

**Brought to You by**



**Like the book? Buy it!**

# Editorial Pointers



THIS MONTH MARKS MY 17TH year with *Communications*. During that time this magazine has chronicled the evolution of practically every known—and heretofore unknown—discipline within computer science. Indeed, we watched the computing field itself evolve

into a global arena of unprecedented triumphs and erratic spirals. But with it all, it's still difficult to accept that events would one day warrant the coverage we feature here—U.S. homeland security.

Naïve? Perhaps. ACM is indeed a global village; we are well aware there are many readers of and authors in this magazine who live in parts of the world where homeland security—or lack thereof—has always been a tangible part of everyday life. For many more of us, however, terrorism on home soil is fresh ground. In its aftermath, governments worldwide have turned to their science and technology communities to build the tools and intelligence capabilities to better secure their citizens and defend their borders against attack.

The U.S. government has drawn upon a vast universe of technologists from all disciplines to develop new digital techniques for defending national borders and interests. Although homeland security incorporates myriad research branches, this month's special section introduces some of the information and communications technologies that have emerged as a

result, particularly as they pertain to intelligence gathering, protecting the global network infrastructure, and enhancing emergency response. Guest editor John Yen says such protective efforts require “tremendous science” and that major challenges remain, given the secretive nature of terrorist activities and daunting environmental constraints. We hope this section prompts the IT community—worldwide—to help develop the solutions together.

Also in this issue, Wang et al. contend the criminal mind is no match for a new record-linkage method designed to match different, deceptive criminal identity records. And Escudero-Pascual and Hosein argue that government attempts to create a one-size-fits-all technology policy to protect communications infrastructures do not translate on an international scale.

Dalal et al. describe a framework for enterprise process modeling that should enhance the design of the next generation of ERP systems. Kim and Schneiderjans offer a unique interpretation of distance learning, asserting success of Web-based education programs may depend on the personality characteristics of employees. And Sethi interprets a series of studies on stress among IS professionals and how it affects productivity and costly employee turnover.

*Diane Crawford*

Editor

## COMMUNICATIONS

OF THE ACM • A monthly publication of the ACM Publications Office

ACM, 1515 Broadway, New York, New York 10036-5701 USA (212) 869-7440 FAX: (212) 869-0481

**Editor:** Diane Crawford

**Managing Editor:** Thomas E. Lambert

**Senior Editor:** Andrew Rosenbloom

**Editorial Assistant:** Mailyn Chang

**Copyright:** Deborah Cotton

### Contributing Editors

Phillip G. Armour; Hal Berghel;

Michael Cusumano; Peter J. Denning;

Robert L. Glass; Seymour Goodman;

Meg McGinity; Rebecca Mercuri;

Peter G. Neumann; Pamela Samuelson

**Art Director:** Caren Rosenblatt

**Production Manager:** Lynn D'Addesio

### Advertising

ACM Advertising Department

1515 Broadway, New York, NY 10036-5701

(212) 869-7440; Fax: (212) 869-0481

**Account Executive:**

William R. Kooney email: [acm-advertising@acm.org](mailto:acm-advertising@acm.org)

For the latest media kit—including rates—contact:

Graciela Jacome: [jacome@acm.org](mailto:jacome@acm.org)

### Contact Points

**CACM editorial:** [crawford\\_d@acm.org](mailto:crawford_d@acm.org)

**Copyright permission:** [permissions@acm.org](mailto:permissions@acm.org)

**Calendar items:** [calendar@acm.org](mailto:calendar@acm.org)

**Change of address:** [acmcoa@acm.org](mailto:acmcoa@acm.org)

### Communications of the ACM

(ISSN 0001-0782) is published monthly by the ACM, 1515 Broadway, New York, NY

10036-5701. Periodicals postage paid at

New York, NY 10001, and other mailing

offices. POSTMASTER: Please send address

changes to Communications of the ACM, 1515

Broadway, New York, NY 10036-5701 USA

### Printed in the U.S.A.



# ACM *The Association for Computing Machinery*

ACM (founded 1947) is an international scientific and educational organization dedicated to advancing The art, science, engineering, and application of information technology, serving both professional and public interests by fostering the open interchange of information and by promoting the highest professional and ethical standards.

**Executive Director and CEO:** John White  
**Associate Director, Office of Director, ACM U.S. Public Policy Office:** Jeff Grove  
**Deputy Executive Director and COO:** Patricia Ryan  
**Director, Office of Information Systems:** Wayne Graves  
**Director, Office of Financial Services:** Russell Harris  
**Financial Operations Planning:** Darren Ramdin

**Director, Office of Membership:** Lillian Israel

**Director, Office of Publications:** Mark Mandelbaum  
**Deputy Director:** Bernard Rous  
**Deputy Director, Magazine Development:** Diane Crawford  
**Publisher, ACM Books and Journals:** Jono Hardjowirogo

**Director, Office of SIG Services:** Donna Baglio  
**Assistant Director, Office of SIG Services:** Erica Johnson  
**Program Director:** Ginger Ignatoff

**ACM Council**  
**President:** Maria Klawe  
**Vice-President:** David S. Wise  
**Secretary/Treasurer:** Telle Whitney  
**Past President:** Stephen R. Bourne  
**Chair, SGB Board:** Alan Berenbaum  
**Chair, Publications Board:** Robert Allen

**Members-at-Large:** Roscoe Giles (2002–2006); Denise Güler (2000–2004); David S. Johnson (1996–2004); Michel Beaudouin-Lafon (2000–2004); Edward Lazowska (2000–2004); Barbara Ryder (2002–2004); Gabriel Silberman (2002–2006)  
**SGB Council Representatives:** Stuart Feldman (2002–2004); Mark Scott Johnson (2001–2005); Jim Cohoon (2001–2005)

## Board Chairs and Standing Committees

**Education Board:** Russell Shackelford; **SGB Board:** Alan Berenbaum;  
**Membership Board:** Terry Coatta; Mike Macfaden;  
**Publications Board:** Robert Allen **USACM Committee:** Eugene Spafford, Barbara Simons

## SIG Chairs

**SIGACT:** Harold Gabow; **SIGAda:** Currie Colket;  
**SIGAPL:** Robert G. Brown; **SIGAPP:** Barrett Bryant;  
**SIGARCH:** Norman Jouppi; **SIGART:** Maria Gini;  
**SIGBED:** Janos Sztipanovits; **SIGCAPH:** John Goldthwaite;  
**SIGCAS:** Tom Jewett; **SIGCHI:** Joseph Konstan;  
**SIGCOMM:** Jennifer Rexford; **SIGCSE:** Henry Walker;  
**SIGDA:** Robert Walker; **SIGDOC:** Scott Tilley;  
**SIGecom:** Michael Wellman; **SIGGRAPH:** Alain Chesnais;  
**SIGGROUP:** Wolfgang Prinz; **SIGIR:** Jaime Callan;  
**SIGITE:** Edith Lawson; **SIGKDD:** Won Kim;  
**SIGMETRICS:** Leanna Golubchik; **SIGMICRO:** Kemal Ebcioglu;  
**SIGMIS:** Janice Spior; **SIGMOBILE:** Victor Bahl;  
**SIGMOD:** M. Tamer Ozsu; **SIGMULTIMEDIA:** Ramesh Jain;  
**SIGOPS:** Keith Marzullo; **SIGPLAN:** Michael Burke;  
**SIGSAC:** Sushil Jajodia; **SIGSAM:** Emil Volcheck;  
**SIGSIM:** John Tufarolo; **SIGSOFT:** Alexander Wolf;  
**SIGUCCS:** Robert Paterson; **SIGWEB:** Peter Nuernberg

For information from Headquarters: (212) 869-7440

## ACM U.S. Public Policy Office:

Jeff Grove, Director  
1100 Seventeenth St., NW  
Suite 507  
Washington, DC 20036 USA  
+1-202-659-9711—office  
+1-202-667-1066—fax  
jeff\_grove@acm.org

# COMMUNICATIONS OF THE ACM • A monthly publication of the ACM Publications Office

ACM, 1515 Broadway, New York, New York 10036-5701 USA (212) 869-7440 FAX: (212) 869-0481

## Editorial Advisory Board

Gordon Bell; Hal Berghel; Grady Booch;  
Nathaniel Borenstein; Vinton G. Cerf;  
Kilnam Chon; Jacques Cohen; Larry L. Constantine;  
Jon Crowcroft; Peter J. Denning; Mohamed E. Fayad;  
Usama Fayyad; Christopher Fox; Ravi Ganesan;  
Don Gentner; Don Hardaway; Karen Holtzblatt;  
Barry M. Leiner; Pattie Maes; Eli Noam;  
Cherri Pancake; Yakov Rekhter; Douglas Riecken;  
Ted Selker; Dennis Tschirtzis; Ronald Vetter

## Publications Board

Chair: Robert Allen; Gul Agha;  
Michel Beaudouin-Lafon; Ronald F. Boisvert;  
Adolfo Guzman-Arenas; Wendy Hall;  
Carol Hutchins; Mary Jane Irwin; M. Tamer Ozsu;  
Holly Rushmeier

## ACM Copyright Notice

Copyright © 2004 by Association for Computing Machinery, Inc. (ACM). Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and full citation on the first page.

Copyright for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or fee. Request permission to publish from: Publications Dept. ACM, Inc. Fax +1 (212) 869-0481 or email <permissions@acm.org>

For other copying of articles that carry a code at the bottom of the first or last page or screen display, copying is permitted provided that the per-copy fee indicated in the code is paid through the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, 508-750-8500, 508-750-4470 (fax).

## Subscriptions

Annual subscription cost is included in the society member dues of \$99.00 (for students, cost is included in \$40.00 dues); the nonmember annual subscription is \$179.00. See top line of mailing label for subscription expiration date coded in four digits: the first two are year, last two, month of expiration. Microfilm and microfiche are available from University Microfilms International, 300 North Zeeb Road, Dept. PR, Ann Arbor, MI 48106; (800) 521-0600.

**Single Copies** are \$8 to members and \$17 to nonmembers. Please send orders prepaid plus \$7 for shipping and handling to ACM Order Dept., P.O. Box 11414, New York, NY 10286-1414 or call (212) 626-0500. For credit card orders call (800) 342-6626. Order personnel on duty 8:30-4:30 EST. After hours, please leave message and order personnel will return your call.

## Notice to Past Authors of

**ACM-Published Articles** ACM intends to create a complete electronic archive of all articles and/or other material previously published by ACM. If you were previously published by ACM in any journal or conference proceedings prior to 1978, or any SIG newsletter at any time, and you do not want this work to appear in the ACM Digital Library, please inform permissions@acm.org, stating the title of the work, the author(s), and where and when published.



# News Track |

---

## Blurred Vision

The U.S. Government, bowing to repeated Secret Service requests, is now deliberately obscuring its highest-quality aerial photographs over Washington in an effort to hide visible objects on the roofs of the White House, Capitol, and Treasury departments. The Associated Press reports this blurring policy also includes views of the Naval Observatory where the Vice President lives. Experts are concerned the unusual decision reflects a troublesome move toward new government limits on commercial satellite and aerial photography. Moreover, the effectiveness of blurring one set of government-funded photos is questionable since tourists can see the roofs of these noted buildings from dozens of taller buildings in downtown D.C. Secret Service spokesperson John Gill explained the agency worried that high-altitude photographs, so detailed that pedestrians can be seen on crosswalks, “may expose security operations.” Interestingly, the policy does not extend to detailed pictures of the Pentagon, Supreme Court, Justice Department, or FBI and CIA headquarters.

## (Keep) Phone Home

An increasing number of companies are joining the ban on camera phones. Health clubs and popular gambling casinos were quick to forbid the discrete picture-taking devices for obvious reasons of patron pri-

vacy and security. Now major employers are banning camera phones on the job amid growing fears these devices pose serious threats to company secrets and worker privacy, reports *USA Today*. Firms worry employees will use the phones to send images of new products or other company information, or else take pictures of unsuspecting co-workers in locker rooms or bathrooms, which may lead to business or legal risks. General Motors, Texas Instruments, DaimlerChrysler, BMW, and Samsung are a few of the recent corporations to issue “no camera phone” zones in the workplace. U.S. courthouses are also starting to adopt the ban fearing camera phones can be used to photograph jurors or undercover agents serving as witnesses.

## Love/Hate Relationship

In other phone news, the cell phone topped the list of the annual Lemelson-MIT Invention Index as the invention users love to hate. Unlike recent U.K. surveys that found the majority of users developing a strong, sentimental attachment to their phones, the U.S.-based study concluded that cell phones have become a necessary part of life, but the devices in fact drive most users crazy. “The interconnections you get from the cell phones is a very positive thing. The downside is that you sometimes want to be alone,” explained Lemelson Center Director Merton C. Fleming. Among the other inventions getting a collective raspberry: the alarm clock, television, razor, microwave oven, computer, and answering machine.

## Image Makeover

The image of the stereotypical Internet user, long characterized as an antisocial geek with no friends and no interest in the real world, has been shattered by a new survey that finds the typical Web surfer one

that shuns TV and is quite the social animal. The UCLA World Internet Project ([ccp.ucla.edu/pages/internet-report.asp](http://ccp.ucla.edu/pages/internet-report.asp)), a three-year survey of Net and non-Net users in 14 countries, produced global comparisons data on social, political, and economic effects of the Net. The survey shows a digital gender gap in all participating countries and surprisingly high levels of online use among the poorest citizens of the surveyed countries. The gender gap is most prevalent in Italy where 41.7% of the users were men; 20.1% women. The lowest gap was in Taiwan (where user makeup is 25% men, 23.5% women). Internet users in all surveyed countries spend more time socializing and exercising. In fact, Net surfers also read more books (except in Germany and the U.S.). South Koreans trust Net info the most; Swedes are the most skeptical. And China boasts the most active Net socializers.

---

## Robot Scientist

A robot that can formulate theories, perform experiments, and interpret results (and does all more cheaply than its human counterpart) has been created by a team of scientists at the University of Wales in Aberystwyth, U.K. *Nature* magazine reports the Robot Scientist isn't as intelligent as, say, the leading chess-playing computer, but combining the smarts of a computer with the agility to perform real scientific problems is a major engineering feat. In a recent test against human competitors, the robot worked out which genes in yeast are responsible for making vital amino acids; even beating out a biologist who claims he hit the

wrong key at one point. The robot was also more sensitive in financial concerns; human players were penalized for overspending. Ron Chrisley, University of Sussex, is quick to point out that having robots work in research labs is really nothing new. "We've had robot scientists for a long time now. But in the past we've always called them 'grad students'."

---

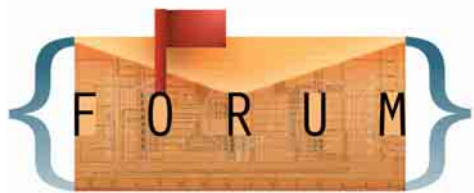
## Rooms with a View

A million-dollar two-story home constructed of walls made mostly of windows that turn from clear to opaque, or to computer screens, to speakers, or to television screens was a main attraction at the recent Sundance Film Festival. The 6,000 square-foot prototype house, built in Park City, UT, is the creation of Anderson Windows and Time Warner and considered by both a research project. The windows, found inside and out including the roof, between rooms, stacked atop each other in bed-

rooms and bathrooms, are fitted with a microfiber LCD screen, which makes them opaque or able to display light from a TV projector. Touch-screen computer monitors are fully integrated into each window, allowing them to receive and display information without projection. Architect Michael James Plutz designed the abode with "wall plans," not floor plans. Of the resulting structure, he says: "We are a multitasking species. We want to do everything in every room. We should expect our windows to do the same." ■

---

Send items of interest to [cacm@acm.org](mailto:cacm@acm.org)



# Principles of Logical Thought Before Technical Details

**S**IMONE SANTINI'S COMMENT (Forum, Dec. 2003) brought to mind something I constantly tell my students: There is nothing new about computers; they simply mechanize the application of logical principles that have been around since Socrates, a logic that can be formally expressed through the mathematics invented by George Boole.

The two best courses I ever took for learning to work with computers were formal logic and Boolean algebra. The former was taught by the philosophy department, the latter by the mathematics department. Neither professor knew much about computers. Indeed, the subject of computers never came up. Yet my experience suggests that everything a computer does is based on the principles expressed in these two courses.

Thus, Santini's assertion appears to be correct: The mind should be developed first; technical details should come later. It's a mistake to sit people in front of a computer as the first step in teaching them how to program it to do something useful.

I have found that people using their brains get ahead further and faster than people who know only technical details. In the end, it is not the computer that is important but the results they help achieve. Knowing what to do with the machine is far more important than knowing how to do things with the machine. Well-considered definitions of what things to do inspire us to greater accomplishments than trivial applications. Applying basic principles of logical thought shields us from being overwhelmed by technical detail.

PETER G. RAETH  
*Fairborn, OH*

**T**HE NOTION THAT WE'RE going to fix, or even help, U.S. educational problems by putting a computer on every desk or even in every classroom is ridiculous. Simone Santini's Forum comment (Dec. 2003) was eloquent on this point.

Computers in the classroom are what the law calls an "attractive nuisance," like a swimming pool in your neighbor's backyard. The definition is "any inherently haz-

ardous object ... that can be expected to attract children to investigate or play." In the case of classroom computers, the hazards are encouragement for superficial thinking, facilitation of general academic laziness up to and including plagiarism, ease of access to materials children ought not to see, and general time wasting, including game playing and aimless Web surfing.

As Santini pointed out, computers are useful in a school library, but their value in the classroom is far from proven. If Aristotle, Newton, and Einstein were educated without computers, how much of a handicap is not having one?

My one disagreement with Santini concerns is over the issue of school funding. I do not lament "the idea that public funding of education is destined to decrease." I embrace it. When it decreases to zero, children now confined to public schooling will finally get the same chance to excel as the U.S.'s seven million private school and home-schooled children have had for decades. The best alternative to public



funding is not corporate sponsorship but self-funding.

For decades, government has brainwashed parents into trusting the bureaucrats to educate their children. The result has been a disaster. The only way they can take back control is to pay fully for their children's schooling themselves.

MARK WALLACE  
Irvine, CA

## Why Reinvent When We Have Open Source?

**R**OBERT L. GLASS'S PRACTICAL Programmer column "A Sociopolitical Look at Open Source" (Nov. 2003) discounted the practicality of users fixing their own source, since the only users capable of doing so are interested only in system programs. Application end users hire consultants to do the fixing for them, and open source works out very nicely indeed.

I'd like to understand the reasons software developers insist on reinventing the wheel despite the availability of open source solutions. As pointed out in Eric Steven Raymond's *The Art of Unix Programming* ([www.faqs.org/docs/artu/](http://www.faqs.org/docs/artu/)), the notion of "not invented here" is not simply a response to the lack of transparency in proprietary solutions. The consequence in the open source world of programmers' roll-your-own tendencies is that, except for a few areas with category-killer solutions, most problems are covered with haystacks of partial solutions of questionable utility.

DAVID HAWLEY  
Tokyo, Japan

## Still Seeking Software Productivity

**I**N OUR ARTICLE "MEASURING Software Productivity in the Software Industry" (Nov. 2003), we developed a rationale for measuring the productivity of development and support environments as a first step toward solving the problem of software productivity.

A 1995 *BusinessWeek* article surveyed productivity in 25 industries, deriving percent productivity change over the previous five years while reaching some astounding conclusions outlined in another article in *Software Developer & Publisher* (July 1996). Computers (hardware) were first (+153%), semiconductors second (+84%), and software dead last (-11%). Independently, the Standish Group in Feb. 1995 published a report on the software industry, further supporting the negative productivity findings.

These facts were at odds with the implications of a 1991 *BusinessWeek* article "Software Made Simple," which interviewed key players in the world of object-oriented programming. The article cited the naysayers who compared OOP to artificial intelligence as just another computer industry buzzword. Defending this comparison, the authors wrote that, unlike AI, object technology would have "an immediate, practical payoff."

Much more recently, *BusinessWeek* (Jan. 12, 2004) discussed the annual percent productivity gains of various major industries from 1998 to 2001. Computer chips (+19%) and consumer electronics

(+18%) had the greatest gains. Software was last (-1%).

These facts continue to support our observations, measures, and rationale about low productivity using current software technology, as well as the steps needed to turn it around. This has not seemed to stop the software industry from continuing to put itself further behind in the productivity race each year.

DONALD ANSELMO  
Phoenix, AZ  
HENRY LEDGARD  
Toledo, OH

## Spare Me the Self-Service

**T**HE NEWS TRACK ITEM "Checked Out" (Jan. 2004) cited the skyrocketing use of self-service kiosks. I, for one, am not convinced that the 14.5 million passengers statistic really means what it suggests. As a recent first-class passenger on a major airline, I was directed to the self-service kiosk. Glancing at the counter, I noticed that all passengers were being directed to the "self-service" line; there was no personal service. This is just another way station on the road to eliminating personal (but costly) face-to-face service without lowering the price to the consumer. It didn't cost me any less in effort (checking in) or money (ticket price), but it most certainly saved the airline some of both.

Personally, I would prefer if the self-serve kiosks at airlines and grocery stores (along with telephone menu systems corporations use to punish us) would go away.

DWAYNE NELSON  
Washington, D.C.

## Lineage of the Internet's Open Systems Foundation

I'D LIKE TO POINT OUT AN important contradiction in Coskun Bayrak's and Chad Davis's "The Relationship between Distributed Systems and Open Source Development" (Dec. 2003), which stated: "The original specifications of a layered protocol stack upon which heterogeneous distributed systems could build their communication systems, and upon which TCP/IP was closely modeled, is of course the Open Systems Interconnect model [5]." Reference 5 is Peterson's and Davie's *Computer Networks: A Systems Approach*, which stated: "The Internet and ARPANET were around before the OSI architecture, and the experience gained from building them was a major influence on the OSI reference model."

Though this obvious contradiction was made by the authors, a thorough review should have caught it.

YITZCHAK GOTTLIEB  
*Princeton, NJ*

THE ARTICLE BY COSKUN Bayrak and Chad Davis would have been more informative and more interesting had it let us in on the mechanisms the Linux community uses to sort the good from the bad features and code so we "can bet the product will be substantially better in only a few weeks," as they say.

Moreover, the authors assert that TCP/IP was closely modeled on ISO's OSI. Vinton Cerf and Robert Kahn began their work on TCP in 1973, whereas OSI

came out in 1983. What is the basis for the authors' claim?

ALEX SIMONELIS  
*Montreal, Canada*

### *Authors Respond:*

THOUGH WE ACKNOWLEDGE our semantic error we were lax not to catch prior to publication, it shouldn't be viewed as an indictment of the general line of our argument. A key part of our purpose, and of this citation in particular, was to highlight the open system nature of the Internet's technological foundations, at the core of which is the TCP/IP suite.

From a historical perspective, the evolution of OSI vs. TCP/IP is a subjective issue. One may be interested in the invention of specific protocols or, alternatively, the invention of open system concepts, including layering and interoperability. However, the citation reinforced the openness of the Internet's core communications mechanisms. The notion of what won—OSI or TCP/IP—is not relevant in this sense. Relevant is the nonproprietary, open system qualities of the Internet protocol suite.

We do not support the suggestion that the protocols within the TCP/IP suite were derived from OSI. We would, however, like to know whether anyone finds fault with our characterization of the TCP/IP stack as an essentially and significantly open system. In regards to history and our discussion of openness, the OSI project was meant to clarify the role of openness in the Internet's developing infrastructure.

The lineage of the individual protocols in the TCP/IP stack clearly predate the concept of the standardized and layered stack itself; TCP itself dates to the 1960s. However, the notion of the TCP/IP protocol stack isn't the same as the individual protocols themselves. It wasn't until 1989 that the architecture of Internet communications was written into RFCs 1122 and 1123, long after OSI and other projects brought attention to the significance of openness and interoperability.

We appreciate the opportunity to revisit these issues. The history of these systems had been on the margins of our thinking prior to this reexamination of our topic. We thank Gottlieb and Simonelis for initiating this dialogue, encouraging us to delve further into the genealogical elements of our topic. We also hope to hear more about our discussion of the Internet's fundamental reliance on communication systems demonstrating high levels of openness.

Concerning the mechanisms of development utilized by open source projects, including Linux itself, we were unable in the context of the article to fully explore the inner structures of the open source development model and its intriguing similarities to the inner structures of distributed software systems.

COSKUN BAYRAK  
CHAD DAVIS  
*Little Rock, AR*

Please address all Forum correspondence to the Editor, *Communications*, 1515 Broadway, New York, NY 10036; email: [crawfordd@acm.org](mailto:crawfordd@acm.org).



# Superscaled Security

Exponential increases in computational speed, memory capacity, and bandwidth impose futuristic security demands and challenges.

