are requested to submit a teaching portfolio as part of their application. For further information regarding the position please contact Maureen Russell, School Manager on 9380 3134 or email maureen@ee.uwa.edu.au or access the web link http://www.ee.uwa.edu.au/research. CLOSING DATE: Friday, 30 January 2004. Located adjacent to the picturesque banks of the Swan River, The University of Western Australia offers an attractive benefits package including generous superannuation, fares to Perth (if applicable) for appointee and dependants along with a removals allowance, generous leave provisions and an enviable working environment. These and other benefits will be specified in the offer of employment. APPLICATION DETAILS: For copies of the selection criteria please access the website below. Applicants must address the selection criteria. Written applications quoting the reference number, personal contact details, qualifications and experience, along with contact details of three referees should be sent to Director, Human Resources, The University of Western Australia, M350, 35 Stirling Highway, Crawley WA 6009 or emailed to jobs@uwa.edu.au by the closing date. http://jobs.uwa.edu.au/. Prime Minister's Employer of the Year in Higher Education 2002 and 2003.

EMBEDDED SOFTWARE ENGINEERING MANAGER. Leads a team of s/w engineers in development of real-time embedded systems and Telecommunication Management Network (TMN) applications; provides technical directions in design, development, documentation and quality of software components. Must be knowledgeable in ITU/TMN, Bellcore, ANSI, NMF standards and procedures. Uses TMN tools to develop element management systems, embedded control systems and protocol stacks. Experienced in formal software development lifecycle processes and quality management (ISO 9000). Must be a B.S. with minimum 5 years experience in Unix, pSOS, VxWorks, C/C++ Cross Compilers, Marben OSIAM stack, DSET's TMN Toolkit Suite and LNP packages. Prior experience in Number Portability and Broadband systems is desirable. Apply to Recruitment Manager, NE Technologies Inc., 5085 Avalon Ridge Pkwy Suite 100, Norcross, GA 30071.

TEXAS TECH UNIVERSITY. The Department of Computer Science invites applications in all areas of Computer Science for one or more tenure track positions at all levels for the academic year 2004-05. We anticipate having faculty openings at the satellite graduate campus in Abilene, Texas. Specific areas of need include software engineering, formal methods, and high dependability software. Applicants must have a Ph.D. degree in computer science or a closely related field. Faculty are expected to teach existing graduate and undergraduate courses, develop new courses, and contribute to the research mission of the university. Texas Tech University offers a Ph.D. and an M.S. in Computer Science and an M.S. in Software Engineering. We offer competitive salaries, a friendly and cooperative environment, and excellent research facilities. State-of-the-art two-way video instructional facilities allow the Lubbock and Abilene sites to exchange course offerings and to conduct collaborative faculty and committee meetings. More information on the department is available at http://www.cs.ttu.edu/. Applicants should send curriculum vitae, including a two-page research and teaching statement, and the names of at least three references. Electronic submission of application materials in the form of postscript or PDF is preferred. Electronic submission should be sent to Mysti.Digby@coe.ttu.edu or hardcopy can be mailed to Dr. Susan A. Mengel, Chair Faculty Search Committee, c/o Ms. Mysti Digby, Department of Computer Science, Texas Tech University, Box 43104, Lubbock, TX 79409-3104. Review of applications will begin as soon as they are received. Applications will be accepted until the positions are filled. Candidates must be currently eligible to work in the United States. Texas Tech University is an equal opportunity/affirmative action employer and actively seeks the candidacy of women and minorities.

SYSTEMS ANALYST. Must have B.S. or Foreign equivalent in Engineering, Computer Science or related. Must have 9 months of experience in the job offered or as a Software Engineer. Duties are to convert data from project specifications and statements of probability and procedures; develop software applications, prepare work flowcharts and diagrams using business and engineering analysis, math models, computer languages and databases; resolve questions of programming, data input/output requirements, and inclusion of internal checks and controls; convert detailed logical flowcharts to software code; input test data into computer systems; monitor performance aspects such as data propagation and optimization; develop acceptance criteria, install software, train users and monitor results; prepare reports, user and instructional manuals; and analyze, review and rewrite programs to increase operating efficiency or to adapt programming to new requirements. Must have Oracle and Visual Basic experience and be will to travel frequently. Qualified candidates must send resumes to S. Vishwanathan, VSoft Corp., 6455 E. Johns Xing, Ste. 450, Duluth, GA 30097.

METROPOLITAN COLLEGE OF BOSTON UNIVERSITY, Chairman, Science and Engineering. Boston University's Metropolitan College seeks a Chairman for the Science and Engineering Program (SEP) for September 2004. SEP is an innovative program that provides promising students who want to major in science or engineering, with an alternative to direct admission into the University's four-year programs. See http://www.bu.edu/met/programs/undergraduate/science-engineering/index.html. The ideal candidate will hold a doctorate in a technical or scientific field, be familiar with engineering, and appreciate the challenges confronting aspiring young engineers and scientists. The Chairman's duties include: overseeing admissions, meeting parents, managing the curriculum, and working with other University units. The Chairman will hold a faculty appointment. Send an application letter, a resume and three references to Dr. Eric Braude, Chairman, SEP Search Committee, Metropolitan College, Boston University, 808 Commonwealth Avenue, Boston MA 02215. Deadline: January 31, 2004. Boston University is an Equal Opportunity Affirmative Action Employer.

SOUTH DAKOTA STATE UNIVERSITY, Software Engineering Position. The department of Electrical Engineering, Computer Science, and Software Engineering, South Dakota State University, invites applicants for a tenure-track faculty position in Software Engineering to begin August 15, 2004 (www.engineering.sdstate.edu/~eeweb/). As part of the newest program in the college, the successful candidate will provide significant direction for the growth and development of the Software Engineering program. Applicants must have an earned Ph.D. in Software Engineering, Computer Engineering, Computer Science, or

a closely related field and the capability to successfully conduct research in software engineering. Experience with embedded systems research/ design, industry, ABET accreditation, and evidence of successful teaching is desired. For a full list of qualifications, visit the website above. To apply, send a cover letter, curriculum vitae, names of three references with phone numbers, unofficial transcript, and application (available at www.engineering.sdstate.
edu/~eeweb/se_app.pdf) to: SE Search Committee Chair, EE/CS/SE Department, South Dakota State University, P.O. Box 2220, Brookings, SD 57007. Completed application postmarked by March 5, 2004 are assured of receiving full consideration. Continuous recruitment until the positions are filled. Inquiries: (605) 688-4526. SDSU is an AA/EEO employer and encourages applications from women and minorities. ADA Accommodations: TTY (605) 688-4394.

GRAPHICS PROGRAMMER. Out of The Blue Design, located in San Francisco, CA, seeks a graphics programmer for ad-vanced internet graphics design position. Requires exp. with 3D graphics design and mathematical modeling. Must possess 4 years internet applications development exp. and B.S. in math or comp. science. Fax resume to 415-546-7921, or tel. 415-348-8711, ext. 203 for info.