**A**dvances in high-performance computing have found their counterpart in new security threats. Yet there is an interesting twist in that computational expansion tends to be relatively predictable, whereas security challenges are typically introduced and mitigated (when possible) in a more chaotic fashion. Few would have surmised, for example, that spam would have exceeded 60% of all email transmissions by the end of 2003, nor that detection software would require sophistication approaching Turing Test intelligence levels. It is useful, therefore, to consider some of the impacts of scaled-up computing on our overall security environment.

Certain rules continue to apply to computational evolution. Moore's Law (which anticipates processing power doubling approximately every 18 months while the equivalent price halves

in the same amount of time) has continued to endure. It is likely to surpass even Gordon Moore's own 1997 prediction [5] that it will "run out of gas" around 2017, as new materials and fabrication technologies emerge (including those introduced through the recursive application of improved computers into the manufacturing process). In 2003, Moore told the International Solid-State Circuit Conference "I remember thinking one micron (a milestone the industry passed in 1986) was as far as we could go because of the wavelength of visible light used at the time. Engineers then switched to ultraviolet light."

But as advancements in hardware continue to occur, security appears to be declining (although not necessarily at an equivalent rate). As Paul Kocher of Cryptography Research, Inc. asserted,

"Moore's Law, coupled with the business imperative to be more competitive, is driving vendors to build systems of exponentially increasing complexity without making security experts exponentially smarter to compensate."

"The current trend is to build systems that conceptually are secure, but in practice and probability the systems' ability to resist the efforts of a creative attacker are less," said Kocher in a pre-USENIX 2002 interview. "The idea is that security should be getting better, however, design flaws are becoming an increasingly catastrophic problem" [2].

Some believe that cryptography can help reverse this trend, and cryptosystems are playing a significant role in providing security assurances. Yet each generation of computers brings with it an obsolescence of some earlier cryptographic methods, usually considerably sooner than has been predicted. This will likely continue to be the case, and may even accelerate as new paradigms for algorithmic attacks, including distributed techniques, evolve. So this implies that if cryptography is used to sign and protect data and

software (especially for archival purposes) then a systematic method for updating these wrappers will need to be devised, lest trust in content be undermined.

Cryptography, though, cannot be expected to solve all security problems. PGP encryption guru Phil Zimmerman told London's 2003 Infosec security conference that Moore's Law is a "blind force" for undirected technology

vast quantities of information also increases the likelihood that this data will eventually be used for heretofore unknown and potentially nefarious purposes.

We may have no choice regarding such data dissemination, since global economic forces may be driving us toward total interconnectivity of all humans on the planet. This is ultimately feasible, since in the networking world,

against us." Whether the negatives will eventually outweigh the positives, in terms of adverse impacts on connectivity and usability, is yet to be determined.

There is a storehouse of data that we are looking forward to having online, which is currently located in libraries, recordings, and research databases. As David Sarnoff predicted in an article he wrote for the *New York Herald* in

---

## The ultimate question must be whether or not there will someday be a computational system that can prevent all forms of nefarious attack.

---

escalation. He explained that "the human population does not double every 18 months, but its ability to use computers to keep track of us does," and added, "you can't encrypt your face." He fears the series of initiatives in U.S. homeland security have far-reaching effects on privacy because "it has more inertia and is more insidious. When you put computer technology behind surveillance apparatus, the problem gets worse." We already yield a tremendous amount of personal information to our PDAs and permit tracking of our movements in exchange for continuous incoming telephone service, so the devices we voluntarily adopt may ultimately prove more untrustworthy than the monitoring being imposed. The increasing ability of computers to store and analyze

Gilder's Law dictates that the total bandwidth of communication systems will triple every 12 months. (By comparison, the number of humans is only expected to double from 6 to 12 billion by 2100.) Metcalfe's Law rates the value of a network as proportional to the number of nodes squared, so the merit of connectivity increases exponentially as units are introduced, while costs tend to remain stable. But this supposition of continually increasing payback is not necessarily correct. Jake Brodsky reminds us of Newton's Third Law ("For every action, there is an equal and opposite reaction"), noting that events are "going on under our noses this very minute: Security holes, hacking, and phreaking. The very tools that make this 'revolution' [in technology] possible are also being used

1922, "It is inconceivable that the development of the transmission of intelligence will go forward at a leisurely pace; everything points to a very great acceleration." One wonders whether Sarnoff might have imagined that such acceleration would, within the next decade, make it feasible for a laptop computer to hold the contents of the entire Library of Congress. Certainly it is imperative to ensure that all information will be replicated correctly in such compendia. Imagine an insidious virus that permutes documents such that history eventually reflects that Thomas Jefferson was the first president of the United States. This might not be so damaging, but other transformations could have dire results. Even if data is maintained intact, accessibility to powerful search and logic engines

might (justifiably or inappropriately) result in different interpretations of the “truth” than are currently commonplace.

Knowledge acquisition, as we know, is not merely equivalent to information gathering. Knowledge involves analysis and conjecture, whereas information can be obtained in a brute force fashion. For a long time, the sheer power of bulk data provided by the information age has prevailed, but as our chess-playing machines have demonstrated, progress eventually must include an “artificial intelligence” component. As Frederick Friedel, Gary Kasparov’s assistant exclaimed, “As Deep Blue goes deeper and deeper, it displays elements of strategic understanding. Somewhere out there, mere tactics are translating into strategy. This is the closest thing I’ve seen to computer intelligence. It’s a weird form of intelligence. But, you can feel it. You can smell it!”

The combination of human reasoning with computer-based data can be extremely powerful. Once the Library of Congress is reduced to the size of a microchip, we might choose to directly implant it into our heads so the contents would be immediately accessible to our thought processes. I recall a conversation with Princeton mathematician John Conway in which we mused about *Matrix*-style cyber-brain enhancements and whether people might eventually be discriminated against in employment if they refused to submit to elec-

tronic augmentation. Although I’d prefer to retain my Luddite status, Conway indicated his willingness to enter this Brave New World of embedded micro-technologies. Ethics and security issues abound—for example, recalls for bug fixes might become a bit daunting. Recent vendor speculations have included weaving chips into our clothing or concealing them in medicines, foods, and beverages. The phrase “I’ve got you under my skin” may take on an eerie meaning in the not-too-distant future.

Back in the world of large computational systems, as these continue to expand, von Neumann architectures will eventually be superseded by or augmented with neural networks, genetic algorithms, and DNA “soups” where even NP-complete problems may someday become solvable. Instead of using detection and eradication, security systems will contain adaptive features—enabling them to encapsulate and incorporate malware into themselves, thus evolving to resist new threats. Of course, viruses will be smarter as well, learning about the systems they infect, beyond basic exploitation of known vulnerabilities. Rather than downloading patches, inoculation software may be injected into networks to counter the effects of currently circulating deleterious code. One might go so far to say that such a mutational process imposed on computational systems will be a necessary aspect of its development, beyond

that which humans would be able to provide through programming and intentional design.

Clues to the manner in which architectural expansion may be mitigated are found in a Mead and Conway principle earlier intended for VLSI, as follows: “The task of designing very complex systems involves managing, in some highly structured way, the space and time relationships between the various levels of system building blocks so that the entire system will function as intended when it is finished.” Citing that statement, a 1998 NSF workshop report [1] concluded that “the software research literature is replete with design principles, from the highest to the lowest levels. In virtually all cases, the principles have neither been carefully classified for applicability nor validated for practical effectiveness.” Clearly more work is necessary. Scalability must be considered. As well, form and functionality in large-scale computing have not kept pace with each other for a number of years, and their disparity introduces the potential for a panoply of adverse consequences.

This disparity includes the relationship of our concept of trust to the necessity that computer systems maintain fundamentally deterministic behavior patterns by disallowing the use of self-modifying code. So if we relinquish control and allow evolutionary processes to occur, it becomes less clear how to assess trustworthi-

ness. Ultimately, we might want to permit systems to adjudicate among themselves whether or not they are to be trusted, since it may not be possible for scientists and engineers to appropriately assess a breed of computers with constantly changing logical infrastructures. But then how can we trust them to make the right decisions about each other?

Perhaps this can be understood by examining some concepts from complexity science. As Jim Pinto explained to the Chaos in Manufacturing Conference [6], critical complexity occurs when processing power becomes intelligence, and when connected intelligence becomes self-organizing. Emergent behavior results from organized complexity, such as can be found in mutations, selection, and evolution. William Roetzheim defines complexity science as “the study of emergent behavior exhibited by interacting systems operating at the threshold of stability and chaos.”

It would appear that such a threshold occurs whenever security breaches reach the point of threatening the stability of computational systems. For example, Gilder’s Law has allowed faster bandwidth to overcome the adverse impacts of spam. As well, Moore’s Law has provided increased computational speeds to allow background virus detection. As attacks have become more sophisticated, new levels of hardware exponentiation have arrived, pushing the instability threshold

back. But this will soon be insufficient, since attacks that include emergent behavior will likely only be able to be thwarted through intelligence and mutation. Self-modifying processes could therefore become the panacea rather than the plague, if we can determine how to deploy them effectively in our security toolkits.

So, the ultimate question must be whether or not there will someday be a computational system that can prevent all forms of nefarious attack. To simplify this issue, let us restrict our consideration to breaches initiated by humans. Ray Kurzweil refers to the time when computers exceed human intelligence as “The Age of Spiritual Machines” [4] and believes we will be entering this era within the next two decades. He conjectures that by the end of this century, self-replicating hardware, decentralization, and redundancy will remove security concerns, for all practical purposes.

On the other hand, Seton Hall’s philosopher/physicist Stanley Jaki, using Gödel’s incompleteness theorem, concludes that “the fact that the mind cannot derive a formal proof of the consistency of a formal system from the system itself is actually the very proof that human reasoning, if it is to exist at all, must resort in the last analysis to informal, self-reflecting intuitive steps as well. This is precisely what a machine, being necessarily a purely formal system, cannot do, and this is

why Gödel’s theorem distinguishes in effect between self-conscious beings and inanimate objects” [3]. The extrapolation from this must be that self-modifying code, without the ability to perform introspection, cannot surpass human intelligence, hence systems will remain vulnerable.

Which answer is correct? With any luck, we shall all live long enough to see how things turn out. If not, then perhaps future readers, happening across this article in their Library of Congress brain chips, will hopefully have retained the ability to either laugh, or shake their heads and sigh. **C**

## REFERENCES

1. Basili, V.R. et. al. NSF Workshop on a Software Research Program for the 21st Century, Oct. 15–16, 1998, published in *ACM SIGSOFT 24*, 3 (May 1999).
2. Hyman, G. The Dark Side of Moore’s Law. (Aug. 7, 2002); [siliconvalley.internet.com/news/article.php/1442041](http://siliconvalley.internet.com/news/article.php/1442041).
3. Jaki, S.L. *Brain, Mind and Computers*, 3rd ed. Regenery Gateway, 1989.
4. Kurzweil, R. *The Age of Spiritual Machines*. Viking Penguin, 1999.
5. Moore, G. An update on Moore’s Law. *Intel Developer Forum*, (Sept. 30, 1997); [www.intel.com/pressroom/archive/speeches/gem93097.htm](http://www.intel.com/pressroom/archive/speeches/gem93097.htm).
6. Pinto, J. Symbiotic life in the 21st century. In *Proceedings of the Chaos in Manufacturing Conference* (May 4, 2000); [www.calculemus.org/MathUniversalis/NS/09/symbiotic.html](http://www.calculemus.org/MathUniversalis/NS/09/symbiotic.html).

---

**REBECCA T. MERCURI** ([mercuri@acm.org](mailto:mercuri@acm.org))  
is a research fellow at the John F. Kennedy  
School of Government, Harvard University.

---

# ACM Fellows

**T**he ACM Fellows Program was established by Council in 1993 to recognize and honor outstanding ACM members for their achievements in computer science and information technology and for their significant contributions to the mission of the ACM. The ACM Fellows serve as distinguished colleagues to whom the ACM and its members look for guidance and leadership as the world of information technology evolves.

The ACM Council endorsed the establishment of a Fellows Program and provided guidance to the ACM Fellows Committee, taking the view that the program represents a concrete benefit to which any ACM Member might aspire, and provides an important source of role models for existing and prospective ACM Members. The program is managed by an ACM Fellows Committee as part of the general ACM Awards program administered by Calvin C. Gotlieb and James J. Horning.

The men and women honored as ACM Fellows have made critical contributions toward and continue to exhibit extraordinary leadership in the development of the Information Age and will be

inducted at the ACM Awards Banquet on June 5, 2004 at the Plaza Hotel in New York City. These 30 new inductees bring the total number of ACM Fellows to 498 (see [www.acm.org/awards/fellows/](http://www.acm.org/awards/fellows/) for the complete listing of ACM Fellows).

Their works span all horizons in computer science and information technology: from the theoretical realms of numerical analysis, combinatorial mathematics and algorithmic complexity analysis; through provinces of computer architecture, integrated circuits and firmware spanning personal computer to supercomputer design; into the limitless world of software and networking that makes computer systems work and produces solutions and results that are useful—and fun—for people everywhere.

Their technical papers, books, university courses, computing programs and hardware for the emerging computer/communications amalgam reflect the powers of their vision and their ability to inspire colleagues and students to drive the field forward. The members of the ACM are all participants in building the runways, launching pads, and vehicles of the global information infrastructure. **C**

## ACM Fellows

Rakesh Agrawal, *IBM Almaden Research Center*

Mostafa Ammar, *Georgia Institute of Technology*

Victor Bahl, *Microsoft Research*

Bonnie Berger, *Massachusetts Institute of Technology*

Elisa Bertino, *University of Milano*

John Carroll, *Pennsylvania State University*

Richard DeMillo, *Georgia Institute of Technology*

Barbara J. Grosz, *Harvard University*

Brent Hailpern, *IBM Thomas J. Watson Research Center*

Jiawei Han, *University of Illinois at Urbana-Champaign*

Mary Jean Harrold, *Georgia Institute of Technology*

Peter E. Hart, *Ricoh Innovations, Inc.*

Mark Horowitz, *Stanford University*

Paul Hudak, *Yale University*

H.V. Jagadish, *University of Michigan*

Anil Jain, *Michigan State University*

Ramesh Jain, *Georgia Institute of Technology*

Niraj Jha, *Princeton University*

Dexter Kozen, *Cornell University*

Yi-Bing Lin, *National Chiao Tung University*

Kathleen McKeown, *Columbia University*

Thomas P. Moran, *IBM Almaden Research Center*

Eugene W. Myers, *University of California, Berkeley*

Craig Partridge, *BBN Technologies*

Daniel A. Reed, *University of North Carolina, Chapel Hill*

Stuart J. Russell, *University of California, Berkeley*

William H. Sanders, *University of Illinois at Urbana-Champaign*

Scott Shenker, *University of California, Berkeley*

Gurindar Sohi, *University of Wisconsin*

Cornelis J. van Rijsbergen, *University of Glasgow*



# ACM Fellows

## Call For 2004 ACM Fellows Nominations

The designation “ACM Fellow” may be conferred upon those ACM members who have distinguished themselves by outstanding technical and professional achievements in information technology, who are current professional members of ACM and have been professional members for the preceding five years. Any professional member of ACM may nominate another member for this distinction. Nominations must be received by the ACM Fellows Committee no later than Sept. 7, 2004, and must be delivered to the Committee on forms provided for this purpose (see below).

Nomination information organized by a principal nominator includes:

- 1) Excerpts from the candidate’s current curriculum vitae, listing selected publications, patents, technical achievements, honors, and other awards.
- 2) A description of the work of the nominee, drawing attention to the contributions which merit designation as Fellow.
- 3) Supporting endorsements from five ACM members.

ACM Fellows nomination forms and endorsement

forms may be obtained from ACM by writing to:

### ACM Fellows Nomination Committee

ACM  
1515 Broadway  
New York, New York 10036-5701, USA

[nominate-fellows@acm.org](mailto:nominate-fellows@acm.org)

The forms can also be accessed from:  
[www.acm.org/awards/nomination\\_packet/](http://www.acm.org/awards/nomination_packet/)

Completed forms should be sent by Sept. 7, 2004 to one of the following:

ACM Fellows Committee  
ACM  
1515 Broadway  
New York, New York 10036-5701, USA

or  
[nominate-fellows@acm.org](mailto:nominate-fellows@acm.org)

or  
+1-212-869-0824 - fax

## Stay on Top of ACM News with **MemberNet**

Coming in future issues of MemberNet:

- How ACM's new Membership Charter affects you
- Expanded course offerings in the Professional Development Centre
- New computing initiatives to reach underserved groups
- Reader survey findings focus on content

And much more!

All online, in MemberNet: [www.acm.org/membernet](http://www.acm.org/membernet).

## Beware of Counting LOC

Seeking a better method for estimating system size in an attempt to measure knowledge content.

**B**y far the most common software sizing metric is source Lines of Code (LOC). When we count LOC we are trying to “size” a system by counting how many pre-compiled source lines we have created, or expect to create, in the final system. Since one of the primary purposes of counting LOC is to determine how big a system might be before we go and build it—say as an input to a project estimation process—we are presented with a dilemma. Before we build a system, we don’t have any lines of code yet, so how can we count them? Once we have written the lines of code and we can count them, we don’t need to estimate since we have the actual values. LOC counts for estimation, it seems, are only accurate

when we don’t need them and are not available when we do.

So how about using another sizing method, say Function Points (FPts) [1]? The standard International Function Points User Group (IFPUG) FPts approach involves counting and weighting input, output, and data storage elements with an adjustment thrown in for some aspects of the environment in which the system will operate. The hope is that such counts are available early on in a project, around the time when an estimate is needed. But there are a number of observations we can make. The first is that a common step of the FPts sizing procedure involves translating between FPts and LOC through a process called “backfiring” [2]. Given

that the original purpose of FPts was to get away from LOC altogether, this seems a little ironic. Another issue is that FPts as a software product measure suffers from a few drawbacks. Depending on which FPts convention we use, they may not count transform behavior (what many people actually mean when they say “function”), state behavior, platform interaction, design dependence, time-related activity, requirements volatility, and a number of other attributes that legitimately affect the “size” of a system. Some estimation procedures advertise the ability to size a system by module, component, object, business rule, or any number of other aggregations of requirements, design and executable software elements. However, in most cases, these counts must be accompanied by a factor that determines how many LOC there are per whatever is being counted. The procedure simply multiplies the input count by this factor and we are back at LOC again.

Clearly, in order to measure the putative size of a system, we must count *something*, but it seems every metric leads back to LOC. Some of the estimation

# The Business of Software

tools available are very good at converting a size into a projected estimate including schedule, headcount, effort, even predicting defect discovery and risk—provided, of course, we can give them an “accurate” size in LOC. Since many methods employ exponential equations based on size input, any variance on the input predicted size tends to be

knowledge is placed. We are actually measuring how much room the knowledge would take up if we put it somewhere. A LOC count assesses the amount of paper the system would consume if we printed it out. In fact, given an average figure for paper density, we could conceivably produce a weight-based metric for system size(!). Other metrics such

Compounding this is the fact that we really don’t want to measure knowledge anyway. What we really want to measure is our *lack* of knowledge which, of course, we don’t know.<sup>1</sup>

## Measurement and Indication

The art of sizing systems is not really measurement as we usually know it. The metrics we collect,

---

**We have to count something and executable LOC are countable, albeit too late to really be useful in early estimating.**

---

compounded on the projected output.

## Why We Can’t Count

**T**here are two reasons why counting LOC (or their equivalent) is difficult and may be ineffectual:

- We actually want to count how much knowledge there is (will be) in the system and there is no way to empirically measure knowledge.
- Even worse, what we really want to count is not the knowledge we have, but the knowledge we don’t have, since it is this (lack of) knowledge that really takes the time in building systems.

Interestingly, if we look at any of our software size measures, we see we are not counting knowledge at all—we are really sizing the substrate on which the

as requirements counts, memory size, object size, even FPts, really count how much space the knowledge would take up if we deposited it in different locations.

Since the days of Plato, we have pondered the subject of knowledge and wondered how to measure it. Plato believed that all knowledge was simply “remembered” from a previous life, as described in *The Republic*. In that regard, he predated most estimation methods, which look to historical calibration as the source of “accurate sizing.” But there is still no such metric as a “knowledge unit.” We can determine the number of pages or lines in a book, or even weigh it, but there is no way to empirically measure its knowledge content. The same is true, unfortunately, for computer systems.

such as LOC and FPts, are really indicators of the likely knowledge content and so, we expect and hope, of the time and effort necessary to build the system. The medical profession understands the difference between measures and indicators quite well. If I walk into a doctor’s office with a fever of 101°F, what is wrong with me? Well, we don’t know. This single metric does not diagnose my condition. What most doctors will do is collect further metrics: blood pressure, chemistry, and various other symptoms. When “sufficient” metrics have been collected, members of the medical profession use a specific phrase to describe their

---

<sup>1</sup>We really want to count our Second Order Ignorance (what we don’t know we don’t know), which is the major component of effort. Armour, P.G. *The Laws of Software Process*, Auerbach, 2003, p. 8

analysis: "...these metrics *indicate* a particular condition." Sometimes, metrics will contra-indicate the condition and must be explained. So it is with LOC or any other system size measure. If one system is expected to have twice as many LOC as another, it indicates it will take a lot longer to create (generally proportional to the exponent of the size). But it may not.

### Other Factors

**W**e have long recognized that size is not everything in estimation. For instance, certain types of systems require much more effort than others. Embedded real-time systems typically require a lot more effort, and usually more time, than business systems. Many estimation approaches qualify the size by assigning a "productivity factor" related to the type of system. This name is misleading. People who create real-time systems are not less productive than people who create business systems even though their "productivity factors" are usually lower. The factors do not address productivity, they address knowledge density. Real-time systems factors are lower because such systems have a higher density of knowledge than business systems—we have to learn more about them to make them work. Anyway, the work, and the knowledge, in these domains are quite different and any attempt to equate them is suspect.

### LOC is Not Line of Code

If we look more closely at what LOC represents we see there is a fundamental misassumption. For sizing and estimation purposes it does not actually mean "Line of Code." Let me explain. Some estimation methods allow you to calculate a "productivity rate." This is the average rate at which the sizing metric is created, which is derived by dividing the total system size by the total effort. Using LOC, this unit would be LOC per staff month. If we then determine which phase of the life cycle is most productive by dividing the total size by the effort expended in that phase, it sometimes turns out that the most productive phase is project management and the least productive is programming due to the high effort in that phase. Clearly this is bogus, since the only phase that actually produces LOC is the programming phase. LOC is an indicator of system size, but it is not the system size. In fact, if we consider the activity as knowledge acquisition rather than transcribing LOC, it is entirely possible that programming would be less efficient than planning at uncovering systems knowledge.

Another indicator of the true nature of LOC is the way we count them. A company I work with created a voluminous book just on how to count LOC: ignore comments, one source statement per line (what about Lisp?), don't count test code (say for stubs), only count the invoca-

tion of reused code, and so forth. The question is why? Why don't we count comments or test code? If we need comments or need test code, why should they be left out? The usual response to this is "they are not included in the final product delivered to the customer." But so what? Comments require work, in a manner similar to actual LOC. If we need to write test code, we need to write it just as we need to write the final code. Anyway, what if we write executable code that doesn't work, is redundant, or is written to support a future release that this customer will not use. Why do we count those?

### Fully Burdened LOC

The "...not included in the product" criterion is not relevant. The real reason we don't count comments is *scalability*. We make the assumption that the comments and test code (and redundant and future code) are scalable with respect to the executable lines of code. Therefore, we don't need to count them separately if we assume that the executable LOC contains them. The same is true for requirements, designs, and plans, none of which we deliver to the customer either, but all of which we must create if we want to build executable LOC that will actually execute. We assume that if the LOC include all of these, then they are the LOC we are looking for. The concept is identical to the idea of "fully burdened labor rate." The cost of an

# The Business of Software

employee to a company is not just that employee's salary, it is the salary plus the apportioned overhead of all the expenses necessary to allow that employee to function in the workplace. This includes costs such as lighting, heating, and rent, none of which accrue to the employee, but are necessary. The employee does not work alone in a field, but in a building with other employees and an infrastructure that allows each employee to be effective within a system.

The same is true for LOC. A line of code doesn't do anything, unless it is included with all the overhead necessary to make it work with all the other LOC in a real system. For estimation purposes LOC does not mean "line of code," it means "line of commented, test-code-written, requirements gathered, planned, designed...code." This is not the same as LOC.

## The Count is Not the Code

We can cheat and produce more LOC by not commenting, not planning, and other tactics. But then we are not talking about the same LOC. In reality, an executable LOC count is simply a count of something we assume will scale proportionally to the knowledge content of the system. But it is not the knowledge and it is not the system. Function Points are not particularly special; they simply count other countable things. The estimation processes we have created are all tunable to some degree to get

over the issues that limit the LOC → knowledge content relationship. If we have experienced developers, we have less effort. The LOC count is not actually smaller, but because they are experienced, they have less to learn. If we reuse code, we can build a bigger system (more LOC) at the same level of effort, since we can borrow the knowledge without having to get it for ourselves. Real-time systems have a higher knowledge density than IT systems, so the knowledge-per-LOC is higher and the number of LOC is usually lower.

## Doctor the Numbers

The trick is to take a page from the medical profession's metrics book. Simply collecting one metric rarely gives us the answer—on the contrary, it usually gives us the question. What we should do is collect a bunch of sizing metrics and track them to see what the set of them indicate. A company I worked with did this. In collecting around 20 metrics to see how useful they might be in predicting performance, they found only 12 that seemed to have any correlation at all. Only eight had a strong correlation. The winner? The strongest independent correlating indicator was the number of customer-initiated change requests received in the first month of the project. Clearly, this number has very little to do with the actual final size of the system, but proved to be a powerful guide to the overall

effort that would be expended and the time it would take, for reasons that are quite intuitive.

We have to count something and executable LOC are countable, albeit too late to really be useful in early estimating. But we must have some common metric that helps us size systems, and if a couple of millennia of epistemology haven't come up with a unit of knowledge or a way of counting it, I guess we'll have to make up our own.

## A LOC by Any Other Name

I have thought for a long time that the worst thing about LOC is the name. If we had just thought to call it "System Sizing Metric" or "Knowledge Unit" which (coincidentally!) has a nearly 1:1 ratio with a count of the executable LOC (allowing of course, for levels of commenting, type of system, amount of test code, and so forth) we might be closer to counting what we really need to count. ■

## REFERENCES

1. Albrecht, A.J. Measuring application development productivity. In *Proceedings of the IBM Application Development Symposium*, (Monterey, CA, Oct. 1979), pp. 83–92.
2. Jones, C.T. Backfiring: Converting lines of code to function points. *IEEE Computer* 28, 11 (Nov. 1995), 87–88.

---

## PHILLIP G. ARMOUR

(armour@corvusintl.com) is a vice president and senior consultant at Corvus International Inc., Deer Park, IL.

---



# Who is Liable for Bugs and Security Flaws in Software?

Attempting to determine fault and responsibility based on available evidence.

Last October, the *New York Times* published a story on a lawsuit in California being launched against Microsoft in the State Superior Court [3]. The suit, which is trying to get class-action status, claims Microsoft violated California consumer protection laws by “selling software riddled with security flaws.” One concern is that, so far, software companies have protected themselves from product liability lawsuits by selling customers a license to use their software rather than the actual product and by requiring customers to sign off on a lengthy list of caveats and disclaimers. But this mode of licensing has come under fire as the computing world has become deluged with viruses that exploit flaws in software products.

When a reporter asked me to comment on this story, I found myself deliberating over several questions: One is whether or not software companies should indeed be held responsible for the quality of their products, like

other companies. If General Motors or Ford sell faulty automobiles, you can be sure the courts will hold them liable

when people or property are damaged. Why not software companies? Second, even if software companies are liable for product quality, can they realistically be expected to eliminate security flaws as well as more common defects? A third question is why has Microsoft become the focus of the latest lawsuit. Is the world’s largest

software company particularly bad at quality and security?

The potential damage to individuals and organizations from software defects is real and costly. The National Institute of Standards and Technology (NIST), for example, issued a study in July 2002 that claimed software quality cost the industry nearly \$60 billion a year. Users bear about two-thirds of that cost, so this is clearly an issue for everyone [2].

First, as for whether software companies should be held accountable for product quality, the answer has always been yes, to a degree. The issue is really to what extent software companies are being held financially liable. Successful firms, including Microsoft, have learned how to respond to customer complaints and fix bugs in their products, either in special “point releases” or new versions. So far, though, no court of law seems to have found Microsoft legally liable for flawed software, and that is why there is a new lawsuit in California [3]. In other cases, however, customers have received large financial settlements from soft-

ware companies. For example, in February 2003, the manufacturing company Daskocil won \$2.3 million in damages in an arbitration case against J.D. Edwards for shipping a “defective product” [4]. This case is not unusual. Almost every enterprise software company has faced customer lawsuits, and there are no doubt more on the way, with

bly always have some defects. But companies delivering software that exceeds the bounds of common industry practice are vulnerable to penalties. Because some software companies are much better than others at preventing, detecting, and fixing defects, it seems to me that many firms can do better and that courts should hold software firms more

ment, even 0.15 defects per thousand lines of code translates into 150 bugs for a million-line software program. Given that many software products today contain tens of millions of lines of code, it is no surprise to see hundreds of bugs in even the best-quality software from the best companies.

Moreover, when it comes to

---

**Most of the legal debates I know of center not around whether there is product liability, but around what is “acceptable” industry practice.**

---

security flaws a rising issue. In fact, there is an entire industry of expert witnesses and law firms that deals primarily with customer liability claims and desires to hold software companies more responsible.

Most of the legal debates I know of center not around whether there is product liability, but around what is “acceptable” industry practice—such as for making delivery dates as promised or producing features that meet or do not meet original specifications in a customer contract, as well as for the type and levels of defects. The general philosophy held by software customers, software vendors, the American Arbitration Association, and the U.S. courts seems to be that software is a uniquely complex product that will proba-

accountable for what they license or sell.

As for whether software companies can ever make their products error-free or invulnerable to security flaws, I think the answer is clear: no, they cannot. My own research suggests that companies have made remarkable progress in reducing defects, but they are still far from perfect. In a sample of 104 software projects from around the world, collected during 2002–2003, my colleagues and I found a median level of 0.15 defects per thousand lines of code as reported by customers in the 12 months after receiving a product.

In a smaller sample of 40 projects from the U.S. and Japan, reported on in 1990, we found a median level of about 0.6 defects [1]. But, despite the improve-

Web software, we have another problem: The Internet is an open system, like a mass-transit network in a city or like the world airline system. Sure, companies can make subways and airplanes more secure, and they are doing so, but at great cost and inconvenience to users. And, despite everyone’s best efforts, a determined villain—a terrorist or hacker—will find a way into the system. So unless software developers and users go back to closed systems with very limited access, they will always have to deal with the inconveniences of criminal activity. Nonetheless, as the industry continues to establish more demanding norms for what are acceptable levels of quality and security, software companies will have to respond—on their own or under pressure from gov-

ernment and the courts.

Finally, when it comes to whether Microsoft's products are particularly riddled with bugs and other security flaws, I think not. Again, when products like Windows 2000 have upward of 30 million of lines of code, hundreds of bugs are inevitable. Microsoft is also the most vulnerable software company because Windows and Internet Explorer constitute a ubiquitous platform for the desktop, its server products are inexpensive and gaining in popularity, and the other flagship product, Office, has some 95% of the applications productivity market. But I also think Microsoft is still struggling to figure out how to design more secure software in an open, insecure world. Since Internet Explorer appeared in 1995, followed by an array of server products, Microsoft has no longer been developing software simply for the desktop PC.

A little bit of research confirms that security flaws are hardly a Microsoft-only problem. I did a search of approximately 100 articles appearing in *Computerworld* on the subject during 2002–2003, and approximately half related to Microsoft products (various versions of Windows, Office, Passport, Windows Media, SQL Server, Commerce Server, and Internet Explorer). The other articles pointed out actual or potential flaws in a wide variety of commercial and open source products. Software

developers have been busy fixing these problems, but the number and scope are still worrisome.

For example, nearly every version of Unix and Linux distributed by Sun Microsystems, IBM, and Red Hat, as well as Apple's Mac OS X server software, which is based on Unix, had security problems in how they transferred data between different systems. Linux also had problems in its compression library. Cisco's Internetworking Operating System (IOS) had a flaw in how it processed data packets that could lead to a hacker attack. Sun Microsystems reported flaws in its Java Virtual Machine that could allow hackers to take control of Web browsers and steal user identifications and passwords. Netscape reported similar problems with its browser as well as JavaScript. Sun also reported problems in its XDR library product and user authentication software. Oracle warned of security flaws in its e-business suite of applications that could allow hackers to create a buffer overflow and cripple the browser interface program. IBM had to fix security problems in its iNotes product and Domino servers, made by its Lotus division.

In the open source community, Apache, the leading Web server, had flaws related to a potential stack buffer overflow and remote access features. Sendmail, the commonly used Internet email server, didn't properly check long email addresses and

this flaw could allow a hacker to gain control over the server. The Secure Sockets Layer (SSL) encryption software had an extensive list of problems that led to insecure implementations. Even the antivirus software company Symantec had a security flaw due to a feature that allowed users to access a free online service to check their computers.

In sum, software companies should be held responsible for security flaws and other defects, but within reason. Software products are complex to design and harder to test. No one has yet built a flawless software product of any size on the first try. And it will take more than a class-action lawsuit to fix human behavior when we live in an imperfect and open world. **C**

## REFERENCES

1. Cusumano, M. et al. Software development worldwide: The state of the practice. *IEEE Software*, (Nov.–Dec. 2003).
2. Keefe, T. Software insecurity. *Computerworld* (Aug. 5, 2002); [www.computerworld.com](http://www.computerworld.com).
3. Lohr, S. Product liability lawsuits are new threat to Microsoft. *The New York Times* (Oct. 6, 2003), p. C2.
4. Songini, M. J.D. Edwards user wins arbitration case against ERP vendor. *Computerworld*, (Feb. 21, 2002); [www.computerworld.com](http://www.computerworld.com).

---

**MICHAEL CUSUMANO** ([cusumano@mit.edu](mailto:cusumano@mit.edu)) is the Sloan Management Review Distinguished Professor at the MIT Sloan School of Management.

---

# How Computer Games Affect CS (and Other) Students' School Performance

Compulsive game playing, especially of the role-playing variety, risks failing grades and withdrawal of financial support from tuition-paying parents.



Discussions with hundreds of students at the University of Texas at Austin have convinced me that computer and video games, particularly when they involve role playing, do in fact ruin the social and scholastic lives of many students. I don't claim this is the world's top social malady, but many students—particularly those in computer science at my university—are addicted to these games, and it can be inferred that students at other institutions face the same problem. The evidence for my claim is anecdotal, so it suffers from all the shortcomings of any unscientific investigation. Despite this caveat, I am convinced of the negative effect games have on college students' academic performance and social relationships.

My methodology was simple: I asked students—all computer science majors in an undergraduate program—whether they knew someone whose scholastic or social life had been harmed by computer games. About 90% answered affirmatively, describing students whose fascination chained them to their apartments or dorm rooms for days, weeks, even semesters. Many admitted to having or having had this problem themselves. The effect is exacerbated by so-called role-playing games like *Age of Kings*, *Dark Age of Camelot*, and *Everquest*, with addictive power so great some call it *Ever-crack*. Players create characters and alter egos in cyberspace,

living out their personal fantasies, usually by adopting the traits they believe they lack in the real world. My informal surveys suggest there is something particularly addictive, if not sinister, about role-playing games.

Students have told me of parents withdrawing financial support from their children who play games at the expense of their studies; of intimate relationships undermined by an obsession with virtual worlds; and of roommates who no longer respond to human interaction while playing, transfixed as they are by the interface bridging virtual experience and human mind.<sup>1</sup>

Escapism is the primary appeal. Moreover, as the graphics get better and the game play more sophisticated, playing becomes even more engrossing.<sup>2</sup> It is easy to understand why anyone would want to escape our difficult and complicated world and fall into a vivid, compelling game environment. One can live there with little or no interaction with the ordinary world. With money, online bill paying, and groceries delivered to the door, one can peer almost full time into a computer screen.

Why would anyone choose to live as a character in a game instead of in the real world? Virtual real-

<sup>1</sup>A graduate student reported that when his roommate did not get out of his chair for days, he offered to buy the chair for \$50. The transaction was concluded, but the roommate just dragged over a pile of (dirty) clothes, sat on it, and continued to play. Other students told me of friends who manage tech-support computer help lines from home while simultaneously playing computer games.

<sup>2</sup>My informal surveys suggest that obsessive computer game playing is particularly problematic for male students.

ity, increasingly indistinguishable from real reality, is almost here, and many like it. Today's students are among the first generation to experience these games, and their life choices—or rather, the choice of a cyberlife—may herald the future choices of the general population.

How can I convince them not to compulsively play such games? One strategy is to point out the

objection, gamers might say they aren't worthy of others' interest or want time alone. But even if we acknowledge the validity of such claims, what of the disappointment they cause their parents? Here, gamers might question whether college students should make decisions based on whether or not they disappoint their parents.

Parental approval/disapproval may be a considera-

---

## My informal surveys suggest there is something particularly addictive, if not sinister, about role-playing games.

---

negative consequences games can have. Suppose they don't care. Suppose they say their lives are terrible, especially compared to their counterparts in cyberspace. Suppose further they claim role-playing games allow them to be courageous heroes they cannot be in real life or that communication with online friends teaches them something about relationships they would otherwise never know.

It is easy enough to imagine college students who don't care about their coursework and other scholarly pursuits. But a committed gamer might say that familiarity with the fast-paced computer world is actually job preparation for the 21st century. In short, if they prefer playing games to studying, finding an effective rejoinder is difficult. I could assert my preference for reading over gaming, but this would only reveal my subjective preference. Who am I to say anything about their preferences, especially when they don't seem to hurt anyone, other than possibly the players themselves? Gamers could claim that role-playing games make their lives better than the ones they live in the real world; thus, they might add, the consequences of playing are in fact positive.

If gamers deny the harm games might do to them—about which they may be correct—we could argue that they harm others. For example, real-life human relationships are more difficult to maintain with people who play games, since they spend so much time chained to their computers. To this

tion, but it is hardly the only or even most important one. If it were indeed the most important, how many of us would still have disappointed our parents? And how many of us would choose to live in accord with our parents' desires? Most of us would grant that personal autonomy holds sway over others' preferences, even those of our parents.

What about students who depend on their parents for financial support for their habit? In this case, parents are likely justified in pulling back from a project they deem worthless. Students who are financially independent can circumvent this obstacle. In fact, it is now possible to make a living from role-playing computer games, my students tell me, by creating and developing characters with special powers or virtues, then selling them for profit. One can expect entrepreneurs to find yet other ways to make money via role-playing and other computer games. Short of financial dependence on people who frown on compulsive game playing, there seems no conclusive argument concerning either harming oneself or others against a gamer's personal fixation. So, without good reason, we have failed to convince either the gamer or ourselves to give up a potentially addictive habit.

### The Gamer's Lament

Gamers could ask whether all role-playing games are really as bad as I claim. First, I doubt the benefits can possibly outweigh the costs in terms of per-



sonal time and energy. It is possible that games do facilitate social interaction, since many require players play together as a team and get to know each other. For the shy or the friendless, this is surely a comfort. Still, games' addictive effects—I feel justified in calling them that—suggest they may be more pleasant and engaging than real life; otherwise their appeal wouldn't be as great as it is.

In defending their habit, gamers might yet cite several more arguments:

- Most gaming experience doesn't lead to the collapse of one's social life;
- Gaming offers at least some positive lessons, especially as it relates to human-computer interaction; and
- Some gaming might actually aid a computer science education illustrating, say, the lessons of proper design for human-computer interaction.

Strong counterarguments can be made against all of them. The first concerns how much time might be spent gaming before the gamer's social relationships begin to break down. Determining with accuracy is difficult, but given my anecdotal evidence, the social and academic lives of people playing several hours a day are affected. As for positive lessons, some could involve cooperation and strategy, as well as how to design yet more games. But the fact is that games, especially those involving role playing, target primal areas of the brain satisfying primitive needs in the (primarily male) psyche.

As for the claim that at least some amount of gaming rounds out a contemporary computer science education, I simply reject it. Students can take all the interface/game design courses they want.

### Games and the Addictive Personality

If it could be shown scientifically that gamers become addicted to role-playing games, the case against them would be stronger. Why? Because addictions involve compulsive behavior that harms the lives of the people doing it. The key to understanding why we view addictions as harmful is the

conjunction of compulsivity and negativity. If only one, but not the other, is present we'd likely not refer to a particular behavior as an addiction. It may measurably hurt me to smoke a cigarette, but I'm hardly an addict if I quit immediately afterward. Likewise, I may be compulsive about many things without being addicted to them in a negative sense. I may be compulsive about eating healthy meals and exercising 30 minutes a day, but few would label me an addict. Compulsivity by itself doesn't make someone an addict—at least not in the negative sense. And the reason we aren't likely to view such people as addicts is primarily because there isn't anything negative about eating or exercising in moderation. However, if I did nothing but breath, eat, and exercise, it might be said I was compulsive in the negative sense, precisely because my behavior doesn't reflect moderation or temperance.

I have now introduced another idea—moderation—in my effort to understand this harmful addiction. Thus, I define addiction as a compulsive behavior, engaged in without moderation, that directly harms one's life.

Compulsive gamers are therefore addicts. While it would be difficult for them to deny their behavior is compulsive, they might still claim it affects their lives only in positive ways. Cigarettes, which may seem good to smokers, really are harmful to human health, independent of anyone's desire for a smoke. Similarly, role-playing games may appear challenging and fun to those playing them—because they may be so uncomfortable with or fearful of the world as it is.

We can only hope gamers begin to recognize that the real world holds much more reward for those with the courage to face it, promising more positive experience, knowledge, joy, and love than any world of computer-generated reality. ■

---

JOHN G. MESSERLY ([messerly@cs.utexas.edu](mailto:messerly@cs.utexas.edu)) is a lecturer in the Department of Computer Sciences at The University of Texas at Austin.

---



By John Yen

# EMERGING TECHNOLOGIES FOR HOMELAND SECURITY

**T**he catastrophic events of September 11, 2001, dramatically demonstrated the reach and effects of terrorism and made protecting the security of citizens a top priority and a major challenge for many governments worldwide. The formation of the Department of Homeland Security is an exemplar response by the U.S. to such a challenge, drawing upon the intellectual and technological capabilities of scholars, scientists, and technologists. In this special section, we highlight some of the key emerging technologies related to several critical areas in the realm of homeland security.

As outlined in *The National Strategy for Homeland Security*,<sup>1</sup> the scope of U.S. homeland security is quite broad. Six of the mission areas considered critical include: intelligence and warning; border and transportation security; domestic counterterrorism; protecting critical infrastructures; defending against terrorism; and emergency preparedness and response. The first three areas focus on, among other things, preventing terrorist attacks against the U.S., the next two on reducing vulnerabilities within the U.S., and the last area on minimizing the damage and recovering from terrorist attacks that have occurred in the U.S. Information and communication technologies (ICTs) must play a pervasive, central role in overcoming the many inherent informational challenges embod-

ILLUSTRATION BY  
ROBERT NEUBECKER

<sup>1</sup>*National Strategy for Homeland Security*. U.S. Office of Homeland Security, July 2002; [www.whitehouse.gov/homeland/book/](http://www.whitehouse.gov/homeland/book/).

ied within these six mission areas. Due to this wide scope, this special section seeks to provide a snapshot of some of the key emerging ICTs related to three of the mission areas (intelligence, protecting infrastructure, and emergency response). These three areas were selected because their informational challenges are not only critical but also highly inter-related.

The special section includes three articles on technologies to support intelligence and warning (including a summary by authors from the Defense Advanced Research Project Agency), two articles about protecting critical infrastructure, specifically cyber infrastructures (including an overview about technology and strategy for cyber security), and two articles regarding ICTs for enhancing emergency preparedness and response.

### **Intelligence and Warning**

**T**he article by Popp et al. surveys several DARPA-sponsored research thrusts for counterterrorism. These include: center-edge collaboration, analysis and decision support tools to support multi-agency information sharing and collaborative problem solving; ICTs involving transcription, machine translation, cross-language information detection and retrieval, and summarization, whose use will help to exploit the wealth of available foreign language speech and text; and pattern analysis tools intended to detect terrorist signatures from textual sources, representing and detecting patterns indicative of terrorist plots, and learning new terrorist patterns. The authors describe experiments conducted jointly by DARPA and several agencies within the U.S. intelligence and counterterrorism communities. The experiments, conducted by real intelligence analysts solving actual foreign intelligence problems using their own foreign intelligence data, indicated that analysts were far more productive using the IT tools provided by DARPA as opposed to using manually driven conventional means. Specifically, analysts spent much less time searching and preprocessing data (preparing data for analysis) and generating intelligence reports (summarizing analysis for decision makers) and much more time on doing the actual analysis (thinking about the problem).

Coffman, Greenblatt, and Marcus elaborate on one of the DARPA research thrusts for counterterrorism: pattern analysis. More specifically, two graph-based techniques for detecting suspicious activities of terrorist groups are described: sub-graph isomorphism algorithms (graph matching),

and social network analysis (SNA). To better deal with the complex nature of terrorist activities, the authors enhanced traditional algorithms using these techniques to operate on graphs whose nodes and edges are labeled by attributes. Moreover, because intelligence data is often incomplete, ambiguous, and/or unreliable, these enhanced algorithms also consider inexact matches between the intelligence data and the pattern graphs. Based on the difference of social interactions between normal non-terrorist groups and those between terrorists, SNA metrics can be defined to characterize suspicious activities. Bayesian classifiers are then used to classify suspicious activity graphs and time-varying graphs.

Kogut et al. describe a research effort designed to support counterterrorism analysts using software agents that can dynamically anticipate their information needs. The approach is inspired by psychological studies suggesting effective human team behaviors are based on maintaining a shared mental model of the team. The authors use an agent architecture called CAST (Collaborative Agents Simulating Teamwork) to support a computational shared mental model about the structure and the process of the team, enabling software agents to dynamically anticipate information needs of analysts, and to assist them by finding and delivering information relevant to their needs.

### **Protecting Cyber Infrastructures**

**B**oth government agencies and global enterprises rely on a secure network infrastructure for sharing critical information and conducting business transactions; therefore protecting IT infrastructures from cyber attacks is critical. The article by Saydjari provides a general overview of the components of cyber defense, discussing a variety of challenges and issues ranging from strategies and technologies to performance assessment. One of the challenges discussed is the lack of an experimental infrastructure and rigorous scientific methodologies for developing and testing next-generation cyber security technology in support of deploying large-scale cyber security systems. The article by Bajcsy et al. describes a project with an extensive research agenda to address this very challenge. The goal of the project, which involves nine teams from academia and industry, is to create an experimental infrastructure network to support the development and demonstration of next-generation information security technologies for cyber defense.

## Emergency Preparedness and Response

**W**hen a terrorist attack occurs, emergency response organizations and agencies at the federal, state, and local levels must quickly collaborate to assess the nature, severity, and effects of the attack, as well as to plan and coordinate their response actions. One of the two articles in this area focuses on wireless technology in support of first responders, while the other article describes the use of robotics technology for rescue operations. The article by Sawyer et al. describes a field study of police in Pennsylvania using mobile access technology to access an integrated justice information system. The goal of the study is to assess the potential impacts of 3G wireless networks on first responders. The authors' observations suggest that introducing wireless technology is unlikely to change existing organizational links within the legacy command, control, and communications infrastructure.

Murphy's article describes the use of robots after Sept. 11 in searching for victims and in assisting first responders in assessing the structural integrity of the World Trade Center foundation. The article discusses several research issues identified as a result of the experience: fundamentally, rescue robots must function within the physical constraints of complex environ-

ments and require special considerations for their mobility, sensing, and communication capabilities. Additionally, rescue robots must have good human-robot interaction to ultimately be accepted by the rescue workers.

## Conclusion

It is fortunate the U.S. is able to utilize a wide range of technological bases to develop ICTs for homeland security purposes. This ability is tempered by the new problems and challenges raised by contemporary terrorist activities. In the articles that follow you will see both the tremendous science and the problems of operations that will bind the efforts to make the U.S. safer. Some of the challenges are due to the complex and secret nature of terrorist activities, while others are due to environmental constraints. The widely varying yet highly interrelated homeland security challenges discussed in this section are intended to help spur the global IT community in designing and developing novel and creative multidisciplinary solutions for such challenges. ■

**JOHN YEN** (jyen@ist.psu.edu) is a University Professor of Information Sciences and Technology and the professor in charge of the School of Information Sciences and Technology at Pennsylvania State University.