UNIVERSITY OF CALIFORNIA, SANTA CRUZ, Associate or Full Professor, Computer Engineering #30. The Department of Computer Engineering at the University of California, Santa Cruz (UCSC) invites applications for a tenured (Associate or Full Professor) faculty position. We seek outstanding applicants in the following areas of Computer Engineering: embedded systems, computer networks, computer system design, and robotics. The campus is especially interested in candidates who can contribute to the diversity and excellence of the academic community through their research, teaching and service. UCSC is the UC campus nearest to Silicon Valley and has close research ties with the computer industry. Applicants should submit a CV, a statement of research plans, a statement of teaching interests, URLs of selected reprints, and ensure that at least three confidential letters of recommendation are sent directly, by the deadline of February 10, 2004. We strongly encourage electronic submission of your materials. Directions are provided at http://www.soe.ucsc.edu/ jobs/. All letters will be treated as confidential documents; please direct your references to UCSC's confidentiality statement at http://www2.ucsc.edu/ahr/ policies/confstm.htm). Alternatively, application materials may be mailed to: Computer Engineering Search, Baskin School of Engineering, University of California, Santa Cruz, CA 95064. UCSC is an EEO/AA/IRCA Employer.

UNIVERSITY OF WISCONSIN-MILWAUKEE, Faculty Recruitment in Computer Science. The Computer Science Program in the Department of Electrical Engineering & Computer Science is continuing its development and growth. Our Program has established a very good record in recruiting outstanding junior faculty and in providing them with a nurturing, as well as stimulating environment for their career development. Several of our faculty have been the recipients of the NSF EARLY CAREER Awards. The Computer Science faculty is engaged in research in many areas including Artificial Intelligence, Theory, Cryptography and Data Security, Distributed Systems, Networks, Programming Languages, Software Engineering, and Medical Informatics. The Program is also engaged in collaborative efforts with several academic units at our University as well as other institutions including the Medical College of Wisconsin. We invite applications for tenure faculty positions from strong candidates in the following areas of Computer Science: Artificial Intelligence, Computer and Data Security, Data Bases, Computer Networks, and Medical Informatics. We are particularly interested in recruiting in the areas of Computer and Data Security, and in Medical Informatics. Recently our university received a major NSF collaborative grant in Cyber Security that involves several university of Wisconsin campuses. We are the lead institution in this project. All candidates should have a demonstrated promise for excellence in research and in teaching. Candidates for senior positions should have excellent research credentials. Our University is located in a pleasant residential neighborhood of Milwaukee close to the shores of Lake Michigan. Our metropolitan location facilitates each interactions with many industries and affords numerous cultural and recreational activities. Applicants should send a hard copy of a vita by post or Fax, along with a statement of plans for research and teaching. We also request that at least three references be asked to send letters to: Faculty Recruitment Coordinator for Computer Science, Department of Electrical Engineering & Computer Science, University of Wisconsin-Milwaukee, PO Box 784, Milwaukee, WI 53201-0784. E-mail: recruit@cs.uwm.edu (for inquiries only), Fax: 414-229-6958. Evaluation of all candidates will begin in Spring 2004 and will continue until the position is filled. Women and minority candidates are strongly encouraged to apply. Additional information about our Program may be obtained by visiting our website at http://www.cs.uwm.edu. UWM is an equal opportunity institution committed to diversity.

ARCHITECTURE TECHNICAL SUPPORT ANALYST. BSc in Computer Science related or equiv req. Send ad w/resume to: SunAmerica Inc., 1 SunAmerica Center, Los Angeles, CA 90067.

RESEARCH STAFF SCIENTIST (materials science). Full time position. Job duties include: providing on-site technical expertise to design and assemble a RF-plasma sputtering system; investigating materials based on quantum mechanical and optical properties; developing growths of semiconductor thin-films on optical substrates; developing domain-specific characteristics of nano-scale optical films w/array of microelectronics structures; developing characterization of thin-films for shifts of spectral lines and refractive indices under influence of applied fields; determining optical characteristics of fabricated nano-structures w/applied field; assisting in the setup of Atomic Force Microscope (AFM) Manipulator; characterizing electrical & magnetic properties of bio-element, such as ferritin, for device applications; developing bio-devices for bio-element manipulation and bio-material based power generation; preparing and submitting quarterly and final reports. Min.req.: Ph.D. in Materials Science ; Master's and Bachelor's Degree in Physics (or equiv); proven record & publications of scientific research in atomic layer growth; machine design & construction knowledge of Electron Microscopy & E-Beam Lithography; machine design knowledge & research publications in Scanning Probe Microscopy; substantial coursework in integrated circuit micro fabrication; knowledge on established research capability in bio-nano devices and with related peer reviewed publications. For consideration, send resume to: STC, 10 Basil Sawyer Dr., Hampton VA 23666. No phone calls. EOE.

NETWORK/TELECOMMUNICATIONS MANAGER. Qualifications: BS in Comp. Science or related field, MCSE Certified preferred, working experience within a complex MS NT Network, Security, MS Exchange, SNA, IIS, Great Plains, Cisco and Active Directory. Knowledge of various models of Nortel and Meridian phone switches is required. Mail resume to Attn: Bruce Wantuch YWCA, 1915 N. Dr. Martin Luther King Jr. Dr., Milwaukee, WI 53212 or e-mail to jobs@ywcamilw.org.

JANUARY 2004

Boldface denotes advertisements in this issue.

Advertising Sales Representatives

Mid Atlantic (product/recruitment)
Dawn Becker
Phone: +1 732 772 0160
Fax: +1 732 772 0161
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)
Barbara Lynch
Phone: +1 401 739 7798
Fax: +1 401 739 7970
Email: bl.ieeemedia@ieee.org

Northwest (product)
Peter D. Scott
Phone: +1 415 421 7950
Fax: +1 415 398 4156
Email: peterd@pscottassoc.com

Southeast (product/recruitment)
C. William Bentz III
Email: bb.ieeemedia@ieee.org
Gregory Maddock
Email: gm.ieeemedia@ieee.org
Jana Smith
Email: jsmith@bmmatlanta.com
Phone: +1 404 256 3800
Fax: +1 404 255 7942

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)
Bill Wageneck
Email: bill.wageneck@wageneckassociates.com
Josh Mayer
Email: josh.mayer@wageneckassociates.com
Phone: +1 972 423 5507
Fax: +1 972 423 6858

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

Japan
German Tajiri
Phone: +81 42 501 9551
Fax: +81 42 501 9552
Email: gt.ieeemedia@ieee.org

Europe (product)
Hillary Turnbull
Phone: +44 (0) 1875 825700
Fax: +44 (0) 1875 825701
Email: impress@impressmedia.com

Europe (recruitment)
Penny Lee
Phone: +020 7405 7577
Fax: +020 7405 7506
Email: reception@essentialmedia.co.uk

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://computer.org
advertising@computer.org

Advertising Personnel

Marion Delaney
IEEE Media, Advertising Director
Phone: +1 212 419 7766
Fax: +1 212 419 7589
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

IMSI Updates Popular CAD Tool

A new version of IMSI's CAD software is now available for electrical and mechanical engineers, architects, and designers. TurboCAD Professional 9.5 integrates 2D capabilities with 3D modeling in a Windows environment. Advanced tools specific to mechanical and architectural projects complement the product's general design tools, and its photorealistic-rendering engine is designed to produce professional results. TurboCAD Professional 9.5 costs $695; www.imsisoft.com.

Immersive Data Visualization for PC Users

Advanced Visual Systems has released a new edition of its AVS Express data visualization system, which is designed to help scientific, engineering, medical, telecommunications, and environmental researchers gain insight from all types of complex data. You can use existing AVS Express applications on high-performance PCs and graphics clusters, and experience the models in immersive environments such as the SGI RealityCenter; www.avs.com/express.

Zero G Releases InstallAnywhere 6

Continuing its expansion into the enterprise software deployment market, Zero G Software has updated its multiplatform software-deployment system. InstallAnywhere 6 offers the ability to install native packages on Linux, Solaris, and HP-UX as well as do media spanning for large installers; www.zerog.com.