© 2004 ACM 0002-0782/04/0300 \$5.00

Communications of the ACM  
~ 2004 Editorial Calendar ~

**APRIL** Etiquette for  
Human-Computer Relations

**MAY** Transforming Financial  
Services via ICT Architectures

**JUNE** Sensor Networks

**JULY** Medical Modeling

**AUGUST** Interactive Immersion in  
3D Graphics

**SEPTEMBER** End-User Development

**OCTOBER** E-Voting

**NOVEMBER** Bioinformatics

**DECEMBER** Blogging and the  
Ecology of the Internet

## Give Rebooting the Boot With VMware Virtualization Software

- Run multiple operating systems  
— Windows, Linux, NetWare —  
simultaneously on a single PC
- Develop, test, and deploy multi-tier  
apps on one box
- Shorten development cycles and  
increase hardware utilization
- Perform cross-platform server  
consolidation, backup, and disaster  
recovery in minutes, not days



More than 1.4 million users worldwide can't be wrong...  
find out about VMware Workstation 4 now!

[www.vmware.com/specials](http://www.vmware.com/specials)



**By Robert Popp,  
Thomas Armour,  
Ted Senator, and  
Kristen Numrych**

# COUNTERING TERRORISM INFORMATION TECHNOLO

**S**eptember 11, 2001 might have been just another day if the U.S. intelligence agencies had been better equipped with information technology, according to the report of Congress's Joint Inquiry into the events leading up to the Sept. 11 attacks [9]. The report claims that enough relevant data was resident in existing U.S. foreign intelligence databases that had the "dots" been connected—that is, had intelligence analysts had IT at their disposal to access and analyze all of the available pertinent information—the worst foreign terrorist attack to ever occur on U.S. soil could have been exposed and stopped.<sup>1</sup>

In the aftermath of the Sept. 11th terrorist attack, the U.S. Defense Advanced Research Projects Agency (DARPA)—the U.S. Defense Department agency that engages in high-risk/high-payoff research for the defense department and national security community—focused and accelerated its counterterrorism thrust. The overarching goal was to empower users within the foreign intelligence and counterterrorism communities with IT so they could anticipate and ultimately preempt terrorist attacks by allowing them to find and share information faster, collaborate across multiple agencies in a more agile manner, connect the dots better, conduct quicker and better analyses, and enable better decision making [5, 8].

---

<sup>1</sup>It is important to point out the implicit assumption—and the resultant misconception—that all the dots were unambiguously preexisting; a significant portion of them were created by some a priori context, which may or may not have been correct. Moreover, when the Sept. 11 terrorist event shifted into the past, that ambiguous context became much clearer because the event then became part of the historical record [6].

# THROUGH GY

*Developing the information-analysis tools for an effective multi-agency information-sharing effort.*

**T**he world has changed dramatically since the Cold War era, when there were only two superpowers (see Table 1). During those years, the enemy was clear, the U.S. was well postured around a relatively long-term stable threat, and it was fairly straightforward to identify the intelligence collection targets. Today, we are faced with a new world in which change occurs very rapidly, and the enemy is asymmetric and poses a very different challenge; the most significant threat today is foreign terrorists and terrorist networks whose identities and whereabouts we do not always know.

What is the nature of the terrorist threat? Historically, terrorism has been a weapon of the weak characterized by the systematic use of actual or threatened physical violence, in pursuit of political objectives, against innocent civilians. Terrorist motives are to create a general climate of fear to coerce governments and the broader citizenry into ceding to the terrorist group's political objectives [7]. Terrorism today is transnational in scope, reach, and presence, and this is perhaps its greatest source of power. Terrorist acts are planned and perpetrated by collections of loosely organized people operating in shadowy networks that are difficult to define and identify. They move freely throughout the world, hide when necessary, and exploit safe harbors proffered by rogue entities. They find unpunished and oftentimes unidentifiable sponsorship and support, operate in small independent cells, strike infrequently, and utilize weapons of mass effect and the media's response in an attempt to influence governments [1].

There are numerous challenges to counterterrorism today. As we noted earlier, identifying terrorists and terrorist cells whose identities and whereabouts we do not always know is difficult. Equally difficult is detecting and preempting terrorists engaged in adverse actions and plots against the U.S. Terrorism is considered a low-intensity/low-density form of warfare; however, terrorist plots and activities will leave an information signature, albeit not one that is easily detected. In all cases, and as certainly has been widely reported about the Sept. 11 plot, terrorists have left detectable clues—the significance of which, however, is generally not understood until after an attack. The goal is to empower analysts with tools to detect and understand these clues long before an attack is scheduled to occur, so appropriate measures can be taken by decision- and policymakers to preempt such attacks.

### Key Information Technologies for Counterterrorism

**T**he ballistic missiles and satellite surveillance systems that were considered so effective at ending the Cold War are not sufficient to counter this new threat. There are many technology challenges, but perhaps few more important than how to make sense of and connect the relatively few and

	Cold War 1950–1990	Transition 1991–2003	Global futures 2004 and Beyond (Notional)
Global Situation	Bipolar world →	World in transition →	Increasing global cacophony???
Security Policy Environment	* Communist threat * Nation states rely on treaty based internationalism to maintain peace (NATO, UN, World Bank, and IMF)	* Israel-Palestine * Regional conflicts due to collapse of Soviet Union (the Balkans) * WMD proliferation * Terrorism (conventional IEDs)	* North versus South * Israel-Palestine and the Middle East * Continued regional conflicts * Continued WMD proliferation * Terrorism (conventional, WMD, and cyber)
National Security Strategy	* Nuclear deterrence * Contain spread of Communism * Pursue superior technology	* Nuclear disarmament * Two wars * Contain spread of WMD * Precision conventional weapons * Information dominance	* Homeland defense * Preemptive force * Flexible coalitions * Ultra-precise conventional weapons * Transformation of intelligence
Intelligence Strategy	Collection oriented: Penetrate and observe physical objects in denied areas	Production oriented: Analyze, produce, and disseminate	Knowledge oriented: Co-creation, learning
Key Intelligence Technologies	Advanced technology sensors (IMINT, SIGINT, and MASINT)	Information technology (databases, networks, and applications)	Cognitive technology: methodology, models, epistemology. Collaborative technology: center-edge integration

Courtesy of General Dynamics Advanced Information Systems, used with permission.

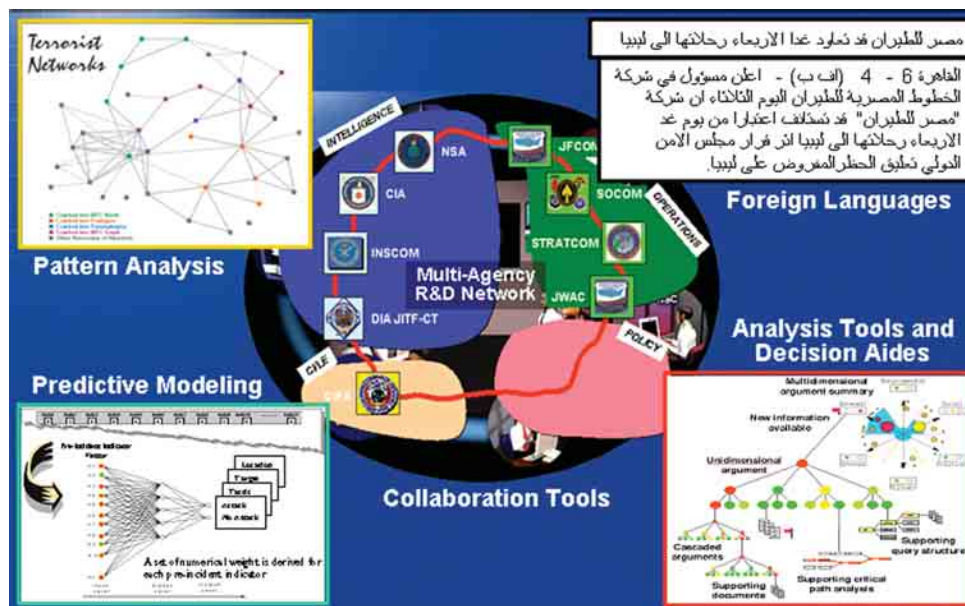
**Table 1. The world has changed dramatically since the Cold War era.**

sparse dots embedded within massive amounts of information flowing into the government's intelligence and counterterrorism apparatus. As noted in [7, 9], IT plays a crucial role in overcoming this challenge and is a major tenet of the U.S. national and homeland security strategies. The U.S. government's intelligence and counterterrorism agencies are responsible for absorbing this massive amount of information, processing and analyzing it, converting it to actionable intelligence, and disseminating it, as appropriate, in a timely manner. It is vital that the U.S. enhance its Cold War capabilities by exploiting its superiority in IT by creating vastly improved tools to find, translate, link, evaluate, share, analyze, and

act on the right information in as timely a manner as possible.

Figure 1 identifies some of the core IT areas we consider crucial for counterterrorism, namely: collaboration, analysis and decision support tools; foreign language tools; pattern analysis tools; and predictive (or anticipatory) modeling tools. We focus on only the first three here, recognizing,

**Figure 1. Critical information technology thrust areas for counterterrorism.**



Information Technology	Description
<b>Biometrics</b>	Identify and/or verify human terrorist (or watchlist) subjects using 2D and 3D modeling approaches over a variety of biometric signatures: face, gait, iris, fingerprint, voice. Also exploit multiple sensor modalities: EO, IR, radar, hyper-spectral.
<b>Categorization, Clustering</b>	Employ numerous technical approaches (natural language processing, AI, machine learning, pattern recognition, statistical analysis, probabilistic techniques) to automatically extract meaning and key concepts from (un)structured data and categorize via an information model (taxonomy, ontology). Cluster documents with similar contents.
<b>Database Processing</b>	Ensure platform, syntactic and semantic consistency and interoperability of multiple types of data stored on multiple storage media (disk, optical, tape) and across multiple database management systems. Desirable aspects include flexible middleware for: data location transparency and uncertainty management, linguistically relevant querying tuned for knowledge discovery and monitoring, scalability and mediation, schema evolution and metadata management, and structuring unstructured data.
<b>Event Detection and Notification</b>	Monitor simple and complex events and notify users (or applications) in real time of their detection. Monitoring can be scheduled a priori, or placed on an ad hoc basis driven by user demands. When an event is detected, automatic notifications can range from simple actions (sending an alert, page, or email) to more complex ones (feeding information into an analytics system).
<b>Geospatial Information Exploitation</b>	Fuse, overlay, register, search, analyze, annotate, and visualize high-resolution satellite and aerial imagery, elevation data, GPS coordinates, maps, demographics, land masses, political boundaries to deliver a streaming 3D map of the entire globe.
<b>Information Management and Filtering</b>	Collect, ingest, index, store, retrieve, extract, integrate, analyze, aggregate, display, and distribute semantically enhanced information from a wide variety of sources. Allow for simultaneous search of any number of information sources, sorting and categorizing various items of information according to query relevance. Provide an overall view of the different topics related to the request, along with the ability to visualize the semantic links relating the various items of information to each other.
<b>Infrastructure</b>	Provide comprehensive infrastructure for capturing, managing, and transferring knowledge and business processes that link enterprise software packages, legacy systems, databases, workflows, and Web services, both within and across enterprises. Important technologies include Web services, service-oriented grid-computing concepts, extensible component-based modules, P2P techniques, and platforms ranging from enterprise servers to wireless PDAs, Java, and Microsoft, .NET implementations.
<b>Knowledge Management, Context Development</b>	Use Semantic Web, associative memory, and related technologies to model and make explicit (expose via Web services) an analyst's personal preferences, intellectual capital, multidimensional knowledge, and tacit understanding of a problem domain.
<b>Predictive Modeling</b>	Predict future terrorist group behaviors, events, and attacks, based on past examples and by exploiting a variety of promising approaches, including neural networks, AI, and behavioral sciences techniques, subject matter expertise, and red teams.
<b>Publishing</b>	Generate concise accurate summaries of recent newsworthy items, ensuring users see topics only once, regardless how many times the item appears in data or in the press.
<b>Searching</b>	Allow users to perform more complete and meaningful searches (free text, semantic, similarity, partial or exact match) across a multitude of geographically dispersed, multilingual and diverse (un)structured information repositories within and across enterprises (any document type located on file servers, groupware systems, databases, document management systems, Web servers).
<b>Semantic Consistency, Resolving Terms</b>	Exploit ontologies, taxonomies, and definitions for words, phrases, and acronyms using a variety of schemes so users have a common and consistent understanding of the meaning of words in a specific context. Resolve semantic heterogeneity by capitalizing on Semantic Web technologies.
<b>Video Processing</b>	Analyze, detect, extract, and digitally enhance (reduce noise, improve image color and contrast, and increase resolution in selected areas) user-specified behaviors or activities in video (suspicious terrorist-related activities).
<b>Visualization</b>	Provide graphical displays, information landscapes, time-based charts, and built-in drill-down tools to help analysts and investigators discover, discern, and visualize networks of interrelated information (associations between words, concepts, people, places, or events) or visually expose non-obvious patterns, relationships, and anomalies from large data sets.
<b>Workflow Management</b>	Create optimized workflows and activities-based business process maps using techniques, such as intelligent AI engines by watching, learning, and recording/logging the activities of multiple users using multiple applications in multiple sessions.

**Table 2. Other information technologies considered important for counterterrorism.**

results recently obtained through experiments via partnerships that DARPA conducted with several entities within the U.S. intelligence and counterterrorism communities.

The purpose of the experiments was for analysts to assess the merits of several IT tools developed and integrated under DARPA sponsorship applied to various foreign intelligence problems. The experiments involved real analysts solving real foreign intelligence problems using their own lawfully collected foreign intelligence data. The tools provided by DARPA spanned the three core IT areas: peer-to-peer collaboration

however, there are numerous other information technologies that are equally important. Table 2 identifies and describes some of these core areas.

Figure 2 shows how the three core IT areas map onto a typical intelligence analysis process. These technologies will allow users to: search, query, and exploit vastly more foreign speech and text than would otherwise be possible by human translators alone; automatically extract entities and relationships from massive amounts of unstructured data and discover terrorist-related relationships and patterns of activities among those entities; and collaborate, reason, and share information, so analysts can hypothesize, test, and propose theories and mitigating strategies about possible futures, and to enable decision- and policymakers to effectively evaluate the impact of current or future policies and prospective courses of action.

We discuss each of these areas in more detail later in this article. Before doing so, however, we first underscore their critical importance by describing some promising

tools, structured argumentation and analytical tools, foreign language tools for audio searching/indexing and text and audio filtering/categorization, and statistical graph-based pattern analysis tools.

As Figure 3 shows, when doing traditional intelligence analysis, an analyst spends the most time on the major processes broadly defined as research, analysis, and production. The pink “bathtub curve” represents the distribution of time one typically sees.<sup>2</sup> This shows that analysts spend too much time doing research (searching, harvesting, reading, and preprocessing data for analysis), too much time doing production (turning analytical results into reports and briefings for the decision maker), and too little time doing analysis (thinking about the problem). The objective of the experiment was to see if intelligence analysis could be improved through

<sup>2</sup>The three numeric values associated with the pink curve are based on a real-world problem that will be described in more detail here. To interpret the curve, each value denotes the percentage of time—out of 100% total—an analyst spends on research, analysis, or production.



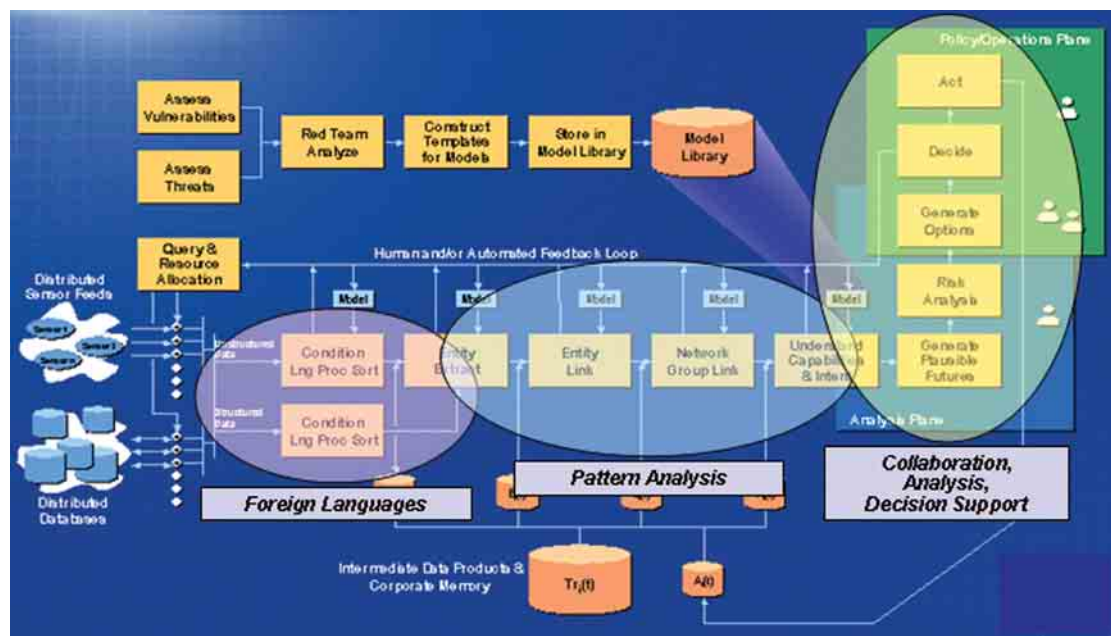


Figure 2. Three core information technology thrust areas mapped onto a typical intelligence analysis process.

IT by reversing this trend and inverting the bathtub curve.

In this experiment, the intelligence question the analysts were asked to analyze was “What is the threat posed by Al Qaeda’s Weapons of Mass Destruction capabilities to several cities in the U.S.?” The data was drawn from a variety of classified intelligence sources, foreign news reports, and Associated Press wire service reports.

The results of the experiment were impressive. As the yellow curve in Figure 3 shows, an inverted bathtub curve, allowing for more and better analysis in a shorter period of time, resulted when analysts used IT to aid their analysis. Results included an impressive savings in analyst labor (half as many analysts were used for the analysis), and five reports were produced in the time it ordinarily took to produce one. Moreover, the time spent in the research phase was dramatically reduced due mainly to using collaboration and foreign language tools to share and preprocess the foreign news and AP wire service data in 76 hours versus the 330 hours it took previously using more traditional manually driven methods.

## Collaboration, Analysis, and Decision Support

**C**ollaboration, analysis, and decision support tools allow humans and machines to analyze (think) and solve complicated and complex problems together more efficiently and effectively. These tools are what transform the massive amounts of data flowing into the government’s intelligence and counterterrorism communities into intelligence. Specifi-

cally, tools are needed to address each element of the “cognitive hierarchy,” namely, tools to transform data (discriminations between states of the world) into information (dots, or evidence, which is data put into context), and information into knowledge (useful and actionable information to decision makers).

*Enable center-edge collaboration.* Combating the terrorist threat requires all elements of the government to share information and coordinate operations. No one organization now has nor will ever have all the needed information or responsibility for counterterrorism. In addition to breaking down organizational barriers and sharing data, collaboration is also about sharing the thinking processes. Sharing of the thinking processes is about multiple perspectives and conflictive argument, and embracing paradox—all which enable humans to find the right perspective lenses in which to properly understand the contextual complexity through which correct meaning is conveyed to data. Collaboration tools permit the formation of high-performance agile teams from a wide spectrum of organizations. These tools must support both top-down, hierarchically organized and directed, “center-based” teams, as well as bottom-up, self-organized and directed ad-hocracies—“edge-based” collaboration. These two modes of operation must also be able to interoperate: “center-edge” coexistence.

*Manage policies.* The U.S. is a nation of laws, and all activities of government are conducted within the bounds of existing laws, policies, and regulations. But the policies and regulations vary tremendously across the variety of organizations that must collaborate to counter today’s threats. Tools are needed to

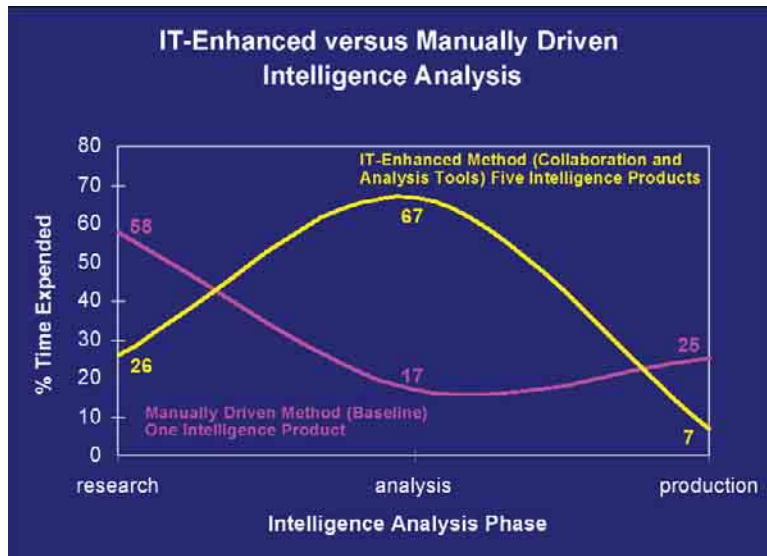


allow policy and regulation to be unambiguously defined and understood at all levels, to permit teams to reconcile their differing policies and regulations into a single coherent regime, to consistently and reliably apply that regime to its operations, and to identify any deviations from policy and regulation to prevent abuses.

*Supporting process.* Teams, especially so-called ad-hocracies, need support in designing and executing strategies embodied in procedures to accomplish their goals. Tools are needed to allow them to develop and execute appropriate processes and procedures throughout their life cycles, and to ensure these processes and procedures are consistent with applicable policy.

*Amplify human intellect.* To effectively deal with the terrorist threat, it is not sufficient for well-informed experts to simply be able to communicate and share information. The counterterrorism problem is an intrinsically difficult one that is only compounded by unaided human intellect. Humans as individuals and in teams are beset by cognitive

**Figure 3. Early results are promising, showing more and better analysis in a shorter period of time by way of collaboration, modeling, and analysis tools.**



biases and limitations that have been partially responsible for some intelligence failures [3]. Analysts must be given assistance in the form of structured analytic approaches and methodologies to amplify their cognitive abilities and allow them to think together [11]. Additionally, rich toolsets are needed to allow users to understand the present, imagine the future, and generate actionable options for the decision maker.

*Reinvent policy/intelligence interface.* As U.S. Secretary of Defense Donald Rumsfeld has indicated, pol-

icymakers must not be simply passive consumers of intelligence. Instead, senior policymakers must “engage analysts, question their assumptions and methods, seek from them what they know, what they don’t know, and ask them their opinions [10].” Also, because the current policy/intelligence interface model was developed in the Cold War era during a time of information scarcity (unlike today’s information-abundant environment), some of the basic assumptions that underlie it are no longer optimal. Novel technology is needed to reinvent the interface between the worlds of policy and intelligence, allowing for intelligence that is, for example: aggressive, not necessarily cautious; intuitive, not simply fact-based; metaphor-rich, as opposed to concrete; collaborative, in addition to hierarchical; precedent-shattering, not precedent-based; and opportunistic, as well as warning-based.

*Explanation generation.* It is not enough to simply connect the dots. The fact that the dots are connected must be persuasively explained and communicated to decision- and policymakers. Traditional methods, such as briefings and reports, lack on both counts and also demand a significant amount of analysts’ time to produce (recall the bathtub curves in Figure 3). Explanation-generation technology is critical to producing traditional products, as well as making possible newer forms of intelligence products.

## Foreign Languages

**F**oreign language speech and text are indispensable sources of intelligence, but the vast majority is unexamined: Volumes are huge and growing; processing is labor intensive; and the U.S. intelligence and counterterrorism communities have too few people with suitable language skills. Because it would be impossible to find, train, or pay enough people, creating effective foreign language technology is the only feasible solution. New

and powerful foreign language technology is needed to allow English-speaking analysts to exploit and understand vastly more foreign speech and text than is possible today.

*Transcription.* Automatic transcription technology is needed to produce rich, readable transcripts of foreign news broadcasts—despite widely varying pronunciations, speaking styles, and subject matter. The two basic components of transcription are speech-to-text conversion (finding the words) and metadata extraction (pulling out more information). Inter-

Arabic	Human Translation	Machine Translation
مصر للطيران قد تعاود عدا الإرباعا رحلاتها الى ليبيا اعلان - (اف ب) 4 - القاهرة مستوفين في شركة الخطوط للمصريه مصر " للطيران اليوم الثلاثاء ان شركة قد سئلف اعتبارا من يوم غد " للطيرون الاربعاء رحلاتها الى ليبيا لتر فرار مجنن الامن لادولي تطبق المظروفروض على ليبيا	<b>Egypt Air May Resume its Flights to Libya Tomorrow</b> Cairo, April 6 (AFP) - An Egypt Air official announced, on Tuesday, that Egypt Air will resume its flights to Libya as of tomorrow, Wednesday, after the UN Security Council had announced the suspension of the embargo imposed on Libya.	<b>Egyptair Has Tomorrow to Resume Its Flights to Libya</b> Cairo 4-6 (AFP) - said an official at the Egyptian Aviation Company today that the company Egyptair may resume as of tomorrow, Wednesday its flights to Libya after the International Security Council resolution to the suspension of the embargo imposed on Libya.

**Table 3. Recent machine translation results of Arabic news text show great promise.**

ested readers can find more information on basic speech-to-text technology in [12]. Recent achievements include word error rates of 26.3% and 19.1% at processing speeds seven and eight times slower than real-time rates on Arabic and Chinese news broadcasts. The goal is 10% or less at real-time rates.

*Translation.* A key finding in [9] was that the intelligence community was not adequately prepared to handle the challenge it faced in translating the multitude of foreign language intelligence data it collected. The challenges to contend with are numerous, including massive amounts of foreign text from an ever-growing number of foreign data sources, large unconstrained vocabularies, and numerous domains and languages with limited linguistic resources. Although the problem is far from solved, researchers are making considerable progress on the automatic translation of text. Table 3 shows the promising results obtained recently in translating an Arabic news article.

*Detection.* Advanced techniques to detect and discover the exact information a user seeks quickly and effectively and to flag new information that may be of interest are needed. Cross-language information retrieval is the current focus of the research community, with recent results showing it works approximately as well as monolingual retrieval.

*Extraction.* More sophisticated ways to extract key facts from documents are needed. Although name translation remains problematic, automatic name extraction (or tagging) works reasonably well in English, Chinese, and Arabic. Researchers are now focusing on sophisticated techniques for extracting information about entities, relationships, and events.

*Summarization.* Substantially reducing the amount of text that people must read in order to perform analysis is absolutely critical. Researchers are now working on techniques for automatic headline generation (for single documents) and for multi-document summaries (of clusters of related documents).

*Language independence.* Researchers are pursuing a wide variety of approaches that are substantially

language-independent and empirically driven. Algorithms are exploiting the continuing advances in computational power plus the large quantities of electronic speech and text now available. The ultimate goal is to create rapid, robust technology that can be ported cheaply and easily to other languages and domains.

## Pattern Analysis

Many terrorist activities consist of illegitimate combinations of otherwise legitimate activities. For example, acquisition of explosives, selection of a location, and financing of the acquisition by external parties are all legitimate activities in some contexts, but when combined or when performed by individuals known to be associated with terrorist groups or when the location is not, for example, a demolition/construction site but a landmark or other public building, suggest that further investigation may be warranted. While examples of terrorist activities are rare, examples of the component activities are not. Pattern analysis tools, therefore, must be able to detect instances of the component activities involving already suspicious people, places, or things and then determine if the other components are present to separate situations warranting further investigation from the majority that do not. Comprehensive overviews of some of the key technologies are available in [2, 4].

*Graphical representations.* One key idea that enables connecting the dots is representing both data and patterns as graphs. Patterns specified as graphs with nodes representing entities, such as people, places, things, and events; edges representing meaningful relationships between entities; and attribute labels amplifying the entities and their connecting links are matched to data represented in the same graphical form. These highly connected evidence and pattern graphs also play a crucial role in constraining the combinatorics of the iterative graph processing algorithms, such as directed search, matching, and hypothesis evaluation.

*Relationship extraction.* The initial evidence graph is comprised of entities and their relationships extracted from textual narratives about suspicious activities, materials, organizations, or people. Advanced techniques are needed to efficiently and accurately discover, extract, and link sparse evidence contained in large amounts of unclassified and clas-

sified data sources, such as public news broadcasts or classified intelligence reports.

**Link discovery.** Starting from known or suspected suspicious entities, patterns are used to guide a search through the evidence graph. Patterns can be obtained from intelligence analysts, subject matter experts, and intelligence or law enforcement tips, and are subject to extensive verification and testing before use. Statistical, knowledge-based, and graph-theoretic techniques are used to infer implicit links and to evaluate their significance. Search is constrained by expanding and evaluating partial matches from known starting points, rather than the alternative of considering all possible combinations. The high probability that linked entities will have similar class labels (often called autocorrelation or homophily) can be used to increase classification accuracy.

**Pattern learning.** Pattern learning techniques can induce a pattern description from a set of exemplars. Such pattern descriptions can assist an analyst to discover unknown terrorist activities in data. These patterns can then be evaluated and refined before being considered for use to detect potential terrorist activity. Pattern learning techniques are also useful to enable adaptation to changes in terrorist behavior over time.

## Conclusion

The results shown in Figure 3, based on the three core IT areas discussed in this article, represent the tip of the iceberg. Many other information technologies are important for successfully conducting the global war on terror (see Table 2). Experiments, such as the one described here, will help validate the merits and utility of these tools. Ultimately, such tools will create a seamless environment where analysts and decision- and policymakers can come together to collaborate, translate, find, link, evaluate,

share, analyze, and act on the right information faster than ever before to detect and prevent terrorist attacks against the U.S. **C**

## REFERENCES

1. Benjamin, D. and Simon, S. *The Age of Sacred Terror*. Random House, New York, 2002.
2. Goldszmidt, M. and Jensen, D. EELD recommendations report. In *Proceedings of the DARPA Workshop on Knowledge Discovery, Data Mining, and Machine Learning (KDD-ML)*, Arlington, VA, 1998.
3. Heuer, R. *Psychology of Intelligence Analysis*. Center for the Study of Intelligence, Central Intelligence Agency, 1999.
4. Jensen, D. and Goldberg, H., Eds. *Artificial Intelligence and Link Analysis: Papers from the 1998 AAAI Fall Symposium*, AAAI Press, Menlo Park, CA, 1998.
5. Jonietz, E. Total information overload. *MIT Technology Review* (Aug. 2003).
6. Lazaroff, M. and Sickels, S. *Human Augmentation of Reasoning Through Patterning (HARP)*. DARPA Genoa II PI Meeting, Austin, TX, May 2003.
7. *National Strategy for Combating Terrorism*. Submitted by the White House, Feb. 2003.
8. *Report to Congress Regarding the Terrorism Information Awareness (TIA) Program*. Submitted by the Secretary of Defense, Director of Central Intelligence and Attorney General, May 2003.
9. *Report of the Joint Inquiry into the Terrorist Attacks of September 11, 2001*. Submitted by the House Permanent Select Committee on Intelligence (HPSCI) and the Senate Select Committee on Intelligence (SSCI), July 2003.
10. Shanker, T. For military intelligence, a new favorite commando. *New York Times* (Apr. 11, 2003).
11. Schum, D. *The Evidential Foundations of Probabilistic Reasoning*. Wiley, New York, 1994.
12. Young, S. *Large Vocabulary Continuous Speech Recognition: A Review*. Technical Report, Cambridge University Engineering Department, Cambridge, U.K., 1996.


**ROBERT POPP** (rpopp@darpa.mil) is a special assistant to the DARPA Director for Strategic Matters and was formerly the deputy director of the Information Awareness Office.

**THOMAS ARMOUR** (tarmour@darpa.mil) is a program manager with DARPA's Information Processing Technology Office and was formerly a program manager with the Information Awareness Office.

**TED SENATOR** (tsenator@darpa.mil) is a program manager with DARPA's Information Processing Technology Office and was formerly a program manager with the Information Awareness Office.

**KRISTEN NUMRYCH** (knumrych@darpa.mil) is an assistant director for Program Management at DARPA.

© 2004 ACM 0002-0782/04/0300 \$5.00



**CONNECTING COMMUNITIES**  
UPA: Network in Our Community

**Marriott City Center Minneapolis, Minnesota**  
**Workshops & Tutorials June 7-8, 2004**  
**Presentations & Panels June 9-11, 2004**  
**UPA 2004 Conference**

**CONFERENCE REGISTRATION:**  
Register on-line at [www.usabilityprofessionals.org](http://www.usabilityprofessionals.org),  
or download a fax/mail registration form.

**MINNEAPOLIS MARRIOTT CITY CENTER**  
Room rates for UPA 2004 are \$119 per room per  
night, plus tax. In order to receive the group rate,  
room reservations must be made by **May 14,**  
**2004.** For reservations, call 612/349-4000.

**Keynote Speaker:**  
**Janice (Ginny) Redish, PhD**

**Closing Plenary Speaker:**  
**Howard Rheingold**

**By Thayne Coffman,  
Seth Greenblatt, and  
Sherry Marcus**

# GRAPH-BASED TECHNOLOGIES FOR INTELLIGENCE ANALYSIS

**W**hen intelligence analysts are required to understand a complex uncertain situation, one of the techniques they use most often is to simply draw a diagram of the situation. Natural language processing has matured to the point where the conversion of freeform text reports to these diagrams can be largely automated. The diagrams are attributed relational graphs (ARGs), an extension of the abstract directed graph from mathematics. In these ARGs, nodes represent people, organizations, objects, or events. Edges represent relationships like interaction, ownership, or trust. Attributes store the details of each node and edge, like a person's name or an interaction's time of occurrence. ARGs function as external memory aids, which are crucial tools for arriving at unbiased conclusions in the face of uncertain information [4].

The intelligence community's focus over the past 20 years on improving intelligence collection has come at the cost of improving intelligence analysis [4]. The problem today is often not a lack of information, but instead, information overload. Analysts lack tools to locate the relatively few bits of relevant information and tools to support reasoning over that information. Graph-based algorithms help security analysts solve the first problem—sifting through a large amount of data to find the small subset that is indicative of threatening activity. This activity is often suspicious not because of the characteristics of a single actor, but because of the dynamics between a group of actors. In contrast with databases and spreadsheets, which

*Enhancing traditional  
algorithmic techniques  
for improved pattern  
analysis.*



tend to facilitate reasoning over the characteristics of individual actors, graph representations facilitate reasoning over the relationships between actors. Subgraph isomorphism and social network analysis are two important graph-based approaches that will help analysts detect suspicious activity in large volumes of data.

Subgraph isomorphism algorithms search through large graphs to find regions that are instances of a specific pattern graph [8]. The analyst defines, as a graph, activity patterns that are believed to be indicative of

details, or the analyst may have defined some aspects of the pattern incorrectly. Finding inexact matches also alerts the analyst to activity that “breaks the mold” of previous threats, and can prevent the kind of surprises for which intelligence agencies have been criticized.

In our work, we have developed a set of genetic algorithms that solve the exact and inexact subgraph isomorphism problem. The search algorithm distributes nicely, allowing it to run on a server farm for improved performance. This enables efficient searches for patterns containing as many as

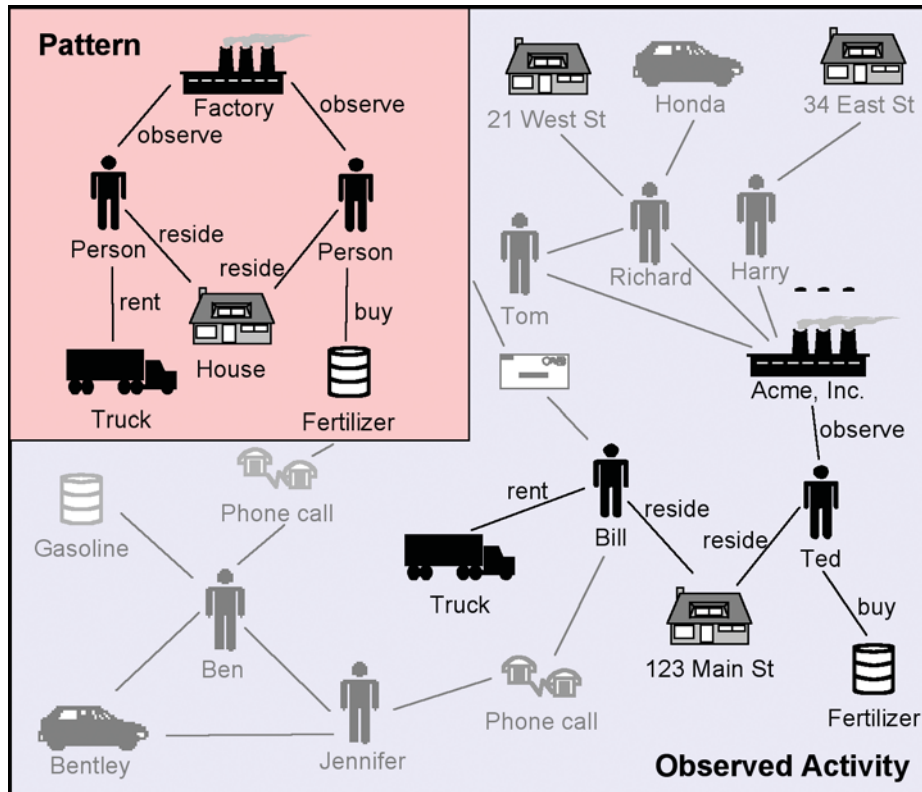
75 nodes and  $10^7$  different possible realizations.

Social network analysis (SNA) is the study of human social interaction. Graph representations are ubiquitous throughout SNA—sequences of interactions between people are usually represented as an ARG. SNA metrics quantify different aspects of the ARG’s topology, and the metric values can be used to characterize the roles of individuals within a group, or the state of a group or organization as a whole. The key opportunity for intelligence analysis is that “normal” social interaction and the social interaction of illicit groups tend to exhibit significantly different SNA metric values.

The geodesic assumption

and the redundancy assumption state that for human interaction, “People with strong relationships usually communicate via the shortest path” and “Normal social networks are redundant.” Studies have shown that both of these assumptions are typically false for groups trying to hide their activities [1]. For those groups, information compartmentalization and robustness to the compromise of group members overrule the efficiency concerns that otherwise lead to the geodesic and redundancy assumptions. The resulting differences in the groups’ structures can be quantified by SNA metrics. The SNA theory of homophily argues that most human communication occurs between people similar to each other. Thus people pursuing illicit activities are likely to be found communicating with others pursuing illicit activities. This “relational autocorrelation” further drives these groups’ SNA metrics toward anomalous values [5].

Our work combines SNA metrics with statistical



**Match to a pattern graph in an activity graph.**

threatening activity. Once those patterns are defined, the algorithm identifies regions of observed activity that match the patterns. One possible pattern is shown in the figure here, along with an inexact match to that pattern embedded in an activity graph containing both threatening and innocuous activity. Supporting evidence may come from many different sources, but once evidence is incorporated into the global activity graph, the algorithm lets the analyst quickly pinpoint subsets of activity that warrant further attention. Without advanced search algorithms like these, the analyst’s task of identifying suspicious activity within a huge body of evidence is much more difficult.

Being able to find inexact pattern matches is critical. Foremost, analysts operate in an environment with limited observability. In addition, the analyst might need to match a general pattern without knowing all of the



pattern classification to give the analyst a tool for automatically pinpointing suspicious group dynamics within large volumes of data [2]. This combination will yield algorithms that constantly scan incoming information, alerting the analyst to anomalous social patterns that might indicate threatening activity. Some illicit groups (for example, terrorist cells moving from a “sleeper” to an “active” state) finish in a relatively normal configuration, but the history of how they arrived there is highly abnormal [6]. Therefore, it is also important to develop techniques to classify activity based on the evolution of a group’s SNA metric values [3]. Because analysts have imperfect visibility into these interactions, we must also consider the sensitivity of various SNA metrics to limited observability [7].

Subgraph isomorphism and statistical classification via SNA metrics are two important classes of techniques that operate on attributed relational graphs, a representation familiar to the intelligence analyst. These two techniques help the analyst solve one of today’s most common intelligence problems: finding significant combinations of events in a deluge of information. **C**

## REFERENCES

1. Baker, W.E. and Faulkner, R.R. The social organization of conspiracy: Illegal networks in the heavy electrical equipment industry. *American Sociological Review* 58, 6 (Dec. 1993), 837–860.
2. Coffman, T. and Marcus, S. Pattern classification in social network analysis: A case study. In *Proceedings of the 2004 IEEE Aerospace Conference* (Big Sky, MT, Mar. 2004).
3. Coffman, T. and Marcus, S. Dynamic classification of groups using social network analysis and HMMs. In *Proceedings of the 2004 IEEE Aerospace Conference* (Big Sky, MT, Mar. 2004).
4. Heuer, R.J. *Psychology of Intelligence Analysis*. Center for the Study of Intelligence, Central Intelligence Agency, 2001.
5. Jensen, D., Rattigan, M., and Blau, H. Information awareness: A prospective technical assessment. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2003.
6. Krebs, V. Mapping networks of terrorist cells. *Connections* 24, 3 (Winter 2001), 43–52.
7. Thomason, B., Coffman, T., and Marcus, S. Sensitivity of social network analysis metrics to observation noise. In *Proceedings of the 2004 IEEE Aerospace Conference* (Big Sky, MT, Mar. 2004).
8. Ullman, J.R. An algorithm for subgraph isomorphism. *Journal of the ACM* 23, 1 (Jan. 1976), 31–42.

---

**THAYNE COFFMAN** (tcoffman@21technologies.com) is a principal scientist at 21st Century Technologies in Austin, TX.

**SETH GREENBLATT** (sgreenblatt@21technologies.com) is the chief scientist at 21st Century Technologies in Austin, TX.

**SHERRY MARCUS** (sem@21technologies.com) is the president and founder of 21st Century Technologies in Austin, TX.

---

This research was sponsored by the Defense Advanced Research Projects Agency (DARPA), under the EELD and Genoa 2 programs. The views and conclusions contained in this article are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of DARPA or the U.S. government.

---

© 2004 ACM 0002-0782/04/0300 \$5.00

---

**By Paul Kogut,  
John Yen, Yui Leung,  
Shuang Sun, Rui Wang,  
Ted Mielczarek, and  
Ben Hellar**

# PROACTIVE INFORMATION GA HOMELAND SECURITY TEAMS

**I**magine a software assistant agent (AA) that proactively provides relevant concise personalized information to an analyst or first responder and collaborates with other AAs to share knowledge and arrange person-to-person contact. The following descriptive scenario provides an example application of this concept. A report of an explosion at a chemical plant is received by a homeland security (HLS) analyst, local firefighters, and state police. The AA for the analyst “reads” the report and immediately retrieves information about what chemicals exist in the plant and a map of nearby dangerous and vulnerable facilities.

The firefighters arrive on the scene and identify the sector from which flames and smoke are emanating. An AA provides the firefighters with information on what chemicals are likely to be burning, how to extinguish the fire, and the potential health hazards of the smoke. The AA notifies the HLS analyst that it is a harmless gas. Plant personnel tell state police they saw a suspicious individual in a car in the parking lot. The police enter the license plate number of the suspicious car and the analyst AA immediately searches for aliases of the owner and links to terrorist organizations. A link is found and the HLS analyst’s AA searches for an AA of an expert on that terrorist group. The terrorist expert AA notifies the HLS AA that an associate of the suspicious person is a chemical engineer that works in a nearby plant where another explosion has just been reported. The HLS AA discovers that if the two smoke plumes intersect they will create a deadly acid mist. The AA plots the smoke plumes on a map and notifies the HLS analyst that the combined plume will reach a crowded sports stadium in approximately 15 minutes. The AA immediately initiates a phone call between the HLS analyst and stadium security.

This scenario illustrates how a team of software agents could support a team of people to accomplish a time-critical task related to U.S. homeland security activities. One of the major technical issues in designing software agents with these capabilities is they need to be

# ATHERING FOR

*Supporting counterterrorism analysts with software agents that dynamically anticipate their information requirements.*

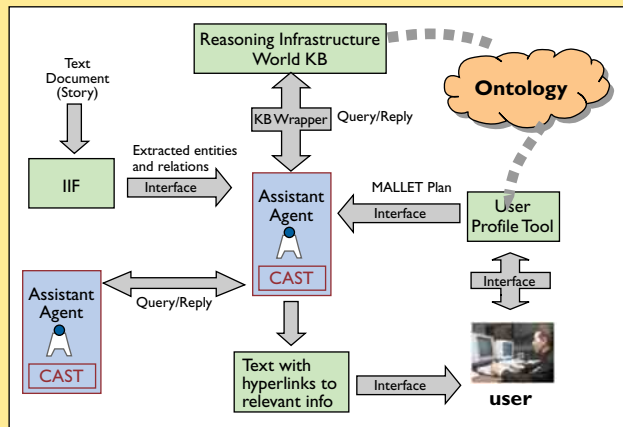
able to anticipate and reason about information needs of teammates in a highly dynamic environment.

## Realization of the Vision

Lockheed Martin and Pennsylvania State University's School of Information Sciences and Technology are collaborating on a project to realize this vision using CAST (Collaborative Agents for Simulating Teamwork) agent technology [4–6]. There are several similar agent research efforts: CoABs (see [coabs.globalinfotek.com/](http://coabs.globalinfotek.com/)) and STEAM [3] are two examples.

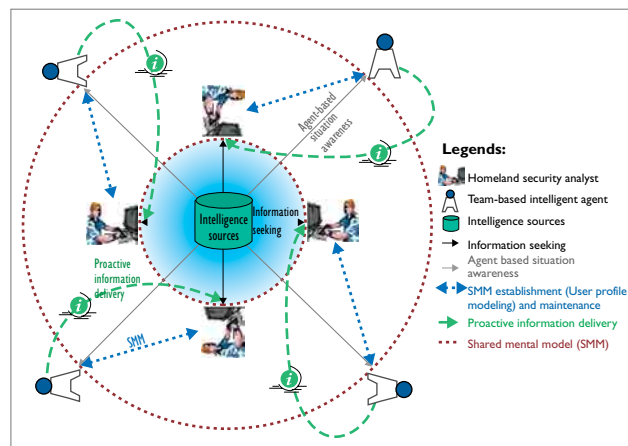
One of the major features distinguishing CAST from these approaches is that CAST enables an agent to dynamically infer information needs of its teammates from a shared mental model about the structure and the process of the team, which is expressed in the knowledge representation language MALLT (Multi-Agent Logic-based Language for Encoding Teamwork).

Figure 1 shows the overall architecture of using CAST agents to assist homeland security teams. The user guides the creation of the shared mental model (that is, MALLET knowledge of CAST agents) by selecting components from an ontology that pertain to the user's role in the team. Incoming messages and documents are pre-processed by the M&DS Intelligent Information Factory (see [mds.external.lmco.com/mds/products/gims/iif/index.html](http://mds.external.lmco.com/mds/products/gims/iif/index.html)) to extract important entities, relations, and events. Based on the shared mental model captured in MALLET, the extraction results trigger appropriate information gathering actions by the CAST agent. The CAST agent then interacts with the reasoning



**Figure 1. Architecture for proactive information gathering.**

infrastructure/world knowledge base (RIWKB) or other CAST agents to retrieve and present relevant concise personalized information to the user. Heterogeneous knowledge from various sources is loaded into the RIWKB by techniques such as Web scraping, information extraction, and importing Semantic Web



**Figure 2. Shared mental models.** markup [2]. The output to the user is the original text with superimposed hyperlinks to fine-grained information (with drill-down links to original source documents).

## Agents for Proactive Information Exchange


**A**s mentioned earlier, CAST is an agent architecture that empowers agents in a team to have a shared mental model about the structure and the process of the team so they can anticipate potential information needs of teammates and proactively deliver it to them. Psychologists who studied teamwork have identified overlapping shared mental models as an important characteristic of high-performance teams [1].

To enable agents with such capabilities, CAST provides four key features. First, it uses a high-level language, MALLET, to capture knowledge about the team structure and the team process. Second, each agent establishes a computational shared mental model by transforming its teamwork knowledge in MALLET into a Prolog-like knowledge base and a predicate transition nets—a process representation that extends Petri Nets. Agents maintain their shared mental models about team states by dynamically updating the knowledge base and the predicate transition nets. Third, the CAST kernel provides domain-independent algorithms that enable agents to dynamically allocate responsibilities among members of the team, to infer information needs of teammates, and to proactively deliver relevant information to them. Finally, each agent uses a decision-theoretic communication strategy

for determining how it should assist teammates regarding their information needs.

The CAST agents play the role of AAs by establishing and maintaining shared mental models with the human analysts as shown in Figure 2. The shared mental models contain knowledge about the responsibilities, information-seeking processes, and information-processing methods of human users. The notion of shared mental model extends the concept of user profiles in two ways. Fundamentally, it broadens the scope of user profiles to include dynamic process, tasks, roles, and responsibilities about human users. Additionally, it extends the profile of an individual user to a team.

## Conclusion

CAST-enabled AAs will profoundly change the way HLS teams perform their missions. The architecture we have described here can be tailored to support applications in other domains such as teams of institutional and individual investors in the financial domain and enterprise knowledge management in large corporations with diverse technologies. The architecture could also be augmented with machine learning for automated adaptation of information needs. 