CodeFutures Updates Database Persistence Tool

FireStorm DAO 2.0 from CodeFutures is for Java programmers developing applications that rely on databases. The product generates the business logic for accessing relational databases in both service-oriented and object-oriented architectures. FireStorm DAO 2.0 costs $295; www.codefutures.com/firestorm.

New Streaming-Media System from Xiran

Xiran has announced an entry-level extension to its line of streaming-media-acceleration products. The DPA-1200HS is a digital-media streaming system designed to distribute digital content across IP networks for bandwidths of up to 100 Mbps. The DPA-1200 accelerator helps customers reduce the cost of deploying streaming-media systems by enabling high-throughput streaming content to run on lower-cost hardware platforms. The DPA-1200HS kit starts at $9,990; www.xiran.com.

IMSI's TurboCAD Professional 9.5 lets designers build everything from complex 3D parts to architectural environments using simple profiles and shapes manipulated with the ACIS 10 modeling engine.

Parasoft Offers Software for Web Services Security

SOAPtest from Parasoft is a software package designed for developers wanting to build security into SOAP-based Web services. With the SOAPtest RuleWizard feature, developers can establish and verify specific patterns contained in SOAP messages. As messages are sent from a client, SOAPtest will check for consistencies and send the messages to the appropriate service. If a message is inconsistent with the established pattern, a SOAP fault is returned to the client, thereby ensuring security across the service; www.parasoft.com.

ClearMail 2.0 Now Available

Corrigan Consulting has released ClearMail 2.0, an update of the company's tool for blocking spam within a Lotus Notes environment. Almost all Lotus Notes distributions are supported, enabling most users to exploit ClearMail's whitelist approach to spam. ClearMail also provides administrator-level controls to help minimize exposure to e-mail-borne viruses and control the size of mail files; www.corriganinc.com.

10x Software's Open Source IDE

JewelBox Enterprise Edition 1.0 from 10x Software is an integrated development environment built entirely of open-source software. Like its Web-application counterpart, the product consists of two components, JewelBox Builder and JewelBox Runner. The Enterprise Edition is also centered on the popular Eclipse IDE, extended to include plug-ins for industry standards such as Tomcat, JBoss, MySQL, Ant, CVS, JUnit, Struts, and a complement of application frameworks and tools; www.10xsoftware.com.

Please send new product announcements to products@computer.org.

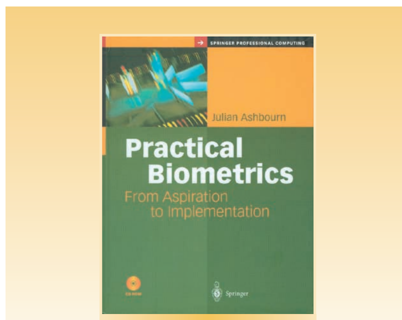
Implementing Verification Biometrics

Practical Biometrics: From Aspiration to Implementation, Julian Ashbourn. This book concentrates on the implementation of biometric verification techniques, with specific regard to wide-scale public applications.

Written from an operational rather than purely academic perspective, it serves as a practical guide, identifying scalability issues in applications that feature biometric verification techniques. Other topics include interoperability, both from a technical and operational perspective; ethnicity and the associated implications for biometric verification checks; and user psychology.

The book also explores nondevice-specific issues such as human factors, the environment, privacy, and data protection.

Springer; www.springer-ny.com; 1-85233-774-5; 180 pp.; $49.95.



FROM ADVENTURE TO ZORK

Twisty Little Passages, Nick Montfort. Interactive fiction—the best-known form of which is the text game or text adventure—has received little critical attention compared to other forms of electronic literature such as hypertext fiction and chatterbots. The book title refers to a maze in the first interactive work of fiction, *Adventure*, and the author examines interactive fiction from both the gaming and literary perspectives.

This study of interactive fiction begins with its most important literary ancestor, the riddle. The author then discusses *Adventure* and its precursors and examines the mainframe text games developed in response, focusing on the most influential work of that era, *Zork*.

The book covers the introduction of commercial interactive fiction for home computers, particularly the contributions produced by Infocom. The author also considers the influence of interactive fiction on other literary and gaming forms.

MIT Press; mitpress.mit.edu; 0-262-13436-5; 286 pp.; $29.95.

ARCHITECTING ENTERPRISES

A Practical Guide to Enterprise Architecture, James McGovern, Scott W. Ambler, Michael E. Stevens, James Linn, Vikas Sharan, and Elias K. Jo. This book helps readers create adaptive strategies for successfully implementing enterprise architectures. Its candid assessments of existing practice draw on detailed examples from the authors' experiences building software infrastructures for leading financial services, telecommunications, media, and e-business firms.

The authors provide start-to-finish guidance for architecting effective system, software, and service-oriented architectures; using product lines to streamline enterprise software design; architecting presentation tiers and user experience; and driving the technical direction of the entire enterprise.

Prentice Hall PTR; www.phptr.com; 0-13-141275-2; 496 pp.; $39.99.

COMPILER WRITER'S HANDBOOK

Engineering a Compiler, Keith D. Cooper and Linda Torczon. The proliferation of processors, environments, and constraints on systems has cast compiler technology into a wider variety of settings, changing the compiler writer's role. Today, code might be judged on how small it is, how much power it consumes, how well it compresses, or how many page faults it generates.

In this evolving environment, the task of building a successful compiler relies on the compiler writer's ability to balance and blend algorithms, engineering insights, and careful planning. Today's writer must choose a path through a design space filled with diverse alternatives, each with distinct costs, advantages, and complexities.

This book explores ways these problems have been solved and the constraints that made each solution attractive. By helping readers understand the problems' parameters and their impact on compiler design, the authors hope to convey both the depth of the problems and the breadth of possible solutions. They seek to cover a broad enough selection of material to show readers that real tradeoffs exist and that the impact of those choices can be both subtle and far-reaching.

Morgan Kaufmann; www.mkp.com; 1-55860-698-X; 801 pp.; $64.95.

Editor: Michael J. Lutz, Rochester Institute of Technology, Rochester, NY; mikelutz@mail.rit.edu. Send press releases and new books to *Computer*, 10662 Los Vaqueros Circle, Los Alamitos, CA 90720; fax +1 714 821 4010; newbooks@computer.org.

Databases Deepen the Web

Thanaa M. Ghanem and Walid G. Aref, Purdue University

The Web has become the preferred medium for many database applications, such as e-commerce and digital libraries. These applications store information in huge databases that users access, query, and update through the Web. Database-driven Web sites have their own interfaces and access forms for creating HTML pages on the fly. Web database technologies define the way that these forms can connect to and retrieve data from database servers.

The number of database-driven Web sites is increasing exponentially, and each site is creating pages dynamically—pages that are hard for traditional search engines to reach. Such search engines crawl and index *static* HTML pages; they do not send queries to Web databases.

The information hidden inside Web databases is called the "deep Web" in contrast to the "surface Web" that traditional search engines access easily.

DATABASE CONNECTIVITY

Querying via direct Structured Query Language is one of the most common ways to access a database. As the number of database servers and query interfaces increased, application developers needed a standard method to access different databases. To this end, Microsoft developed the Open Database Connectivity standard. The ODBC interface defines

Online databases continually generate Web content that users can only access through direct database queries.

- a common way to connect and log on to a database management system;
- a standardized representation for data types; and
- libraries of ODBC API function calls that let an application connect to a DBMS, execute SQL statements, and retrieve results.