## REFERENCES

1. Cannon-Bowers, J.A., Salas, E., and Convers, S.A. Cognitive psychology and team training: Training shared mental models and complex systems. In *Human Factors Society Bulletin*, 1990.
2. Kogut, P. and Heflin, J. Semantic Web technologies for aerospace. In *Proceedings of IEEE Aerospace Conference*, 2003.
3. Tambe, M. Towards flexible teamwork. *Journal of Artificial Intelligence Research* 7 (1997), 83–124.
4. Yen, J., Fan, X., and Volz, R.A. On proactive delivery of needed information to teammates. In *Proceedings of the AAMAS 2002 Workshop of Teamwork and Coalition Formation*, 2002.
5. Yen, J. et al. CAST: Collaborative Agents for Simulating Teamwork. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI-01)*, 2001, 135–142.
6. Yin, J. et al. A knowledge-based approach for designing intelligent team training systems. In *Proceedings of the Fourth International Conference on Autonomous Agents*, 2000, 427–434.

**PAUL KOGUT** (paul.a.kogut@lmco.com) is a research project lead at Lockheed Martin Integrated Systems and Solutions in King of Prussia, PA and an adjunct professor of software engineering at Pennsylvania State University.

**JOHN YEN** (jyen@ist.psu.edu) is University Professor of Information Sciences and the professor in charge of the School of Information Sciences and Technology at Pennsylvania State University.

**YUI LEUNG** (yui.h.leung@lmco.com) is a software engineer at Lockheed Martin Integrated Systems and Solutions.

**SHUANG SUN** (ssun@ist.psu.edu) is a Ph.D. student of information sciences and technology at Pennsylvania State University.

**RUI WANG** (rzw104@psu.edu) is a Ph.D. student of information sciences and technology at Pennsylvania State University.

**TED MIELCZAREK** (ted.a.mielczarek@lmco.com) is a software engineer at Lockheed Martin Integrated Systems and Solutions.

**BEN HELLAR** (bhellar@ist.psu.edu), formerly a software engineer at Lockheed Martin Integrated Systems and Solutions, is a student of information sciences and technology at Pennsylvania State University.





BY O. SAMI SAYDJARI

# CYBER DEFENSE: ART TO SCIENCE

*Seeking the knowledge and means to more methodically detect, defend against, and better understand attacks on networked computer resources.*

Imagine that you lead an organization under cyber attack on your critical information systems. What questions are you likely to ask?

*Am I under attack; what is its nature and origin?  
What are the attackers doing; what might they do next?  
How does it affect my mission?  
What defenses do I have that will be effective against this attack?  
What can I do about it; what are my options?  
How do I choose the best option?  
How do I prevent such attacks in the future?*

Unfortunately, today we must often answer, “We don’t know and we have no way of knowing.” Informally, it is in being able to answer these basic questions that we find the meaning of the term cyber defense.

More formally, we can define cyber defense from its component words. Cyber, short for cyberspace, refers to both networked infrastructure (computers, routers, hubs, switches, and firewalls) and the information assets (critical data on which an organization depends to carry out its mission). Defense is the act of making safe from attack. Therefore, cyber defense refers to an active process of dependably making critical function safe from attack.

ILLUSTRATION BY  
ROBERT NEUBECKER

## Elements of Cyber Defense

Defense in cyberspace is as complex as traditional warfare—it has the same key elements, corresponding to the basic questions listed at the beginning of this article: sensors and exploitation; situation awareness; defensive mechanism; command and control; strategy and tactics; and science and engineering. Simply put, one needs the knowledge and means to defend oneself, detect and understand attacks, and make good decisions about defense configuration. Each of the six elements are discussed in more detail here.

**Cyber sensors and exploitation** are the “eyes” of the system; they determine the attack capability, plans, and actions of an adversary—the essential first step to any dynamic defense. A primitive form of determining adversary actions is what we today call intrusion detection. To succeed we must acknowledge that attacks will sometimes succeed and adversaries will get inside the system. To assume otherwise is foolish.

**Cyber situation awareness** is a process that transforms sensed data into a decision aid by interpreting mission consequences and the context of other activity. For example, situation awareness might tell us that attack A will disable the organization’s logistics function for three days and that the attack is pandemic and is thus not targeting our organization specifically.

**Cyber defensive mechanism** is technology to counter threats. Historically, cyber defense has its roots in this element, with cryptography countering intercepted secret messages, virus scanners countering viruses, and firewalls countering hacker exploitations. Although this element is an important building block, professionals must extend their understanding beyond the cyber defense mechanism to see the bigger picture.

**Cyber command and control** is the process of making and executing decisions—orchestrating defensive systems, based on input from the situation awareness element. Command decision making requires an understanding of options based on the situation, and the means to evaluate them quickly [12]. Control requires a system to communicate the decisions and execute them reliably throughout the system.

**Cyber strategies and tactics** is knowledge of what constitutes a good decision in terms of initial defensive policies and configurations as well as changes needed during operations because of attack situations. Ideally, such knowledge is based on a wealth of historical experiences, but we prefer not to sustain the damages required to gain real cyber battlefield experience. As a substitute, we must begin developing strategies and tactics and testing them experimentally

in models of real systems with mock adversaries.

**Cyber science and engineering** is the foundation yielding an understanding of design, composition, building, and maintenance of effective defense systems. Currently, this foundation is dangerously weak to the extent that it exists at all.

## Dynamic Defense Is Imperative

**S**tatic preventive techniques, while important, are inadequate. In the design of trustworthy cyber defense systems, there is a three-way trade-off among security, performance, and functionality. The security dimension itself has at least three components: confidentiality, data integrity, and availability. One cannot statically optimize all dimensions with respect to all attacks. For example, although spreading many copies of data around a system can hinder denial-of-service attacks, it exacerbates the confidentiality problem by creating more targets of opportunity for the attacker. At the higher level, security functions often degrade both performance and functionality. One would rather not have to incur these costs unless under attack, just as soldiers do not put on chemical suits unless there is a known threat of chemical attack on the battlefield.

We need to create systems that make explicit trade-offs within this space both at design time and at operation time—dynamically moving within the trade-off space depending on the situation. We also, therefore, need systems capable of quickly ascertaining the situation so the correct trade-offs can be made.

## The Art of War—Strategy and Tactics

**C**yber attacks are becoming sophisticated; attackers routinely use attack design toolkits, apply stealth techniques, and target an increasing spectrum of protocols and applications. Cyber attackers are learning to actively evade countermeasures. Soon they will develop sophisticated tactics and will evolve toward strategic campaigns using multi-pronged attacks against strategic objectives. Moreover, attackers have the advantage because they can carefully plan and choose the best time and the weakest points at which to attack. Creating defenses capable of thwarting such attacks will take years; we cannot afford to wait until we see cyber attack methods evolve to this level.

At the same time, defensive mechanisms are proliferating and becoming increasingly complex—exceeding our ability to understand how best to configure

each mechanism and the aggregate of mechanisms.

To effectively manage all the defensive elements, one needs strategy and tactics. Because we have little history in cyberspace, we must look to analogy. We can apply analogies from the battlefield [9]. For example, the battlefield concept of forcing an adversary into disadvantageous terrain has a cyberspace analogue of arranging one's defensive architecture to force adversaries into the "sweet spots" of their intrusion detection algorithms. The battlefield concept of deception has the cyberspace analogue of creating false cyber targets (also known as honey pots) and misleading configurations.

Similarly, one may borrow from the realm of strategic game playing. The "game" of war is extraordinarily complex because of the great variety of moves, changing rules, and changing capabilities. Yet, some general principles apply, especially as a human decision aid [3]. Determining the right strategic decisions is best performed by creative well-informed humans. This makes cyber defense a matter of art, supported by science, not a matter for total automation. Therefore, we should focus on automating the mundane tasks and providing decision aids to qualified humans for the strategic decision making.

To develop strategy and tactics we accumulate hypotheses based on analogy, and then validate them. We can gain experience through simulation on accurate models of our critical systems interacting with human decision makers. We must engage in many scientific experiments within these models. Adversaries must be accurately modeled using our best red teams. Our strategy and tactics—our cyber defense playbook—need to be validated in such simulations to yield the knowledge to defend our critical cyberspace from sophisticated attack. We must learn how to defend against how real attackers will attack.

Although there is much to be learned from physical war strategy and tactics, there are other areas where the differences are big enough to require a completely new way of thinking about strategy and tactics in cyberspace. As a word of caution, consider

some of the key differences. Physical space is three-dimensional; cyberspace is hyper-dimensional, making maneuvering complex. Physical weapon effects are predictable and constrained by physics; cyber weaponry is difficult to predict, often having non-linear damaging effects. Physical attacks occur at human-perceptible speeds; some cyber attacks may aggregate too slowly to be perceived, while many others could occur in milliseconds, making all of them outside the realm of possible human reaction times. Physical attacks often have clear manifestations; cyber attacks can be difficult to detect, making damage assessment problematic.

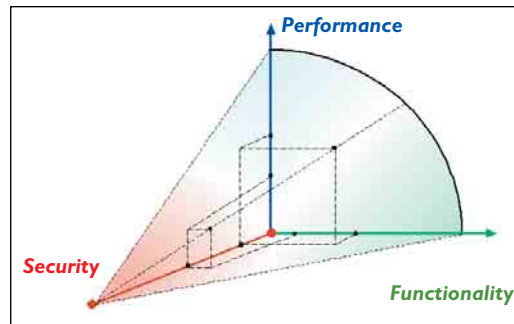


Figure 1. Dynamic design and operating trade-off space.

## Science and Technology Deficits

To achieve a viable cyber defense capability, we need many advances in both science and technology. Here are a few.

*We must learn how to create trustworthy systems from untrustworthy components* [1]. Trustworthy systems are the building blocks of good cyber defense.

The need to create them from untrustworthy components arises from two sources: the vulnerable computer systems that consumers habitually choose and basic system engineering limitations. Trustworthiness, like reliability, is not just in the components, but in the "glue" that holds the components together. Therefore, we need to understand how to achieve trustworthiness through architectures. Without this, we will be building castles in the sand. Some viable approaches have been identified [5] and should be pursued with vigor.

*Intrusion detection needs to get a whole lot better.* It is inadequate to employ a detect-respond paradigm. Recent attacks such as Slammer and Code-Red are just too fast for today's systems, which are based on detecting signatures of previously detected attacks. Experimental schemes to identify attacks based on detecting anomalies deviating from "normal" activity

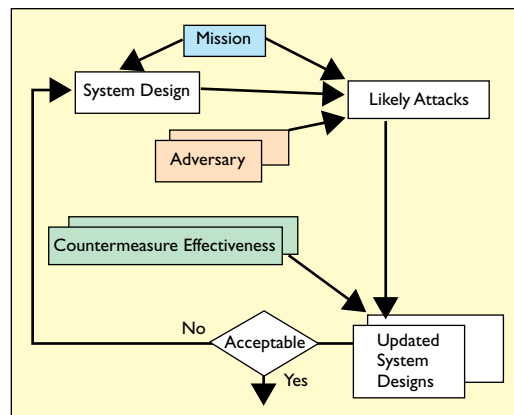


Figure 2. Cyber defense system design models.

have unacceptably high false-alarm rates and relatively poor coverage of the attack space [2]. Further, these schemes often use data from sensors originally designed for auditing security-relevant events, not for detecting attacks. Viable detection requires ground-up analysis of how attacks manifest, custom design of sensors that exploit these manifestations, proper classification algorithms for categorizing relevant events, and detection algorithms that measurably [11] cover the entire attack space.

*Intrusion response must be developed so that actions are timely and effective.* We need some degree of autonomic response for attacks that are too fast for the human decision cycle. We also must develop decision aids for enumerating situation-dependent courses of action, as well as means to evaluate those courses of action. Using human anatomy as an analogy, we need both the autonomic and the central nervous system, and they must work together to create a systematic defense.

*Defending against distributed denial-of-service attacks is needed to ensure availability.* Cutting off the many attack paths available to attackers often cuts off the very availability that one is trying to preserve. Further, traditional security and reliability remedies often worsen the problem. Solutions will almost certainly require that quality of service capabilities are added to the Internet.

*Countering life-cycle attacks is essential to trustworthiness.* If adversaries can infiltrate software development teams and insert malicious code into systems while the software is developed, they can subvert whatever trust that was established. For modern software, the development process and the resulting code are very complex, therefore making preventing and detecting subversion extraordinarily difficult. Nonetheless, we must develop techniques to detect and eradicate malicious code embedded in our code, or find ways to architecturally neutralize it.

*Scientific experimental computer science is needed to make real progress.* Much of the knowledge in cyber defense today has the status of hypothesis rather than fact. Some have significant evidence in their favor, yet they are still hypotheses. We need experimental methods, based on solid metrics, isolating single variables at a time, to convert these hypotheses into knowledge. Only this will create the firm research foundation needed to enable a sound research track.

*Controlled information sharing is needed more than ever.* Computer security has its roots in the requirement for Multilevel Security (MLS) processing. The need continues for controlled sharing among groups of differing trust relationships. Further, the need is

rapidly growing as collaboration becomes the norm in accomplishing organization goals.

On a final note, even if we make the required scientific and technological advances, we still must find ways to better integrate technology results into mainstream products and systems. Industry's and government's track record of employing useful technology results from research has been poor so far. For example, solutions to defend the vulnerable Domain Naming Service and the Border Gateway Protocol have been available for several years now, but have yet to be incorporated into the network infrastructure.

### **Creating a Systems Engineering Discipline**

**T**oday, the process of designing a well-defended system is matter of black art. One simply hopes designers were adequately knowledgeable about the range of relevant attacks and that they did an adequate job of defending against those attacks. We must evolve toward a systems engineering discipline, which urgently requires several elements.

What gets measured gets done. Without adequate metrics to assess the performance of cyber defense systems, progress is impossible to judge. Some primitive metrics have been proposed [10], but much more work remains to be done.

We need a spectrum of system models and an engineering framework analogous to the CAD/CAM framework used by hardware engineers. The community needs adequate threat models, adversary models [7], mission models, and countermeasure effectiveness models. Each type of model will require tremendous energy to produce, yet little effort is under way in these arenas.

Finally, a methodology to quantitatively trade off design factors and achieve a specified system result is needed. A vision for such a framework should be established and it should be realized with dispatch.

### **Achieving a National Cyber Defense Capability**

**S**o far, I've described cyber defense in the abstract, the principles of which apply at all scales. Here, I examine and discuss cyber defense at the macro scale of defending the national critical information infrastructure. To understand what suffices as a defense, one needs to understand vulnerabilities and the consequences of failure. That the threat is serious has been established [6]. If the reader has any doubts as to the gravity of the problem, consider the major damage done by accidental failures of the telephone system, the power grid,



and banking. Any failures that can happen by accident can likely be induced by an attacker, with significantly more damage potential [4]. The degree of that risk is hotly debated among leading professionals, but, unfortunately, opinions are founded almost entirely on speculation, as scientific study of this question has yet to be seriously undertaken. Such a study is a matter of utmost urgency to our government because the nature and scope of a national cyber defense capability must be grounded in a full comprehension of the susceptibility of our systems.

**Engineering Cyber Defense—Manhattan Project.** The National Strategy to Defend Cyberspace (released in February 2003) implies that market forces will be adequate to defend at the national scale. This is unlikely to be true [8]. By the same reasoning, market forces would be adequate to defend people, buildings, and towns against kinetic war. We can wish for that to be true. We can observe that we need a certain level of strength to survive normal stresses of life and that this strength provides some measure of defense against trivial attacks. The notion that such protection levels would withstand a nation-state attack is obviously absurd. Why then would this be different in cyberspace? Only a national scientific study will tell us for sure, but intuition tells us that an engineering capability is almost certainly needed.

How might a national cyber defense capability be engineered? What is clear is that this is a very difficult problem. Some of the requisite technology does not yet exist. Yet, all indications are that such a capability is urgent. It therefore seems reasonable to dedicate the finest scientific and engineering minds toward a concerted effort to develop the capability within three years. We need a project in the style of the Manhattan Project with the requisite national priority, resource levels, and structure. Anything less will likely stumble. History has shown us that a distributed lower-priority investment has failed to create the needed capability despite more than three decades of research and development.

**Sound National Cyber Defense Policy.** Sound national cyber defense policy depends on understanding the problem, its solutions, and the nature of cyber conflicts from economic, behavioral, and political perspectives. Unfortunately, our understanding is currently quite limited. In such a circumstance, a sound policy would be to place high priority and urgency on gaining a deep understanding of all these areas.

Furthermore, the U.S. would benefit by acknowledging that its survival and function now depend on

its information infrastructure. Sound policy would reduce the rate at which that dependency is increasing and find ways to minimize the risk as an interim measure.

Finally, I note that achieving a national cyber defense capability has the potential of infringing on citizen privacy in the effort to detect malicious network-based activity. The U.S. national policy must be to avoid abrogating the very rights it is intending to protect.

## Conclusion

Cyber defense poses serious technical and policy challenges for the U.S. Much work lies ahead in creating a stronger scientific foundation, the required technology for national-scale cyber defense, and an engineering discipline to provide the means. Policy should follow scientifically based knowledge and understanding; gaining that understanding should now be the primary objective. **C**

## REFERENCES

1. Committee on Information Systems Trustworthiness, National Research Council. *Trust in Cyberspace*. National Academy Press, Washington, D.C., 1999.
2. Haines, J., Ryder, D., Tinnel, L., and Taylor, S. Validation of sensor alert correlators. *IEEE Security and Privacy* 1 (Jan./Feb. 2003), 45–56.
3. Hamilton, S.N., Miller, W.L., Ott, A., and Saydjari, O.S. The role of game theory in information warfare. In *Proceedings of the The Fourth Information Survivability Workshop*, Vancouver, B.C., Canada, March 2002.
4. Letter to President Bush, February 27, 2002; [www.uspcd.org/letter.html](http://www.uspcd.org/letter.html).
5. Neumann, P. *Principled Assuredly Trustworthy Composable Architectures*. Draft Final Report (Oct. 2003); [www.csl.sri.com/users/neumann/chats4.pdf](http://www.csl.sri.com/users/neumann/chats4.pdf).
6. President's Commission on Critical Infrastructure Protection. *Critical Foundations: Protecting America's Infrastructure*. Washington, D.C., 1997; [www.ciao.gov/resource/pccip/PCCIP\\_Report.pdf](http://www.ciao.gov/resource/pccip/PCCIP_Report.pdf).
7. Salter, C., Saydjari, O., Schneier, B., and Wallner, J. Toward a secure system engineering methodology. In *Proceedings of New Security Paradigms Workshop* (Sept. 1998), ACM Press, New York, 1998.
8. Saydjari, O.S. Defending cyberspace. *IEEE Computer* 35 (Dec. 2002), 125.
9. Saydjari, O., Tinnel, L., and Farrell, D. Cyberwar strategy and tactics: An analysis of cyber goals, strategies, tactics, and techniques. In *Proceedings of the 2002 IEEE Workshop on Information Assurance*, June 2002, U.S. Military Academy, West Point, NY.
10. Schudel, G. and Wood, B. Adversary work factor as a metric for information assurance. In *Proceedings of New Security Paradigms Workshop* (Sept. 2000), ACM Press, New York, 2001.
11. Tan, K.M. and Maxion, R.A. Determining the operational limits of an anomaly-based intrusion detector. *IEEE Journal on Selected Areas in Communications, Special Issue on Design and Analysis Techniques for Security Assurance* 21 (Jan. 2003), 96–110.
12. Tinnel, L., Saydjari, O., and Haines, J. *An Integrated Cyber Panel System*. Supplementation to DARPA Information Survivability Conference and Exposition, April 2003, Crystal City, VA.

---

**O. SAMI SAYDJARI** ([ssaydjari@CyberDefenseAgency.com](mailto:ssaydjari@CyberDefenseAgency.com)) is the CEO of Cyber Defense Agency, LLC ([www.CyberDefenseAgency.com](http://www.CyberDefenseAgency.com)) and chairman of the Professionals for Cyber Defense ([www.uspcd.org](http://www.uspcd.org)).

---



**By Members of  
the DETER and  
EMIST Projects**

# **CYBER DEFENSE TECHNOLOGY NETWORKING AND EVALUATION**

**A**s the Internet has become pervasive and our critical infrastructures have become inextricably tied to information systems, the risk for economic, social, and physical disruption due to the insecurities of information systems has increased immeasurably. Over the past 10 years there has been increased investment in research on cyber security technologies by U.S. government agencies (including NSF, DARPA, the armed forces) and industry. However, a large-scale deployment of security technology sufficient to protect the vital infrastructure is lacking. One important reason for this deficiency is the lack of an experimental infrastructure and rigorous scientific methodologies for developing and testing next-generation cyber security technology. To date, new security technologies have been tested and validated only in small- to medium-scale private research facilities, which are not representative of large operational networks or of the portion of the Internet that could be involved in an attack.

To make rapid advances in defending against attacks, the state of the art in evaluation of network security mechanisms must be improved. This will require the development of large-scale security testbeds [3] combined with new frameworks and standards for testing and benchmarking that make these testbeds truly useful. Current deficiencies and impediments to evaluating network security mechanisms include lack of scientific rigor [6]; lack of relevant and representative network data [5]; inadequate models of defense mechanisms; and inadequate models of the network and both the background and attack traffic data [1]. The latter is challenging because of the complexity of interactions among traffic, topology, and protocols [1, 2].

To address these shortcomings, we will create an experimental infrastructure network to support the development and demonstration of next-generation information security technologies for cyber defense. The Cyber Defense Technology Experimental Research network (DETER network) will provide the necessary infrastructure—

## *Creating an experimental infrastructure for developing next-generation information security technologies.*

networks, tools, and supporting processes—to support national-scale experimentation on emerging security research and advanced development technologies. In parallel, the Evaluation Methods for Internet Security Technology (EMIST) project will develop scientifically rigorous testing frameworks and methodologies for representative classes of network attacks and defense mechanisms. As part of this research, approaches to determining domains of effective use for simulation, emulation, hardware, and hybrids of the three are being examined.

The goal of this joint effort<sup>1</sup> is to create, operate, and support a researcher- and vendor-neutral experimental infrastructure open to a wide community of users. It is intended to be more than a passive research instrument. It is envisioned to serve as a center for interchange and collaboration among security researchers, and as a shared laboratory in which researchers, developers, and operators from government, industry, and academia can experiment with potential cyber security technologies under realistic conditions, with the aim of accelerating research, development, and deployment of effective defenses for U.S.-based computer networks.

### **Information Security Challenges**

**T**o develop a testbed framework for evaluating security mechanisms, the project focuses on a select subset of the overall problem space. Several different types of attacks and defenses will be studied with two goals: to elevate the understanding of the particular attack or defense by thoroughly evaluating it via different testing scenarios; and to further the understanding of the degree to which these evaluations can be unified into a single framework that spans the diversity of the problem space.

<sup>1</sup>There are nine teams involved in the joint effort: U.C. Berkeley, U.C. Davis, University of Southern California-Information Sciences Institute (USC-ISI), Pennsylvania State University, NAI Laboratories, International Computer Science Institute (ICSI), Purdue, SPARTA Inc., and SRI International. The project also includes an industrial advisory board consisting of equipment vendors, carriers, and ISPs including AOL, Cisco, Alcatel, Hewlett-Packard, IBM, Intel, Juniper, and Los Nettos.

Three different classes of attacks are focus areas for our research: denial-of-service, worms, and attacks on the Internet's routing infrastructure, as well as attacks that are coordinated combinations of these three types. Together they span a broad range of general types of attacks. In addition, the project will closely monitor new Internet security breaches in order to analyze how new attack scenarios can be incorporated into the developing testing methodology. The focus of this effort will be on attacks targeting network infrastructure, server end-systems, and critical end-user applications. Such attacks are difficult to accurately simulate using existing testing frameworks because of the major challenges in accurately simulating Internet phenomena in general [2, 4].

### Security Testing Methodologies

**T**esting frameworks will be adapted for different kinds of testbeds, including simulators such as NS (see [www.isi.edu/nsnam/ns](http://www.isi.edu/nsnam/ns)), emulation facilities such as Emulab [8], and both small and large hardware testbeds. The frameworks will include attack scenarios, attack simulators, generators for topology and background traffic, data sets derived from live traffic, and tools to monitor and summarize test results. These frameworks will allow researchers to experiment with a variety of parameters representing the network environment including attack behaviors, deployed defense technology, and the configuration of the defense mechanisms under test. It will be critical to make progress on the very difficult problems, particularly:

- How to construct realistic topologies, including bandwidth and inter-AS policies,
- How to generate realistic cross-traffic across these topologies,
- How to quantify how accurate the models need to be, and
- How to select the best metrics for evaluating various defense mechanisms.

Conducting these tests will require incorporating defense mechanisms into a testbed (either as models or as operational code), and applying and evaluating the frameworks and methodologies. Conducting these tests will also help to ensure the testbed framework allows other researchers to easily integrate and test network defense mechanisms of their own design. Furthermore, the documentation of the tests will serve as a tutorial for users of the testbed framework as they confirm their results or evaluate their own mechanisms and techniques.