A program can use ODBC to read data from a database without targeting a specific DBMS. All the program needs is the vendor-supplied ODBC driver to link to the required database.

ODBC levels of conformance

To enable applications and drivers to implement portions of the ODBC API specific to their needs, the standard defines conformance levels for drivers in both the API and the SQL grammar.

The ODBC API defines three conformance levels: A core set of functions corresponds to the functions in the X/Open and SQL Access Group Call Level Interface specification; level 1 and level 2 functions extend the core set. The ODBC SQL grammar also has three conformance levels: minimum, core, and extended. Each higher level provides more fully implemented data definition and data manipulation language support.

After ODBC's success, Microsoft introduced OLE DB, an open specification designed for accessing all kinds of data—not only data stored in DBMSs.

Java Database Connectivity

The Java platform subsequently called for a new connectivity standard, and the result was Java Database Connectivity. The JDBC interface provides the same functionalities as ODBC. JDBC-ODBC bridges enable developers in non-Java environments to use JDBC drivers to connect to databases.

DATABASE-TO-WEB CONNECTIVITY

A Web database environment consists mainly of a Web browser (the client), a Web server that understands HTTP, and a DBMS that understands SQL.

Database-to-Web connectivity requires a layer between HTTP and SQL that can translate between them. Several connectivity technologies have emerged that differ primarily in which part of the architecture sends queries to the database. Figure 1 gives a simple view of system architectures based on either two-tier or three-tier technologies.

Two-tier technologies

A two-tier architecture accomplishes the database-to-Web integration in a client tier, consisting of a Web browser and Web server, and a server tier, consisting of the DBMS. Three technologies prevail in this architecture.

Common gateway interface. CGI is a program that runs on the server tier and handles all the transformations between HTTP and SQL. The Web database access form includes a link to the CGI program. When a client references the HTML form, the Web server extracts the query parameters and forwards them to the CGI program on the server.

The CGI program reads the parameters, formats them appropriately, and sends a query to the database. When it receives the query result, the CGI program formats it as HTML pages and sends it back to the Web server.

Perl and JavaScript are two among many popular languages for writing CGI programs.

Java applets. Applets are Java programs that Web browsers can load dynamically and execute on the client's Web browser. Applets use JDBC to connect directly to the DBMS via sockets and thus do not require a Web server.

Server side includes. SSI is code written inside an HTML page and processed by the Web server. When a client invokes the HTML page, the Web server executes the scripts, which in turn use ODBC or JDBC to read data from the DBMS. The Web server then formats the data into HTML pages and sends it back to the browser.

Examples of SSI scripting languages are ASP, ASP.NET, JSP, PHP and ColdFusion.

Three-tier technologies

Three-tier architectures add a middleware tier—the application server—between the client and server tiers. The middleware handles all application operations and connections for the clients, including data transfer between the Web server and DBMS.

Application servers offer other functions such as transaction management and load balancing. Some implementations merge the application and Web server into one Web application server.

Examples of application servers include IBM Websphere, Oracle 9i, and Sun ONE.

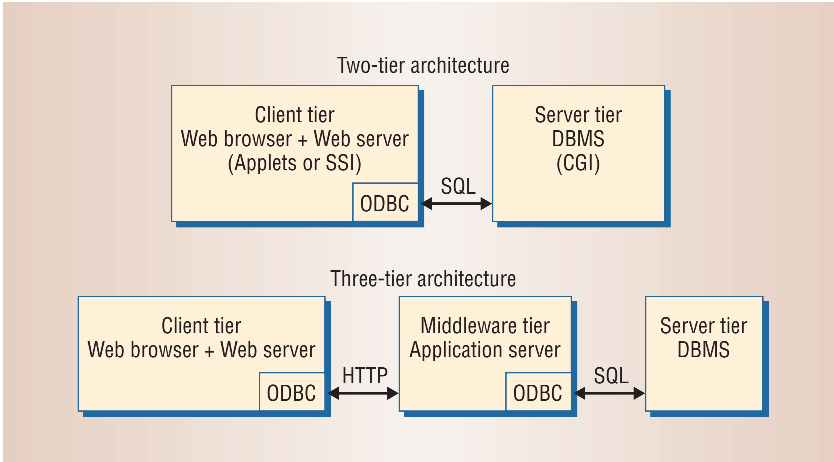


Figure 1. Database-to-Web connectivity: (a) two-tier technologies and (b) three-tier technologies.

DEEP WEB SEARCH ENGINES

A July 2000 survey by BrightPlanet (www.brightplanet.com) showed that the deep Web includes about 550 billion pages and that public information hidden in it is 400 to 500 times larger than what users can access through the surface Web. Access to information in the deep Web comes only through database interfaces and queries.

Comprehensive information retrieval requires simultaneous searching of multiple surface and deep Web resources. Deep Web search engines aim to identify, retrieve, and classify such content. They work by rephrasing the query and sending it simultaneously to multiple databases in real time.

Current commercial products in this area include BrightPlanet's Deep Query Manager (DQM2), Quigo Technologies' Intellisonar (www.quigo.com/intellisonar.htm), and Deep Web Technologies' Distributed Explorit (www.deepwebtech.com/dexpl.shtml).

In the research community, EduMed is a project at Purdue University (www.cs.purdue.edu/edumed/). EduMed helps users find and query online medical databases uniformly. It is built on top of an extensible prototype multimedia database management system called VDBMS (www.cs.purdue.edu/vdbms/).

We expect deep Web search engines and technologies to improve rapidly and to dramatically affect how the Web is used by providing easy access to many more information resources. ■

Thanaa M. Ghanem is a graduate research assistant in the Department of Computer Sciences at Purdue University. Contact her at ghanemtm@cs.purdue.edu.

Walid G. Aref is an associate professor in the Department of Computer Sciences at Purdue University. Contact him at aref@cs.purdue.edu.

Editor: Sumi Helal, Computer and Information Science and Engineering Dept., University of Florida, P.O. Box 116125, Gainesville, FL 32611-6120; helal@cise.ufl.edu

Programming Models for Hybrid CPU/FPGA Chips

David Andrews and Douglas Niehaus, University of Kansas
Peter Ashenden, Ashenden Designs

Components that combine a CPU and reconfigurable logic gates need a programming model that abstracts the computational hardware.

Designers of embedded and real-time systems are continually challenged to meet tighter system requirements at better price-performance ratios. Best-practice methods have long promoted the use of commercial-off-the-shelf components to reduce design costs and time to market, but creating COTS components that are reusable in a wide range of applications remains difficult.

In part, the challenge lies in satisfying the contradictory design forces of generalization and specialization. Systems designers are all too familiar with the tension these opposing forces cause in trying to balance cost versus performance. Adopting COTS components reduces costs and time to market but often fails to meet the most demanding performance requirements; custom-designed components can achieve significantly higher performance but at greater development costs and longer times to market.

HYBRID CPU/FPGA CHIPS

Emerging hybrid chips containing both CPU and field-programmable gate array (FPGA) components are an exciting new development. They promise COTS economies of scale while also supporting significant hardware customization. For example, the Virtex II Pro from Xilinx (www.xilinx.com) combines up to four Power PC 405 cores with up to approximately 4 million free gates, while the Excalibur from Altera (www.altera. com) combines an ARM 922 core with about the same number of free gates.

Designers can configure the free FPGA gates with a widening range of standard FPGA system library components. This intellectual property includes serial and parallel I/O interfaces, bus arbiters, priority interrupt controllers, and dynamic RAM controllers. The IP components let designers select an FPGA IP set to create a specialized system-on-chip. SoC solutions can achieve COTS economies of scale with IP selected to meet specific system requirements.

The free FPGA gates can also support customized application-specific components for performance-critical functions. While FPGA-based implementations do not perform as well as equivalent ASICs, they can often provide acceptable performance with a significantly better price-performance ratio.

FPGA COMPONENT SPECIFICATION

Tapping the full potential of these hybrid chips presents an interesting challenge. System developers can replace current COTS board designs with a single hybrid chip, but specifying custom components within the FPGA requires knowledge of hardware design methods and tools that most system programmers do not have.

Researchers are seeking solutions to this problem by investigating new design languages, hardware/software specification environments, and development tools. Projects such as Ptolemy (ptolemy.eecs.berkeley.edu/), Rosetta (sldl.org), and System-C (systemc.org) are exploring system-level specification capabilities that can drive software compilation and hardware synthesis.

Other projects such as Streams-C (rcc.lanl.gov/Tools/Streams-C/index.php) and Handel C (www.celoxica.com) focus on raising the abstraction level for programming FPGAs from gate-level parallelism to modified and augmented C syntax. System Verilog (eda.org/sv-cc/) and a newly evolving VHDL standard (eda.org/vhdl-200x/) attempt to abstract away the distinction between the two sides of the traditional low-level hardware/software interface into a system-level perspective.

Although these approaches differ in the scope of their objectives, they all share the goal of raising the abstraction level required to design and integrate hardware and software components.

Crossing the boundary

A question remains as to whether high-level FPGA programming languages will mature to a point that lets software engineers apply their skills across the CPU/FPGA boundaries.

Unfortunately, current hybrid programming models are still immature. They generally treat FPGAs as independent accelerators with computations outside the scope of the programs running on the CPU. They generally use simple I/O queues for communications between the FPGA-based and CPU-based computations, and this hybrid model is not mature enough for synchronizing the execution of component computations—a critical capability in distributed and parallel computation.

A mature high-level programming model would abstract the CPU/FPGA components, bus structure, memory, and low-level peripheral protocols into a transparent system platform. In general, programming models provide the definition of software components as well as the interactions between these components, as Edward Lee described in a discussion of embedded systems' software frameworks ("What's Ahead for Embedded Software?," *Computer*, Sept. 2000, pp. 18-20).

Message passing and shared-memory synchronization protocols are two familiar component interaction mechanisms. Practitioners have successfully used both in embedded systems and enjoy debating the relative merits of their personal choice. We describe the multithreaded shared-memory model here, though many aspects of the discussion are equally appropriate for the message-passing model.

MULTITHREADING MODEL

The multithreaded programming model is convenient for describing embedded applications composed of concurrently executing components that synchronize and exchange data. Its popularity is apparent in the widespread use of the Posix threads standard.

High-level abstraction

The multithreaded programming model specifies applications as sets of threads distributed flexibly across the system CPU and FPGA assets. At the highest abstraction level, the computational structure of hybrid applications remains familiar. Whether the threads implementing a computation are CPU- or FPGA-based becomes just another design and implementation parameter with resource use and application performance implications.

The programming model suports iterative application development.

How to perform this partitioning to best support application or system requirements is yet another challenging problem. However, the model supports iterative application development that begins with an exclusively CPU-based multithreaded implementation and gradually transfers specific threads to FPGA support.

All of these attributes speed the time to market. They also let designers focus FPGA support on those portions of the application that performance measurements indicate will benefit most from it.

Policy and mechanisms

We can draw a useful distinction between policy and mechanism. The multithreaded model *policy* is fairly simple: to allow the specification of concurrent execution threads and protocols for accessing common data and synchronizing the execution of independent threads. The *mechanisms* that a general-purpose processor uses to achieve this policy include the definition of data structures that store thread execution state information and the semantics of thread synchronization interactions with the operating system thread scheduler.

Both the synchronization and the thread-scheduling portions of the system software access data structures for semaphore control and thread context. In addition, most microprocessors provide the minimum hardware support required to implement atomic semaphore operations—for example, test-and-set or compare-and-swap support.

The FPGA computational model is expressed at an abstraction level different enough from that of the CPU to leave no immediately obvious equivalent to the CPU thread context of register set, program counter, and stack pointer. Additionally, current FPGA technology synthesizes the data paths and operations that represent the thread computations and maps them into the FPGA before runtime.

These differences require new mechanisms for achieving the basic multithreaded model policy relative to threads running within the FPGA and threads interacting across the CPU/FPGA boundary. Although the lack of an existing computational model seems to be a liability at first glance, it actually presents an opportunity to create efficient mechanisms for implementing FPGA threads and for supporting thread synchronization.

OPERATING SYSTEM CODESIGN

A key challenge in giving programmers access to these new hybrid computations is extending the operating system across the CPU/FPGA boundary in a form that abstracts the differences between the computational models used within the CPU and FPGA components.

Operating systems provide the underlying synchronization and control mechanisms for higher-level programming models as well as a generic set of interfaces through which application programs can access system functions. These OS functions relieve application programmers from needing to know the low-level protocols and device-specific requirements. For hybrid systems, this functionality will require hardware/software codesign across the CPU/FGPA boundary.

System developers have always regarded OS codesign as a means to increase system performance through parallelism and to improve the predictability of system behavior. Thus, the codesign work required to enable a hybrid programming model can also enhance general OS performance.

Function migration to FPGA domain

A wide range of OS functions will certainly benefit from either a partial or a complete migration into the FPGA domain. Examples include receiving and evaluating interrupts, time keeping, event queue management, task scheduling, clock synchronization, support for distributed computation, and concurrency control. FPGA support can increase accuracy, performance, and predictability.

System benefits

In this context, FPGAs provide a means to refine system properties such as the resolution and precision of the system time standard. This translates to finer event scheduling, a fundamental aspect of any real-time system.

FPGA codesign can also reduce scheduling jitter, or variable delays of the scheduling decision. Jitter can originate from several sources, including variable execution time of system software, existing mechanisms used to implement interrupt-handling methods, task scheduling, and concurrency control. FPGA-based implementation of some of these OS functions will significantly reduce system overhead by transferring computational loads from the CPU into FPGA-based concurrent state machine components.

These and many other possibilities make codesign of OS functions an exciting area of current research.

System-level multithreaded programming requires new hardware/software codesign approaches that support operating system and application functions. Current efforts to develop this capability will make hybrid CPU/FPGA computational components accessible to a much broader community of system programmers, increasing OS performance and reducing design times and development costs. ■

Acknowledgment

The work is partially sponsored by NSF EHS contract # CCR-0311599.

David Andrews *is an associate professor of electrical engineering and computer science at the University of Kansas. Contact him at dandrews@ittc.ku.edu.*

Douglas Niehaus *is an associate professor of electrical engineering and computer science at the University of Kansas. Contact him at niehaus@ittc.ku.edu.*

Peter Ashenden *is director of Ashenden Designs. Contact him at peter@ashenden.com.au.*

Editor: Wayne Wolf, Dept. of Electrical Engineering, Princeton University, Princeton, NJ, 08544-5263; wolf@princeton.edu

Distributed Learning Environments

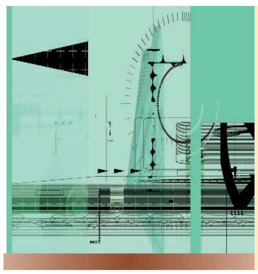
Maryam Alavi, Emory University

Distributed learning is an instructional model that gives students access to a wide range of resources—teachers, peers, and content such as readings and exercises—independently of place and time. It leverages computing, communication, and multimedia technologies to create learning environments that can be richer and more flexible, scalable, and cost-effective than the standard classroom or lecture hall.