### Testbed Architecture and Requirements

**T**he preliminary requirements for the DETER Testbed are drawn from four sources: a DARPA-funded study of security testbed requirements [3], input from network security researchers, general considerations on network research testbeds through a NSF workshop [4], and experience with a variety of earlier experimental and test networks. High-level requirements are briefly described here.

The general objectives for the testbed design require that it must be fully isolated from the Internet and all experiments must be soundly confined within the DETER network. Furthermore, it is expected the network will be subjected to destructive traffic and that experiments may temporarily damage the network. Therefore, there must also be mechanisms for rapid reconstitution of the testing environment.

The scale of the testbed will be approximately 1,000 PCs, each with multiple network interface cards, and a significant number of commercial routers and programmable switches. Within this environment, the network must provide sufficient topological complexity to emulate a scaled down but functionally accurate representation of the hierarchical structure of the real Internet, and to approximate the mixing of benign traffic and attack traffic that occurs. Initially, the network will be formed using a homogeneous network of existing technology. Carefully chosen hardware heterogeneity—commercial router boxes—will be added as the effort progresses. Finally, conducting experiments with large-scale denial-of-service attacks and defense technologies to protect the Internet infrastructure will require high-bandwidth componentry.

In addition to the preceding infrastructure requirements, there are various requirements for software to facilitate experimentation. The utility of DETER will depend on the power, convenience, and flexibility of its software for setting up and managing experiments including registration, definition, generation control, monitoring, check-pointing, and archiving. An important aspect of the management software will be the requirement for sophisticated network monitoring and traffic analysis tools for both experimenters and DETER network operators. Experimenters will also require traffic generation software to generate attack traffic and typical day-to-day (legitimate use) traffic.

**Preliminary Architecture.** DETER will be built as three permanent hardware clusters, located at ISI in Los Angeles, ISI-East in Virginia, and UC-Berkeley. To provide the earliest possible service to experimenters, initial development during the first six months focuses on building software and configurations for cyber security experimentation on PlanetLab and/or Emulab [8].

The architecture will also deploy aspects of the X-bone (see [www.isi.edu/xbone](http://www.isi.edu/xbone)) to allow topologies with revisitation, where, for example, a 10-node ring can be used to emulate a 100-node ring by visiting the same node multiple times. During the early stages of the testbed, this will enable the simulation of topologies that are larger than can be supported with one-to-one mapping of physical resources. Meanwhile, a phased development effort, moving from carefully controlled emulation environments to a mix of emulation and real network hardware will occur.

## Conclusion

**T**he development of testing methodologies complemented by an experimental infrastructure will support the realistic and consistent evaluation of mechanisms purported to mitigate large-scale attacks. This is an extremely challenging undertaking—no existing testbed or framework can be claimed to be effective. The research described here requires significant advances in the modeling of network attacks and the interactions between attacks and their environments, including deployed defense technology, background traffic, topology, protocols, and applications. It will also require advances in the understanding of metrics for evaluating defense mechanisms.

Our results will provide new scientific knowledge to enable the development of solutions to cyber security problems of national importance. This will be accomplished through experimentation and validation of cyber defense technologies using scientific methods. The lack of open, objective, and repeatable validation of cyber defense technologies has been a significant fac-

tor hindering wide-scale adoption of next-generation solutions. Results obtained using the DETER testbed will contribute to the development of innovative new technologies that increase commercial availability and viability of new production networks and services, providing true cyber protection. **G**

## REFERENCES

1. Floyd, S. and Kohler, E. Internet research needs better models. *Homets-I* (Oct. 2002).
2. Floyd, S. and Paxson, V. Difficulties in simulating the Internet. *IEEE/ACM Transactions on Networking* 9, 4 (Aug. 2001), 392–403.
3. Hardaker, W. et al. *Justification and Requirements for a National DDoS Defense Technology Evaluation Facility*. Network Associates Laboratories Report 02-052, July 26, 2002.
4. Kurose, J., Ed. *Report of NSF Workshop on Network Research Testbeds* (Nov. 2002); [gaia.cs.umass.edu/testbed\\_workshop](http://gaia.cs.umass.edu/testbed_workshop).
5. McHugh, J. Testing intrusion detection systems: A critique of the 1998 and 1999 DARPA intrusion detection system valuations as performed by Lincoln Laboratory. *ACM Transactions on Information and System Security* 3, 4 (Nov. 2000), 262–294.
6. Pawlikowski, K., Jeong, H., and Lee, J. On credibility of simulation studies of telecommunication networks. *IEEE Communications Magazine* (Jan. 2001).
7. Peterson, L., Anderson, T., Culler, D., and Roscoe, T. A blueprint for introducing disruptive technology into the Internet. In *Proceedings of the 1st ACM Workshop on Hot Topics in Networks (HotNets-I)* (Oct. 2002), 4–140.
8. White, B. et al. An integrated experimental environment for distributed systems and networks. In *Proceedings of the Fifth Symposium on Operating Systems Design and Implementation (OSDI02)*, (Dec. 2002).

**MEMBERS OF THE DETER AND EMIST NETWORK PROJECT INCLUDE R. BAJCSY, T. BENZEL, M. BISHOP, B. BRADEN, C. BRODLEY, S. FAHMY, S. FLOYD, W. HARDAKER, A. JOSEPH, G. KESIDIS, K. LEVITT, B. LINDELL, P. LIU, D. MILLER, R. MUNDY, C. NEUMAN, R. OSTRENGA, V. PAXSON, P. PORRAS, C. ROSENBERG, J.D. TYGAR, S. SASTRY, D. STERNE, AND S.F. WU.**

This project is funded jointly by the U.S. Department of Homeland Security (DHS) and the National Science Foundation (NSF) under grant ANI-0335241.

© 2004 ACM 0002-0782/04/0300 \$5.00

## Coming Next Month in Communications

### *Etiquette for Human-Computer Relations*

A growing community of computer scientists, researchers, sociologists, psychologists, educators, and industry practitioners are taking the “etiquette perspective” in designing, building, and analyzing human interaction with computers and other forms of advanced automation. And they are finding, among its many advantages, the etiquette approach facilitates user acceptance of systems and products and, more importantly, improves the accuracy and speed with which the users develop trust in such products.

It’s a practice in its infancy and next month’s special section introduces the concept of etiquette and provides a variety of perspectives on its use and importance, including a number of examples of current research and applications.

*Also in April* Software Project Risks and their Impact on Outcomes • Cross-Cultural Issues in Global Software Outsourcing • Who Should Work for Whom? • Managing Academic E-Journals • Economics of Wireless Networks • Managing Knowledge in Distributed Projects • Information Cascades in IT Adoption

By Steve Sawyer, Andrea  
Tapia, Leonard Pesheck,  
and John Davenport

# Mobility and the First

ILLUSTRATION BY  
ROBERT NEUBECKER

**H**ow can information and communications technologies be better used to support the more than eight million first responders in U.S. homeland security? First responders are members of organizations and agencies such as emergency communications centers; emergency medical services; fire, rescue, and hazardous material response teams; law enforcement agencies; the Red Cross, and other disaster relief organizations [1]. In most cases first responders are those people who, during an event or incident, are the prime evaluators of threat and risk to homeland security. These people become the primary link in a chain of information exchanges that lead to making critical, perhaps lifesaving, decisions.

A typical first responder's work is characterized by routine occurrences punctuated by periodic emergencies. First responders' work is often structured around responding to incidents and events, the cause, severity, and consequences of which are not readily discernable. These incidents rarely occur at predetermined places or times. Thus, the routine patrol of a police officer can shift to emergency mode as a result of a single call from the police dispatcher. Many first responders are mobile as part of their routine work and must relocate to an incident site in an emergency, which means they must bring what they need with them or function with what they have to contain the situation [6]. However, some assessments of U.S. homeland security lament that first responders' needs for information access and sharing are not well supported, and are often disconnected from both the information systems and databases central to effective homeland security [5, 8].

Our interests in first responders, homeland security, and uses of computing motivated us to conduct a field trial of mobile access to the Commonwealth of Pennsylvania's Justice Network (JNET). For the purposes of this trial, mobile access meant using third-generation (3G) public wireless networks. A 3G network provides enough bandwidth to transmit photos and other large files securely to mobile and remote users. Several commercial wireless service providers (such as Horizon PCS, a Sprint subsidiary used for our trials) have built out their 3G networks and one goal of the field trial was to assess whether first responders could effectively use this public telecommunications



# Responder

*Supporting secure wireless access to databases via public telecommunications infrastructure.*

infrastructure for their routine and emergency needs.

We partnered with JNET because this application provides a secure means to search more than 20 public-safety-related databases<sup>1</sup> [2]. A key aspect of the JNET architecture is the access it provides to residents' driver's license photos. Secure (and authorized) access to this range of available criminal justice data has been considered both critical and not possible for most public safety and police officers to date. The JNET approach—federated databases connected through a Web-based application/portal—reflects the new range of systems being developed to support criminal justice and homeland security work [2]. JNET's fixed-site desktop access currently experiences 45,000 hits and 2,000 interagency notifications per month and has been a significant asset to Pennsylvania's criminal justice efforts [7]. Thus, a second goal of the field trial was to better understand the technical needs, operational uses, and strategic opportunities of first responders' mobile access to JNET (and to the Web more generally) via laptop computers and PDAs using public 3G networks—see the sidebar, “The Field Study Method.”

## Observations from the Field Trials

The four observations we discuss are drawn from interviews, time diaries, observations, and ride-alongs, call logs, and unobtrusive traffic records.

*Mobile access to JNET is a 'killer application' for first responders.* Driver's license photos were the most requested information, just as they are for fixed-site JNET users. Driver's license photos provide a means of identifying and linking people (and their pictures) to vehicle registration, addresses, and other activities such as warrants, tickets, and other criminal justice incidents. JNET's

---

<sup>1</sup>The JNET program is operated by the Pennsylvania Office of IT and 15 Pennsylvania criminal justice agencies. For more on JNET and its role in both public safety and U.S. homeland security, see [www.pajnet.state.pa.us](http://www.pajnet.state.pa.us).

value to trial participants is evident even though they had to grapple with the constraints of limited coverage on, and unstable access to, the 3G wireless network. For example, trial participants are highly conscious of security of information and they valued the steps taken by JNET to keep information secure during the field trial even though it added several steps to the log-in process.

We learned that connection reliability is more important to officers than is data download speed. When officers need identity or criminal justice data during an incident, they radio the police dispatcher. The dispatcher's proxy query takes place in parallel with the officer's incident management at the scene. Thus, there is no time penalty during a situation in which officers at the scene typically cannot divert their attention to deal with a query/response. Therefore, officers depend on the dispatcher to return information. More generally, we realize that the dispatch model is so central to current first-responder processes that it should be considered an integral aspect of new applications and not seen as an organizational work structure to be changed.

This observation suggests two things. First, that the value of high-speed access to data for trial participants is not tied to how much or how often, it is a matter of connectivity when it is needed. The common conceptualizations of value being measured by volume or use time are incorrect. Second, that future developments of JNET (and applications to support first responders) should be designed to work with existing dispatcher schemes. For example, if a JNET request was initiated by an officer (perhaps as a voice-driven query) and this request was completed and sent through dispatch, the officer would continue to have hands-free (and eyes-free) operations, the current working dispatch-centric model would be supported, and the query's results could be easily sent on to other units as needed.

*Trial participants have welcomed the mobile devices and advanced information and communications tech-*


*nologies in general.* Participants are hopeful about the roles that mobile devices and wireless access can play in making their work life safer and also better enable them to perform their duties. They see the devices as part of a larger ensemble that could include local printers, digital cameras, driver's license scanners, and the ability to file reports via wireless connectivity: they want more. The officers in our trial are patient and willing to wait until something is proven to work before incorporating it as part of their daily routine. Based on the positive results of the laptop trial both the participants and study team expected significant usage of the PDA. However, the PDA battery life was not sufficient to maintain connectivity with the 3G network over long periods and this led trial participants to stop using their PDAs for mobile access. Instead, officers used PDAs for scheduling, contacts, note-taking and many other tasks. Our experience indicates first responders are willing to take on new tools, but will not compromise their (or anyone else's) safety if the device or application does not work. The JNET applications that are very useful for deskbound workers are neither fast enough nor focused on the needs of mobile workers, making use difficult during incident response.

We also note that the value of mobile access seems to be tied to particular aspects of their work. Mobile access and JNET use seems important to only certain tasks and events in the work of our participants. For example, in the eight-hour shifts we observed during ride-alongs, officers typically were engaged in information-seeking tasks for less than 15% of the total shift time. Self-reported time-diaries corroborate that information-seeking activities are a small but very critical aspect of police officer's work.

*Use of mobile JNET does not alter existing organizational links.* We imagined at the trial's outset that increased access to information might lead to changed interactions among personnel; there is no evidence of this in our experience. One possible artifact of the field

### The Field Study Method

The trial was designed in two phases. The first was limited to five participants and focused on laptop usage. The second involved 13 participants and focused on PDA usage. Both phases lasted three months. Participants were police and other public-safety officers from within one Pennsylvania county. We used a four-pronged mixed-method approach to gather data: interviews and focus groups; ride-alongs and direct observation; pre- and post-trial survey data; and unobtrusively collected data on actual Web use, JNET use, and wireless connection use. This combination of methods allowed us to answer questions

about where, when, and why this technology was used and why not. These methods also allowed us to answer important questions about first responders' and criminal justice organizations' unique use of mobile technology. The field trial arises from an ongoing partnership among: Pennsylvania state government (Office of Information Technology and Justice Network Project); Lucent Technologies; Boston University's Institute for Leading in a Digital Economy; Pennsylvania State University's School of Information Sciences and Technology; Horizon PCS; and Novatel. 

trial is that it created a reason for a number of local and county units to work together, and this has had an unintended but welcome positive effect of collaboration. These newly exercised links have led to demonstration efforts for other county public safety offices and local police units and interest in developing community policing grant proposals. But, in the short span of our six-month, two-phase trial, involving the day-to-day work of policing and public safety, JNET and mobile access has not changed communication patterns.

*First-responder organizations have limited IT support and diverse IT infrastructures.* The officers in our trial relied chiefly on themselves and on each other to learn to use and troubleshoot the laptops and PDAs. Each of the three units participating in the trial had different IT infrastructures and often these were supported through a variety of contracts to different third-party vendors. This is common in public-sector IT: limited IT support and piecemeal IT infrastructures [4]. During the trial we dedicated 20 hours per week of technical support for the participants and we were always over-tasked. It could be that production deployment may be easier to support if the systems are extremely reliable and devices/applications are designed for specific use by first responders. However, increased local IT support is crucial.

## Implications and Issues

**W**ireless communication devices may have a role in facilitating communications between criminal justice personnel, but in this case they do not reduce the number of people involved in the process of completing any task, change the roles that any person currently plays, or reduce the number of steps in any process. The real implication for wireless computing is using the current people and processes—but allowing information to flow more quickly from repositories to people, and from person to person, at very important critical moments.

Successful systems in support of first responders will be driven by their users [3]. Our trial experience indicates laptop computers are too big and too tethered to the officer's car while PDAs are too small and have a limited amount of battery life. We learned that these officers must have adequate power and screen size to use the graphic data they require and cannot always be tethered to their cars to acquire this data. Thus, future trials should test pen-based tablets that are ruggedized to meet first responders' work conditions. In addition, all future devices should be designed to load photos and maps quickly since they are in the highest demand.

In order for wireless access to find a place on the officers' tool belts, it needs to be reliable, as it could be necessary to save their lives or the lives of those they

protect. The data these officers require is time- and task-specific—just as always-on Internet connections are replacing dialup modem connectivity, first responders need an always-on network. In addition, they must know this access will be there whether they need it once a week or once a year. To achieve this, IT security measures must be made more streamlined, seamless, and immediate to provide officers with the information they require on demand.

Beyond the technical constraints and opportunities identified by our trial, we have four findings reflecting the role of this trial in strategic experimentation involving mobility. First, applications like JNET will be key elements in any information system to support first responders. Second, the use of public 3G networks to support U.S. homeland security is possible only if coverage and reliability goals are met. Third, such systems will require a federated view of the entire enterprise, demanding a focus on interoperability. There are too many critical and unique information systems supporting so many disparate organizations to imagine one large system. Instead, our experience suggests that portal and broker models such as JNET are the proper architecture to support first responders. Finally, we have learned that future trials should rely more on current dispatch and local control models with a focus on coordination. These are some of the issues to be tested in the next strategic experiment. **C**

## REFERENCES

1. First Responder Association; [www.wmdfirstresponders.com/](http://www.wmdfirstresponders.com/).
2. Harris, K. *Integrated Justice Information Systems: Governance Structures, Roles and Responsibilities*. The National Consortium for Justice Information and Statistics, 2000; [www.search.org/integration/default.asp](http://www.search.org/integration/default.asp).
3. Manning, P. Information technology in the police context: The 'Sailor' phone. *Information Systems Research* 7, 1 (1996), 275–289.
4. National Association of State Chief Information Officers. *Best Practices in the Use of Information Technology in State Government*. NASCIO, 2002, Lexington, KY.
5. National Science Foundation. Award announcement to support *Responding to the Unexpected*; [www.nsf.gov/od/lpa/news/03/pr03103.htm](http://www.nsf.gov/od/lpa/news/03/pr03103.htm), July, 2003.
6. Nulden, U. Investigating police practice for design of IT. In *Proceedings of CHI 2003*, 820–821.
7. Pennsylvania Justice Network (JNET). *JNET Project Summary, 2003*; [www.pajnet.state.pa.us/pajnet/cwp/view.asp?a=2138&q=75743](http://www.pajnet.state.pa.us/pajnet/cwp/view.asp?a=2138&q=75743).
8. Rudman, W., Clarke, R. and Metzger, J. *Emergency Responders: Drastically Underfunded, Dangerously Unprepared*. Report of an Independent Task Force sponsored by the Council on Foreign Relations, July, 2003; [www.cfr.org/pdf/Responders\\_TF.pdf](http://www.cfr.org/pdf/Responders_TF.pdf).

**STEVE SAWYER** ([sawyer@ist.psu.edu](mailto:sawyer@ist.psu.edu)) is an associate professor in the School of Information Sciences and Technology at Pennsylvania State University.

**ANDREA TAPIA** ([atapia@ist.psu.edu](mailto:atapia@ist.psu.edu)) is an assistant professor in the School of Information Sciences and Technology at Pennsylvania State University.

**LEONARD PESHECK** ([pesheck@lucent.com](mailto:pesheck@lucent.com)) is the manager of Lucent Technologies' Mobility Solutions Trials Department, Naperville, IL.

**JOHN DAVENPORT** ([c-jdavenport@state.pa.us](mailto:c-jdavenport@state.pa.us)) is a technology advisor with JNET Office, Harrisburg, PA.

By Robin R. Murphy

# Rescue Robotics for H

*Applying hard-earned lessons to improve human-robot interaction and information gathering.*

One aspect of homeland security is response: what to do when disaster occurs? Rescue robotics is gaining increasing attention as the first new advance in emergency response since the advent of boroscopes and cameras on poles 15 years ago. The attention is due in part to the use of rescue robots at the site of the collapse of the World Trade Center towers in New York. The WTC response was an example of an urban search and rescue (USAR) mission. USAR deals with man-made structures, has a different emphasis than traditional wilderness rescue or underwater recovery efforts, and can be even more demanding on robot hardware and software design than military applications.

Robots were used from Sept. 11–Oct. 2, 2001 to search for victims and help assess the structural integrity of the WTC foundation under the direction of the Center for Robot-Assisted Search and Rescue (CRASAR). Teams from DARPA, Foster-Miller, iRobot, U.S. Navy SPAWAR, the University of South Florida, and Picatinny Arsenal operated small robots, some small enough to fit into backpacks. The robots were used for tasks that rescuers or canines couldn't perform, for example, to either go into spaces that are too small for a human or to pass through an area on fire or without breathable air in an attempt to reach a survivable void. In the two years since the WTC incident, CRASAR has expanded the utility of rescue robots to include victim management (adding two-way audio, sensors for triage, mechanisms for fluid delivery, remote reachback to medical specialists) and is working to use robots for shoring up structures to speed up extrications of survivors.

The roots of the WTC response can be traced back to the Oklahoma City bombing in 1995, which motivated our interest in the domain of rescue robotics for urban search and rescue. In 1996, we established a small cache of rescue robots with funding from the National Science Foundation, and in 1999 we began extensive field studies and technical search specialist training with Florida Task Force 3, a state regional response team. Also in 1999, both the American Association for Artificial Intelligence and the RoboCup Federation started rescue robot competitions to foster



# Homeland Security

research in this humanitarian application of mobile robots. But good theory does not necessarily lead to good practice. None of the algorithms demonstrated by CRASAR or other groups at various rescue robot competitions or at related DARPA programs were actually usable on robots that could withstand the rigors of real rubble. As a result, all the robots used at the WTC site had to be teleoperated.

**I**n the aftermath of Sept. 11, CRASAR has taken on the role of midwife to smooth the transition of key research from laboratories all over the world to the hands of responders. We became an independent center at the University of South Florida and created a formal response team with scientists and medical personnel trained for USAR. Recently we were awarded a NSF planning grant with the University of Minnesota for an Industry/University Cooperative Research Center on Safety, Security, and Rescue Robotics to further encourage investment by companies in this critical area.

As a result of our expertise as both researchers and users, we have identified many major research issues. The physical attributes of the robot itself require improvement. Currently, no robots are made specifically for USAR. Mobility experts often make the mistake of developing platforms that can climb over rubble, not crawl



**A view of the Inuktun micro-VGTV robot being inserted into a sewer pipe at the World Trade Center site in an attempt to locate an entry to the basement. Note the small size of the robot, the use of a tether as a safety line for the vertical entry, and the use of a camcorder to display the video because of better resolution than the manufacturer's display. This is the only external view of the robots that was allowed to be photographed at the WTC site.**



vertically into the interior. We find a more significant limitation is the lack of sensors that can be mounted on the robots. Many navigation and mapping algorithms exist that are likely to be useful for USAR but require multiple sensors approximately the size of an average countertop coffee maker. Given the most useful robot size is about the dimensions of a shoebox, these sensors can't be used. The complexity of the environment—highly confined, cluttered—viewed with video and FLIR (forward-looking infrared) cameras mounted only a few inches above the ground present a formidable challenge for autonomous control and especially perception. Indeed, several sets of victims' remains were missed at the WTC due to these issues in perception.

Communications remains a huge issue. Micro-sized robots require a tethered connection back to a power and control source. The tether tangles but also serves as a safety line as the robot descends down into the pile. Wireless robots are larger and more mobile, but still require a safety line and communications is easily lost in the dense rubble. Indeed, the only robot lost in the rubble at the WTC and not recovered was a wireless robot, which lost its control link and did not have any onboard intelligence to attempt to move and reacquire the signal. Trade-offs between tethers and wireless connectivity must be explored. All of these issues will be exacerbated by chemical, biological, or radiological events. Robots will have to be hardened, easy to operate from protective gear, and easy to decontaminate (or be inexpensive enough to be disposed of).

**I**n the rush to contribute their particular areas of expertise, it is easy for researchers to forget that security, especially emergency response, is a human endeavor. Robots and agents will not act alone but rather in concert with a spectrum of trained professionals, cognitively and physically fatigued individuals, motivated volunteers, and frightened victims. One impact of the human side is that rescue workers today refuse to consider fully autonomous systems designed to act as "yes/no there's something down there" search devices. Scenarios where a swarm of hundreds of robot insects are set loose to autonomously search the rubble pile and report to a single operator not only appear impractical for search (and certainly for structural assessment, victim management, and extrication as well), but also ignore the organizational context of USAR, which has a hierarchy of operators who check and verify any findings from dogs, search cameras, or other sources. As a result, we believe good human-robot interaction (HRI) is critical to

the acceptance and success of rescue robot systems.

In the USAR domain, the robot exists to provide information to rescue workers as well as interact with the victim. While these groups may not operate the robot, they must interact with it. On the other hand, HRI is not about all the other members of the team, it can also yield insights into traditional robot-operator relationships. We've encountered some surprises as well when we examined the interaction of the robot operator and robot in the field, both from our tapes of the Sept. 11 WTC response and our numerous field studies. Working with cognitive scientists and industrial psychologists, we've discovered strong evidence that it takes two people to operate one robot: one operator focused on the robot (how to navigate through this vertical drop without tangling the safety rope?) and what David Woods and his team at Ohio State refers to as a problem-holder (am I looking at signs of a survivor?).