Over the past several years, the popularity of distributed learning environments (DLEs) in both professional and academic settings has steadily increased due to

- the rising demand from traditional students as well as working adults for postsecondary and professional education,
- advances in information technologies,
- the Internet's ubiquity,
- the emergence of high-capacity wire and wireless networks, and
- the pervasiveness of networked personal computers in both businesses and homes.

Many universities and colleges, for-profit organizations, and corporate training departments now offer DLE courses and programs. For example, Emory University's Goizueta Business School and Duke University's Fuqua School offer executive MBA programs in a distributed learning format, while the for-profit University of Phoenix has online degree programs in various fields.



DLEs must balance technology with student attributes and instructional strategies.

DLE APPROACHES

Information technologies give DLE designers the flexibility to customize learning environments for diverse student communities. For example, content navigation can be predetermined by the instructor or left to student control. A simple DLE might consist of sequential access to static material, while a more complex offering can involve high-level interactions among students as well as between students and instructors.

Distributed learning may be generally categorized as broadcast, online, or collaborative. However, specific DLEs usually include features of more than one of these approaches. For example, a designer might combine broadcast for the efficient delivery of lectures with collaborative learning for student team projects.

Broadcast

The broadcast approach replicates a lecture-style pedagogy. In a typical configuration, lectures and presentation material are transmitted from the teacher's site via cable or satellite to the students' locations. In some cases, the one-way broadcast of learning content is supplemented with direct synchronous and asynchronous links between each student and the instructor to facilitate communication of feedback and questions.

Online

Online distributed learning uses information technologies to provide students with Web-based access to content and resources such as assignments, software simulations, and tests and exercises. This approach lets students choose the time, pace, and frequency of their learning activities.

Collaborative

Learning occurs most effectively when students can interact in small task-oriented groups. Information technologies now make it possible to extend collaborative interactions to distributed environments. These interactions can be synchronous—for example, through videoconferencing and online chat—or asynchronous—for example, through e-mail and computerized bulletin boards.

DISTRIBUTED LEARNING MANAGEMENT SYSTEMS

The Internet has played a key role in the development of DLEs by providing scalable connectivity to bridge geographic distance and establishing the browser as a ubiquitous user interface for various distributed learning software applications.

Learning management systems build upon these and other technical features to provide instructors and students with high-level, easy-to-use technical tools for support of learning activities, including

Distributed Learning Resources

The following resources provide more information about distributed learning environments:

- M. Alavi and D. Leidner, "Technology-Mediated Learning: A Call for Greater Depth and Breadth of Research," *Information Systems Research,* vol. 12, no. 1, 2001, pp. 1-10.
- M. Alavi, Y. Yoo, and D.R. Vogel, "Using Information Technology to Add Value to Management Education," *Academy of Management J.*, vol. 40, no. 5, 1997, pp. 1310-1333.
- S.R. Hiltz and R. Goldman, eds., *Asynchronous Learning Networks: The Research Frontier,* Lawrence Erlbaum Associates, to appear in 2004.
- G. Piccoli, R. Ahmed, and B. Ives, "Web-Based Virtual Learning Environments: A Research Framework and a Preliminary Assessment of Effectiveness in Basic IT Skills Training," *MIS Quarterly,* vol. 25, no. 4, 2001, pp. 401-426.
- G.C. Van Dusen, *The Virtual Campus: Technology and Reform in Higher Education,* Jossey-Bass, 2000.

- multimedia information and document management for storing, retrieving, and organizing content and discussions;
- advanced e-mail with spell check, multiple views for sorting messages, and extensive editing capabilities;
- threaded discussions for asynchronous interactions; and
- instant messaging to support synchronous exchanges.

Most DLEs feature a simple interface that lets students easily navigate the various components and reduces the need to learn new interface styles and commands. They also usually include software support for student registration, testing, grading, and other administrative activities.

Some systems—including Blackboard (http://blackboard.net), the IBM Lotus Learning Management System (www.lotus.com/lotus/offering6.nsf/wdocs/homepage), and WebCT (www.webct.com)—are developed commercially, and can be licensed or "rented" from vendors. Other systems, such as the Virtual Classroom developed at the New Jersey Institute of Technology, are designed for in-house use.

Current tools embedded in most learning management systems focus on electronically warehousing course content and disseminating static lectures and material. Beyond e-mail, live chat, and electronic bulletin boards, technology that supports case-based and collaborative pedagogies tends to be scarce. Future learning management systems must therefore provide support for a wider range of instructional strategies, such as software templates for structured group interactions.

DLE EFFECTIVENESS

Numerous research studies have revealed no significant differences between distributed and traditional learning environments in terms of their learning outcomes. A DLE's advantages include convenience, flexibility, currency of material, possibilities for lowering costs, and individualized learning. However, other studies indicate higher student dropout and course withdrawal rates in DLEs and, in some cases, less satisfaction with the learning process.

DLE effectiveness depends as much on student attributes and instructional strategies as it does on inherent features of the distributed learning model. For example, self-regulation and self-discipline—staying on schedule, logging on frequently, and so on—combined with mindful engagement in learning activities generally result in better outcomes. Other positively correlated attributes include levels of academic ability, motivation, degree of effort, and maturity.

In addition to student characteristics, a major factor in DLE success is the instructional strategy with respect to

- *presentation*—the selection, display mode, and format of information;
- *sequencing*—the order in which different topics are presented to learners;
- *delivery*—the media used as the presentation channels; and
- *synthesis*—the establishment of relationships among various topics.

Technology can greatly enhance DLE instructional strategies. For example, using digitized multimedia such as photographs, video clips, animated graphics, and sound can help retain students' attention, while interactive activities such as live chat can stimulate discussion and collaboration.

Continuing IT improvements will make distributed learning an increasingly viable alternative to traditional face-to-face approaches. At the same time, technology per se does not produce desired learning outcomes. Failure to consider students' attributes and instructional strategies can lead to technologically advanced but ineffective DLEs that waste resources and dissatisfy learners. ■

Maryam Alavi is the John M. and Lucy Cook Chair of Information Strategy at Emory University's Goizueta Business School. Contact her at maryam_alavi@bus.emory.edu.

Editor: Richard G. Mathieu, Dept. of Decision Sciences and MIS, St. Louis University, St. Louis, MO 63108; mathieur@slu.edu

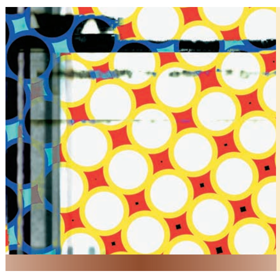
Best Business Practices for Standards Groups

Michael LoBue, LoBue & Majdalany Management Group

In highly competitive industries such as computing and communications, standards serve many purposes: They ensure equal access to primary technology, increase competition, help grow markets, and spark innovation. To guarantee fairness and accessibility, the originators of new standards often form trade associations to manage the technology and oversee the work of developing future technologies.

However, this begs the question of who is setting the business practice standards for these standards groups. No member of a trade association wants to wake up one morning and learn that its critical standards group is facing bankruptcy or a major lawsuit from a conflict of interest, or some other business risk that is easily avoidable with knowledgeable and competent professional management.

The Standard of Good Practices for Association Management Companies (ANSI/IAAMC A100.1-2002) is the basis of a new accreditation program sponsored by the International Association of Association Management Companies (IAAMC; www.iaamc.org). This program offers trade associations and professional societies previously unavailable management choices. Approved by the American National Standards Institute, the criteria are based on specific expertise and documented best practices defined through the ANSI consensus process.



A new standard raises the bar for association management companies.

ASSOCIATION MANAGEMENT COMPANIES

Three basic options are available for association management and administration: member volunteers, employed staff, and an association management company. Table 1 provides a brief overview of the relative strengths and weaknesses of each option. Of these, only AMCs can deliver an accredited association management solution.

AMCs are for-profit firms that provide management and operational support services for nonprofit organizations, freeing members to focus on governance, strategy, and direction. In a survey conducted in 2003, the AMC Institute (www.amcinstitute.org) identified 527 association management companies from 47 US states and five other countries. These AMCs manage more than 3,000 client organizations, serving more than 3.1 million members.

The AMC business segment is established and mature. Operating budgets of AMC-managed organizations range from $50,000 to more than $16 million annually; the average annual budget is $677,000. More than half of these firms have been in business since before 1990, many for decades.

Associations managed by AMCs are free from employer obligations and the many business risks generally associated with their organizations' operations. Association management companies bear many of these operational obligations as a result of the very nature of their service agreements.

AMC STANDARD

Several years ago, the IAAMC membership decided that it should establish a standard for AMCs to use as a benchmark for identifying and providing quality services. After examining various methods, it determined that creating an American National Standard was the best approach because

- ANSI procedures ensure that the most impartial processes are used to create the standard and that no small group of interested parties dominates the standard; and
- given ANSI's worldwide recognition, there is a higher likelihood that the standard will be accepted as an independent measure of high quality.

The new AMC standard establishes detailed performance requirements representing the best practices for association management in 10 key areas: client contracts, servicing the client, project completion, financial management and internal controls, insurance, employee recruitment and selection,

Table1. Association management options.

Option	Strengths	Weaknesses
Member volunteers	• High commitment to and understanding of the organization's mission • Low direct cost (initially)	• Time conflicts with volunteers' employers • No expertise at running an association • Turnover creates inconsistencies • No neutral party; open to conflicts of interest
Employed staff	• Dedicated staff • Specialized knowledge about the association • Continuity	• High overhead (office space, equipment) • Not able to staff key positions or areas of expertise unless they can justify a full-time employee • Higher burden and risk associated with being an employer
Association management company	• Expertise in association management • Lower overhead • Expertise in multiple disciplines • Absence of many typical legal risks associated with operations • Continuity	• Possible mismatch between the association's needs and the AMC's expertise

employee training and professional development, subcontracting and purchasing, record keeping, and internal and external audit requirements.

Why is such a standard necessary? After all, how difficult can it be for any professional service firm, which manages its own business, to manage a nonprofit trade association? The answer is "not very," but neither is designing a next-generation circuit board or operating system, if you know what you are doing.

The same applies to managing industry associations; knowledge, training, and experience all make a difference. Association management also carries an implied fiduciary role—as a fiduciary, you are held to a higher standard of performance and care when acting on behalf of another's affairs than on your own.

IAAMC ACCREDITATION

Accreditation, the process that assures compliance with a standard, is common in the marketplace. For example, motor oil manufacturers demonstrate their adherence to the practices endorsed by the Society of Automobile Engineers by placing the acronym SAE on their products. Likewise, common household appliances bearing the UL mark have been accredited by the independent Underwriters Laboratories.

In the case of the Standard of Good Practices for Association Management Companies, AMCs wishing to demonstrate a commitment to excellence apply to the IAAMC for accreditation. The AMC shoulders the cost of an external auditor, usually a certified public accountant trained in the practice of accounting-firm peer reviews, whose qualifications the IAAMC has accepted. The auditor conducts an onsite investigation of the AMC's policies and practices, interviews staff, and inspects records to verify that the firm meets or exceeds the standard.

Associations shopping for management services are already beginning to specify that only accredited firms are eligible to submit proposals for new business. IAAMC accreditation is thus a powerful tool for associations to ensure that their management services are the most qualified the market has to offer. Currently, 21 AMCs have earned IAAMC accreditation to the new standard and another 11 have applications on file.

BENEFITS OF THE STANDARD

Those who represent their companies' interests in trade associations and standards bodies face two major challenges: their time is extremely scarce, and they are industry and technology experts, not association experts.

Beyond the usual economies of scale associated with management by an AMC, associations managed by accredited AMCs enjoy three important benefits.

First, the leaders and membership of organizations managed by an accredited AMC do not have to spend time learning about best practices or defining them for their needs. Accredited AMCs bring this best-practices framework to their client organizations.

Second, representatives of organizations managed to the standard have more time to focus on advancing their goals rather than questioning how well their organization is managed.

Third, standards organizations that an accredited AMC manages can be assured that their management deeply understands the standards process, the value of standards, and what is involved in achieving and maintaining them. Accredited AMCs face many of the same types of compliance requirements that their client organizations do in standardizing the technical specifications that their members generate.

Technology markets have been well served by standards. Now leaders of critical standards associations can use a rigorous standard based on the industry-accepted ANSI consensus process to choose how their own organization's business needs are met. By adhering to a set of best practices identified by a broad spectrum of service providers and associations, accredited AMCs offer associations something they have never had before —world-class management. ■

Michael LoBue, CAE, is president of LoBue & Majdalany Management Group. Contact him at LoBue@LM-Mgmt.com.

Editor: Gary Robinson, 85 East India Row, Apt. 18C, Boston, MA 02110; robinson_gary@emc.com

From Organization Man to Free Agent

Mohan Babu, Infosys Technologies Limited

Wojciech Cellary brought out a key point in his column on "The Profession's Role in the Global Information Society" (Sept. 2003, pp. 124, 122-123): Computing professionals continually face exclusion from their work because digital technology advances so swiftly. Along with this risk, changes in the global information society have led to a shift in the computing professional's role from "organization man" to free agent. Renowned management guru Peter Drucker outlined this trend in "Managing Oneself" (*Harvard Business Journal*, March-April 1999, pp. 65-74). Drucker advises professionals, "… and we will have to stay mentally alert and engaged during a 50-year working life, which means knowing how and when to change the work we do."

THE ORGANIZATION MAN

The traditional concept of the computing profession originated just after World War II, when most Western nations enjoyed a long growth spell. To cater to the emerging needs of the postwar market, corporations built gigantic factories to manufacture products and serve consumer needs. To manage these operations, organizations also started automating their systems with computers. Spurred by this growth in manufacturing productivity, governments, financial institutions, and retailers began to automate their systems as well. During this period, William H. Whyte wrote his much acclaimed book, *The Organization Man* (University of Pennsylvania Press, 2002), and the term soon caught the fancy of an entire generation of working professionals. Whyte defines organization men as

> … the ones of our middle class who have left home, spiritually as well as physically, to take the vows of organization life, and it is they who are the mind and soul of our great self-perpetuating institutions. In a system that makes such hazy terminology as "junior executive" psychologically necessary, they are of the staff as much as the line, and most are destined to live poised in a middle area that still awaits a satisfactory euphemism. But they are the dominant members of our society nonetheless.…

Note that Whyte wrote his book during an age when men constituted the bulk of the white-collar workforce, and I will not attempt to be politically correct by using the term *organization people* here.

To thrive in a rapidly changing world, computing professionals must become free agents.

For nearly half a century after the book appeared, the organization man typified the professional. In most parts of the world, huge corporations—private, public, and government-owned—employed hundreds of thousands of organization men. That era also saw the rise of the computing professional, personified by legions of IBM employees clad in white shirt and tie. Endless movies idolized devoted company men in gray flannel suits and the stable life they enjoyed. Most white-collar professionals across the world sought and could aspire to this American Dream of a good education that led to a good job, a house in the suburbs, and a wife and kids. During this age the public regarded corporations with reverence and deference, a topic analyzed by authors like Fred Harmon and Garry Jacobs, who note in their book *The Vital Difference* (Amacom, 1985) that "Ma Bell [AT&T] became the ultimate symbol of a benevolent corporation working in and for the public interest."

DEATH OF THE ORGANIZATION MAN

The corporate world experienced a radical transformation in the late 1980s and early 1990s, when cost cutting, downsizing, and rightsizing became the new mantra. Corporations across the globe transformed from benevolent entities to profit centers driven by the interests of their stockholders.

This left the concept of the organization man dying if not completely dead. The most visible cornerstone of the organization man's existence—lifetime employment—eroded as well. Globalization of business and manage-

Continued on page 126

Continued from page 128

Table 1. Organization man versus free agent.	
Organization man	**Free agent**
Works with the HR department to plan a career path	Manages own career path
Career progression assumed to be well defined; top performers promoted	Career is typically a mix of entrepreneurship and projects, with no clear path to the top
Thinks of a career as a hierarchical path in the organization	Will go wherever opportunity and money are
Perceived to be loyal and enjoys job security	Considers the notion of a job-for-life passé; constantly competes for jobs, assignments, or projects
Identified with single employers; for example, business school graduates in the 1960s and 1970s vied for careers at IBM	Now only loosely associated with employers; even those working for blue-chip computing companies often explore newer opportunities

ment practices also meant the globalization of cost cutting, downsizing, and layoffs. Significantly, even Japanese companies—leading proponents of lifetime employment until recently—have revisited their ideals in light of their economy's decade-long downturn. In India, the wave of privatization sweeping through public companies has led to many so-called voluntary retirements. Asian and European companies have followed in their American counterparts' footsteps and now use layoffs as a regular cost-cutting measure.

Individuals have thus begun to realize that even if they wanted to, they could not entrust their career to a single company: Corporations themselves are regularly evolving, transforming, acquiring other businesses or being acquired by them, and sometimes going bankrupt. Professionals are experiencing what Intel cofounder Andy Grove calls a *strategic inflexion point (Only The Paranoid Survive,* Time Warner, 1999), with the traditional notions of work and career giving way to a new model in which individuals are expected to take responsibility for their own career moves.

TRANSITION TO FREE AGENT

Faced with this new reality, computing professionals have made a fundamental shift in how they view their careers. Traditional organizational hierarchies are giving way to project- and performance-oriented groups and structures, ushering in the era of *gold-collar workers*. These free agents are highly skilled professionals who owe a greater allegiance to their profession than to the organizations for which they work.

Daniel Pink first extended the term free agent—borrowed from professional sports—to corporate professionals in his book *Free Agent Nation* (Warner Books, 2002). Pink sees the emergence of moonlighting as one way professionals can hedge their bets in a changing world:

> Diversification—that is, an independent worker spreading her risks across a portfolio of projects, clients, skills and customers is the best hedging strategy…. In the Organization Man era, moonlighting was a big no-no, the very name implied that you were doing something illicit concealing your behavior under the cover of darkness. No more. Today, anybody who holds a job and isn't looking for a side gig—or crafting a business plan, writing a screenplay, or setting up shop on eBay—is out of touch. Moonlighting is a way to diversify your human capital investments—and hedge against the risk of your company collapsing or your job disappearing. In some sense, we're all moonlighters, because in every sense, we're all risk managers.

BECOMING FREE AGENTS

Table 1 shows how careers in IT consulting have evolved from lifelong single-employer jobs to a free-agent model. Y2K, the Internet, and the dot-com boom brought a whole legion of professionals from varied backgrounds into the computing field. Some joined traditional companies' IT departments, but many decided to explore careers in consulting. The industry also saw the appearance of a whole array of consulting companies, ranging from small shops with a handful of consultants to large system integrators like IBM and EDS.

The industry afforded a gamut of vocational choices, from short-term projects spanning a few weeks to long-term maintenance projects lasting a few years. Along the way, computing professionals also realized that the industry was becoming increasingly market driven. Thus, getting certified in vendor technologies, being associated with professional bodies—including the IEEE and ACM—and building expertise in current skills gave them more leverage than being associated with a blue-chip employer.

Individual computing professionals have also shown their market savvy by selling themselves as experts in Cobol, ERP, Java, .NET, the Web, and other technologies the market demands, sometimes juggling multiple hats at once. The career trajectory of many computing professionals has begun to resemble that of free agents who take on a series of projects or assignments that help them market their skills to the highest bidder.

The computing professional may be taking a page from a trend already established by other professionals in vocations such as law, medicine, finance, and academia. Lawyers and financial analysts have long known that their real allegiance is to the profession rather than to individual organizations or companies where they work. Being a corporate attorney or a corporate financial analyst is perceived to be less glamorous and financially rewarding than working for a high-profile partnership or, better still, founding one's own firm. Academicians and professors have refined moonlighting into an art—consulting for large corporations, helping their clients understand and incorporate the latest academic and research ideas, and raking in huge fees—even while continuing their day job of teaching and spearheading university research.

By building and maintaining a brand and attracting a steady stream of clients, free-agent professionals can thrive by following the models established by those in the following fields:

- lawyers and legal professionals;
- chartered accountants and financial professionals;
- doctors and medical specialists;
- management consultants;
- architects, builders, masons, and craftsmen;
- artists, performers, singers, and musicians;
- freelance writers and columnists;
- athletes and sports stars; and
- academicians and professors who moonlight as consultants.

Computing professionals now realize that they need to take active charge of the direction in which their careers are headed. Whether they view a career as a series of assignments or as a mix of traditional jobs and moonlighting, all computing professionals must actively take control of their career. Today's companies value people based on what they bring to the project, assignment, or work task rather than how many years these professionals have spent at one job. Quoting Drucker again:

> The challenges of managing oneself may seem obvious, if not elementary. And the answers may seem self-evident to the point of appearing naive. But managing oneself requires new and unprecedented things from the individual, and especially from the knowledge worker. In effect, managing oneself demands that each knowledge worker think and behave like a chief executive officer.... Every existing society, even the most individualistic one, takes two things for granted, if only subconsciously: that organizations outlive workers, and that most people stay put.... But today the opposite is true. Knowledge workers outlive organizations, and they are mobile. The need to manage oneself is therefore creating a revolution in human affairs.

Academia pays close attention to these industry trends. Engineering schools and universities, especially in the West, have begun introducing technologists to entrepreneurship and business fundamentals. Further, the students graduating into the field of computing are beginning to realize that courses in entrepreneurship will play an essential role in helping them manage their lives and careers.

As professionals in a workforce with evolving expectations of the employer-employee relationship, most of us will need to acquire and apply entrepreneurial and business management skills to manage our careers. Our career trajectories will thus depend on constant marketing and networking rather than climbing the ladder of a predefined career track. ■

*Although **Mohan Babu** is a manager with Infosys Technologies Limited's SETLabs Group, the views expressed in this column are his personally. Contact him at mohan@garamchai.com.*

Editor: Neville Holmes, School of Computing, University of Tasmania, Locked Bag 1-359, Launceston 7250; neville.holmes@utas.edu.au