CRASAR has many publications and reports on rescue robotics, most of which are available on our Web site, [www.crasar.org](http://www.crasar.org). We also have limited opportunities for researchers to accompany us into the field and collect data with fieldable robots and rescue professionals as part of the NSF-funded R4: Rescue Robots for Research and Response project. **C**

#### FURTHER READING

1. Burke, J., Murphy, R.R., Covert, M., and Riddle, D. Moonlight in Miami: An ethnographic study of human-robot interaction in USAR. *Human-Computer Interaction, special issue on Human-Robot Interaction* 19, 1–2 (2004).
2. Casper, J. and Murphy, R.R. Human-robot interaction during the robot-assisted urban search and rescue response at the World Trade Center. *IEEE Transactions on Systems, Man, and Cybernetics* 33, 3 (June 2003), 367–385.
3. Murphy, R.R. Rescue robots at the World Trade Center from Sept. 11–21, 2001. *IEEE Robotics and Automation Magazine* (June 2004).
4. Murphy, R.R., Blitch, J. and Casper, J. AAI/RoboCup-2001 urban search and rescue events: Reality and competition. *AI Magazine* 23, 1 (Jan. 2002), 37–42.

---

**ROBIN R. MURPHY** ([murphy@csee.usf.edu](mailto:murphy@csee.usf.edu)) is the director of the Center for Robot-Assisted Search and Rescue at the University of South Florida.

---

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

---



by GANG WANG, HSINCHUN CHEN,  
and HOMA ATABAKHSH

# AUTOMATICALLY DETECTING DECEPTIVE CRIMINAL IDENTITIES

Fear about identity verification reached new heights since the terrorist attacks on Sept. 11, 2001, with national security issues related to detecting identity deception attracting more interest than ever before. Identity deception is an intentional falsification of identity in order to deter investigations. Conventional investigation methods run into difficulty when dealing with criminals who use deceptive or fraudulent identities, as the FBI discovered when trying to determine the true identities of 19 hijackers involved in the attacks. Besides its use in post-event investigation, the ability to validate identity can also be used as a tool to prevent future tragedies.

Here, we focus on uncovering patterns of criminal identity deception based on actual criminal records and suggest an algorithmic approach to revealing deceptive identities.

Interpersonal deception is defined as a sender knowingly transmitting messages intended to foster a false belief or conclusion by the receiver [1]. Methods have been developed to detect deception

using physiological measures (for example, polygraph), nonverbal cues, and verbal cues. Nonverbal cues are indications conveyed through communication channels such as micro-expression (for example, facial expression), eye movement, and body language. Verbal cues are linguistic patterns exhibited in messages that may include deception. The veracity of verbal cues can be measured

THE CRIMINAL MIND IS NO MATCH FOR SOME OF  
THE LATEST TECHNOLOGY DESIGNED TO DETERMINE  
FACT FROM FICTION IN SUSPECT IDENTITIES.

ILLUSTRATION BY TERRY MIURA

## WE FOCUS ON UNCOVERING PATTERNS OF CRIMINAL IDENTITY DECEPTION BASED ON ACTUAL CRIMINAL RECORDS AND SUGGEST AN ALGORITHMIC APPROACH TO REVEALING DECEPTIVE IDENTITIES.

by empirical techniques (for example, Statement Validity Assessment and Criteria-Based Content Analysis) [7]. Police officers are trained to detect lies by observing nonverbal behaviors, analyzing verbal cues, and/or examining physiological variations. Some are also trained as polygraph examiners. Because of the complexity of deception, there is no universal method to detect all types of deception. Some methods, such as physiological monitoring and behavioral cues examination, can only be conducted while the deception is occurring. Also, there is little research on detecting deception in data where few linguistic patterns exist (for example, profiles containing only names, addresses, and so on). Therefore, existing deception detection techniques developed for applications in communication and physiology are not suitable for discovering deception in identity profiles.

It is a common practice for criminals to lie about the particulars of their identity, such as name, date of birth, address, and Social Security number, in order to deceive a police investigator. For a criminal using a falsified identity, even if it is one quite similar to the real identity recorded in a law enforcement computer system, an exact-match query can do very little to bring up that record. In fact, criminals find it is easy and effective to escape justice by using a false identity.

A criminal might either give a deceptive identity or falsely use an innocent person's identity. There are currently two ways law enforcement officers can determine false identities. First, police officers can sometimes detect a deceptive identity during interrogation and investigation by repeated and detailed questioning, such as asking a suspect the same question ("What is your Social Security number?") over and over again. The suspect might forget his or her false answer and eventually reply differently. Detailed questioning may be effective in detecting lies, such as when a suspect forgets detailed information about the

person whose identity he or she is impersonating. However, lies are difficult to detect if the suspect is a good liar. Consequently, there are still many deceptive records existing in law enforcement data. Sometimes a police officer must interrogate an innocent person whose identity was stolen, until the person's innocence is proven.

Second, crime analysts can detect some deceptive identities through crime analysis techniques, of which link analysis is often used to construct criminal networks from database records or textual documents. Besides focusing on criminal identity information, link analysis also examines associations among criminals, organizations, and vehicles, among others. However, in real life crime analysis usually is a time-consuming investigative activity involving great amounts of manual information processing.

### Record Linkage Algorithm

A literature survey was conducted to identify research that could contribute to our understanding of criminal profile analysis. In his review of this field, Winkler [8] defined record linkage as a methodology for bringing together corresponding records from two or more files or for finding duplicates within a file. Record linkage originated from statistics and survey research. Newcombe [5] pioneered this work in a study designed to associate a birth record in a birth profile system with a marriage record in a marriage profile system if information in both records pointed to the same couple. His work enabled the first computerized approach to record linkage. In recent years, record linkage techniques have incorporated sophisticated theories from computer science, statistics, and operations research [8]. Work on library holdings duplication is also a related field.

Two basic components in record linkage are the *string comparator* and the *weight determination*

method. The string comparator can determine the degree of agreement between corresponding attributes, such as names, in two records. The weight determination is a mechanism to combine agreement values of all fields and of results in an overall degree of agreement between two records. The performance of a string comparator is very important because it is the key component in computing agreement values. Although current string comparator methods employed in record linkage have different limitations, they can be improved significantly for various applications.

**Phonetic string comparator.** To compute agreement values between surnames, Newcombe [5] encoded surnames using the Russell Soundex Code, which represented the phonetic pattern in each surname. According to the rules of Soundex coding, surnames were encoded into a uniform format having a prefix letter followed by a three-digit number. Surnames having the same pronunciation in spite of spelling variations should produce identical Soundex codes. For example, “PEARSE” and “PIERCE” are both coded as “P620.” However, Soundex does not work perfectly. In some cases, names that sound alike may not always have the same Soundex code. For example, “CATHY” (C300) and “KATHY” (K300) are pronounced identically. Also, names that do not sound alike might have the same Soundex code; for example, “PIERCE” (P620) and “PRICE” (P620).

**Spelling string comparator.** A spelling string comparator compares spelling variations between two strings instead of phonetic codes. In another pioneering record linkage study, Jaro [3] presented a string comparator dealing with typographical errors such as character insertions, deletions, and transpositions. This method has a restriction in that common characters in both strings must be within half of the length of the shorter string.

String comparison, whether string distance measures or string matching, has also attracted the interest of computer scientists. A common measure of similarity between two strings is defined by Levenshtein as “edit distance” [4], that is, the minimum number of single character insertions, deletions, and substitutions required to transform one string into the other. The edit distance measure outperforms Jaro’s method because it can deal with all kinds of string patterns. Since edit distance is designed to detect spelling differences between two strings, it

does not detect phonetic errors.

Porter and Winkler [6] showed the effect of Jaro’s method and its several enhanced methods on last names, first names, and street names. In order to compare the Soundex coding method, Jaro’s method, and edit distance, we calculated several string examples (used in [6]) using Soundex and edit distance respectively. Table 1 summarizes a comparison of the results from Soundex, Jaro’s method, and edit distance. Each number shown in the table represents a similarity measure (a scale between 0 and 1) between the corresponding strings. We noticed that Soundex measures gave improper ratings when two strings happened to be encoded similarly, such as “JONES” (J520) and “JOHNSONS” (J525), “HARDIN” (H635) and “MARTINEZ” (M635). Edit distance measures were capable of reflecting the spelling differences in cases where Soundex measures were improper. Jaro’s method could also detect spelling variations between strings. However, it was unable to compare certain string patterns (with scores of zero). In order to cap-

ture both phonetic and spelling similarity of strings, a combination of edit distance and Soundex was selected for our research.

## A Taxonomy of Criminal Identity Deception

In order to identify actual criminal deception patterns, we conducted a case study on the 1.3 million records at Tucson Police Department (TPD). Guided by a veteran police detective with over 30 years of service in law enforcement, we identified and extracted 372 criminal records involving 24 criminals—each having one real identity record and several deceptive records. The 24 criminals included an equal number of males and females, ranging in age from 18 to 70. Records contained criminal identity information, such as name, date of birth (DOB), address, identification numbers, race, weight, and height. Various patterns of criminal identity deception became apparent when we compared an individual’s deceptive records to his or her real identity record.

Or discarded physical description attributes (for example, height, weight, hair color, eye color) that had little consequence for deception detection. With visual scrutiny, suspects apparently do not lie about their height or weight. Eye color and hair color are too unreliable to be of any real importance. Criminals can

A pair of strings		Soundex	Jaro’s	Edit distance
JONES	JOHNSONS	0.75	0.79	0.50
MASSEY	MASSIE	1.00	0.889	0.66
SEAN	SUSAN	0.50	0.783	0.60
HARDIN	MARTINEZ	0.75	0.00	0.50
JON	JAN	1.00	0.00	0.66

Table 1. Comparison between Soundex, Jaro’s method, and Edit distance.

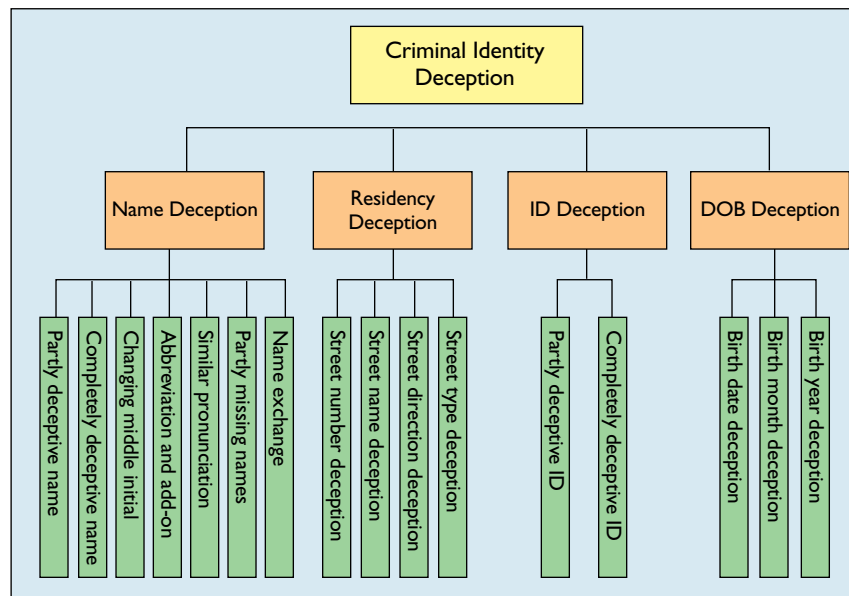


WE PRESENT A RECORD-LINKAGE METHOD BASED ON STRING COMPARATORS TO ASSOCIATE DIFFERENT DECEPTIVE CRIMINAL IDENTITY RECORDS. THE EXPERIMENTAL RESULTS HAVE SHOWN THE METHOD TO BE PROMISING.

easily make changes to those attribute.

In the remaining attributes we found different patterns of deception in each one. Consequently, we cat-

- *Changing middle initial:* Instead of a full middle name, only middle initials are shown in the police profiles. 62.5% of the records had modified middle initials, while the first name and last name remained intact. Criminals either left out or changed their middle initials. Also, they sometimes fabricated a middle initial when there was none.



- *Abbreviation and add-on:* 29.2% of the criminal records had abbreviated names or additional letters added to their real names. An example of this is using “Ed” instead of “Edward,” or “Edwardo” instead of “Edward.”

- *Similar pronunciation:* This means using a deceptive name having the same or similar pronunciation, but spelled differently. In our sample,

egorized criminal identity deception into four types: name deception, residency deception, DOB deception, and ID deception. The taxonomy of criminal identity deception was built upon the case study and is summarized in Figure 1.

**Name deception** can take on a variety of options:

- *Partly deceptive name:* 62.5% of the criminal records in the sample data set had more than once given either a false first name or a false last name. For example, “Ed Garcia” might have been changed to “Ted Garcia.”
- *Using a completely different name:* 29.2% of the records had completely false names. Both first name and last name were false.

42% of the criminals used this method of deception. For example, “Cecirio” can be altered to “Cicero.”

- *Name exchange:* 8% of the criminals transposed last and first names. For example, “Edward Alexander” might have become “Alexander Edward.”

**DOB deception** is easier than name deception to define simply because it consists of year, month, and day. By studying the deceptive cases, we found that suspects usually made only slight changes to their DOBs. For example, “02/07/70” might have been falsified as “02/08/70.” Changes to month or year also were frequent in the sample. In all DOB deception cases in our sample, 65% only falsified one por-

tion of their DOB, 25% made changes on two portions of their DOB, and 10% made changes to all three portions.

**ID deception.** A police department uses several types of identification numbers, such as Social Security number (SSN) or FBI ID number if one is on record. Most suspects, excluding illegal aliens, are expected to have a SSN. Therefore, we only looked into SSN records in our sample data and found 58.3% of the suspects used a falsified SSN. Also, in the falsified SSNs, 96% had no more than two digits different from the corresponding correct ones, for example, “123-45-6789” may be falsified as “123-46-6789” or “123-46-9789.” We found it rare for criminals to deceive by giving a totally different SSN. In our sample data, only one suspect used a deceptive SSN completely different from his real one.

It is possible for a suspect to forget his or her SSN and unintentionally give an incorrect SSN. It is important to note that giving a false SSN does not automatically flag a deceptive record. It just tells police officers to investigate further. When we compared SSNs between two records, we examined other fields as well. If the SSN was the only altered field in a comparison, the person who reported those two records may have simply forgotten his or her number. This was not considered as deception in our case study and we only considered ID deceptions that were accompanied by deception in other fields.

**Residency deception.** Suspects usually made changes to only one portion of the full address; street numbers and street types were typically altered. In our sample, 33.3% of the criminals had changed one portion of the address. Deception in more than one component of the address was not found.

## Deception Detection Algorithm Design and Experimental Results

To detect the deceptions identified in the taxonomy,

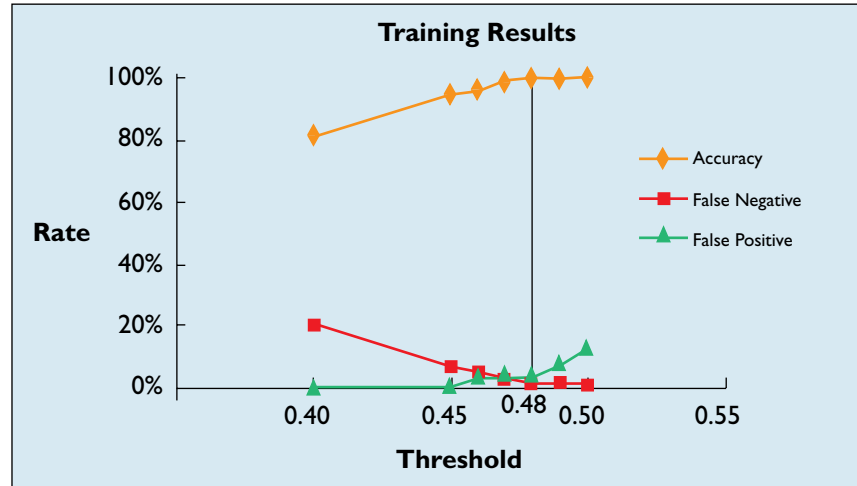


Figure 2. Training accuracy comparison based on different threshold values.

we chose the four most significant fields (name, DOB, SSN, and address) for our analysis.

The idea was to compare each corresponding field of every pair of records. Disagreement values for each field were summed up to represent an overall disagreement value between two records.

As previously discussed, we used a combination of edit distance and Soundex string comparators. To detect both spelling and phonetic variations between two name strings, edit distance and Soundex disagreement values were computed separately. In order to capture name exchange deception, disagreement values were also computed based on different sequences of first name and last name. We took the disagreement value from the sequence that had the least difference (the minimum disagreement value) between two names. Edit distance itself was used to compare nonphonetic fields of DOB, SSN, and address. Each disagreement value normalized between 0 and 1. The disagreement value over all four fields was calculated by a normalized Euclidean distance function. According to our expert police detective, each

field may have equal importance for identifying a suspect. Therefore, we started by assigning equal weights to each field.

**Experiment data collection.** In order to test the

Threshold	Accuracy	False Negative*	False Positive**
0.4	76.60%	23.40%	0.00%
0.45	92.20%	7.80%	0.00%
0.46	93.50%	6.50%	2.60%
0.47	96.10%	3.90%	2.60%
0.48	97.40%	2.60%	2.60%
0.49	97.40%	2.60%	6.50%
0.5	97.40%	2.60%	11.70%

\* False negative: consider dissimilar records as similar ones

\*\* False positive: consider similar records as dissimilar ones

Table 2. Accuracy comparison based on different threshold values.

Threshold	Accuracy	False Negative	False Positive
0.48	94.0%	6.0%	0.0%

Table 3. The accuracy of linkage in the testing data set.

feasibility of our algorithm, a sample set of data records with identified deception was chosen from the police database. At the time, we were not considering records with missing fields. Therefore, we drew from police profiles another set of 120 deceptive criminal identity records with complete information in the four fields. Our veteran Tucson police detective verified that all the records had deception information. The 120 records involved 44 criminals, each of whom had an average of three records in the sample set. Some data was used to train and test our algorithm so that records pointing to the same suspect could be associated with each other.

Training and testing were validated by a standard hold-out sampling method. Of the 120 records in the test bed, 80 were used for training the algorithm, while the remaining 40 were used for testing purposes.

**Training results.** A disagreement matrix was built based upon the disagreement value between each pair of records. Using the disagreement values in the matrix, threshold values were tested to distinguish between the in-agreement pairs of records and the disagreement pairs. Accuracy rates for correctly recognizing agreeing pairs of records using different threshold values are shown in Table 2. When the threshold value was set to 0.48, our algorithm achieved its highest accuracy of 97.4%, with relatively small false negative and false positive rates, both of which were 2.6% (see Figure 2).

**Testing results.** Similarly, a disagreement matrix was built for the 40 testing records by comparing every pair of records. By applying the optimal threshold value 0.48 to the testing disagreement matrix, records having a disagreement value of less than 0.48 were considered to be pointing to the same suspect and were associated together. The accuracy of linkage in the testing data set is shown in Table 3. The result shows the algorithm is effective (with an accuracy level of 94%) in linking deceptive records pointing to the same suspect.

## Conclusion

We have presented a record-linkage method based on string comparators to associate different deceptive criminal identity records. The experimental results have shown the method to be promising. The testing results also show that no false positive errors (recognizing related records as unrelated suspects) occurred, which means the algorithm has captured all deceptive cases. On the other hand, all the errors occurred in the false negative category, in which unrelated suspects were recognized as being related. In that case, different people could mistakenly be

considered the same suspect. This might be caused by the overall threshold value gained from the training process. The threshold value was set to capture as many true similar records as possible, nonetheless, a few marginal dissimilar pairs of records were counted as being similar. Currently, an investigator-guided verification process is needed to alleviate such a problem. An adaptive threshold might be more desirable for making an automated process in future research.

The proposed automated deception detection system will also be incorporated into the ongoing COPLINK project [2] under development since 1997 at the University of Arizona's Artificial Intelligence Lab, in collaboration with the TPD and the Phoenix Police Department (PPD). It continues to be funded by the National Science Foundation's Digital Government Program. ■

## REFERENCES

1. Burgoon, J.K., Buller, D.B., Guerrero, L.K., Afifi, W., and Feldman, C. Interpersonal deception: XII. Information management dimensions underlying deceptive and truthful messages. *Communication Monographs* 63 (1996). 50–69.
2. Hauck, R.V., Atabakhsh, H., Ongvasith, P., Gupta, H., and Chen, H. Using COPLINK to analyze criminal-justice data. *IEEE Computer* (Mar. 2002).
3. Jaro, M.A. *UNIMATCH: A Record Linkage System: User's Manual*. Technical Report, U.S. Bureau of the Census, Washington, DC, 1976.
4. Levenshtein, V.L. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady* 10, (1966), 707–710.
5. Newcombe, H.B., et al. Automatic linkage of vital records. *Science* 130, 3381 (1959), 954–959.
6. Porter, E.H. and Winkler, W.E. Approximate string comparison and its effect on an advanced record linkage system. *Record Linkage Techniques* (1997), 190–202.
7. Vrij, A. *Detecting Lies and Deceit: The Psychology of Lying and the Implication for Professional Practice*. John Wiley, 2000.
8. Winkler, W.E.. The state of record linkage and current research problems. In *Proceedings of the Section on Survey Methods of the Statistical Society of Canada*, 1999. (Also in technical report, RR99/04. U.S. Census Bureau; [www.census.gov/srd/papers/pdf/rr99-04.pdf](http://www.census.gov/srd/papers/pdf/rr99-04.pdf).)

**GANG WANG** ([gang@bpa.arizona.edu](mailto:gang@bpa.arizona.edu)) is a doctoral student in the Department of Management Information Systems, The University of Arizona, Tucson.

**HSINCHUN CHEN** ([hchen@bpa.arizona.edu](mailto:hchen@bpa.arizona.edu)) is McClelland Endowed Professor in the Department of Management Information Systems, The University of Arizona, Tucson.

**HOMA ATABAKHSH** ([homa@bpa.arizona.edu](mailto:homa@bpa.arizona.edu)) is a principal research specialist in the Department of Management Information Systems, The University of Arizona, Tucson.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

The path to technology-neutral policy is fraught with danger as more legislation is updated to deal with new communications infrastructures.

# QUESTIONING LAWFUL ACCESS TO TRAFFIC DATA

< By Alberto Escudero-Pascual and Ian Hosein

After some successes and many missteps, the regulatory environment surrounding technology policy is transforming. Lessons taken from content, copyright, and cryptography policy processes, among others, have resulted in the emergence of technology policy innovations. Two particular innovations are the *internationalization* of policy-making, and *technology-neutral* policies. However, these concepts come with risks that are particularly apparent when we look at policies on lawful access to *traffic data*.

Access to traffic data for law enforcement purposes is a traditional tool for investigation and intelligence gathering. Traffic data is an elusive term, due in part to technology variances. The policies regarding lawful access to traffic data, however, are increasingly set in technology-neutral language, while the language and policies are often negotiated at closed international fora.

Even while policy changes are argued as necessary due to international obligations and new technological realities, these policies tend to ignore technological details. Just as cryptography policies of key escrow were misinterpreted by government as updates “to maintain the

status quo” [8]; updating legal definitions of traffic data while not acknowledging the increased sensitivity of the data is problematic.

In the days of plain old telephone systems (POTS), the content of communications was considered sensitive and therefore any interception required constraint, for example, judicial warrants in the U.S. and politician-authorized warrants in the U.K. The same rule did not apply to traffic data: numbers called, calling numbers, and so on. This data was considered less invasive, and therefore only required minimal constraint. An additional factor was that traffic data was stored by telephone

---

ILLUSTRATION BY PAUL ZWOLAK

The policies regarding lawful access to  
traffic data, however, are increasingly set in  
TECHNOLOGY-NEUTRAL LANGUAGE, while  
the language and policies are often negotiated at  
CLOSED INTERNATIONAL FORA.

---

companies and in turn was available for access by law enforcement agencies, while content was not: traffic data was available, legally less sensitive, and so, lawfully accessible.

The traffic data records collected by telephone companies are generally of a form similar to:

19991003070824 178 165 0187611205 46732112106-----001----  
003sth 46 4673000----0013 14 10260

The most significant fields are date and time of the start of the call, duration of the call, and caller and receiver phone numbers.

Traditional investigative powers of access to traffic data were established with traditional technological environments in mind. Governments are now updating these policies to apply to modern communications infrastructures. If governments insist on applying traditional powers to these new infrastructures, the new policies must acknowledge that the data being collected now is separate from tradition.

Many policy initiatives have involved articulations regarding the importance of being technology neutral. When the Clinton Administration first announced its intention to update lawful access powers, it proposed to “update the statutes in outmoded language that are hardware specific so that they are technologically neutral” [7]. Meanwhile, as noted by the Earl of Northesk in tempestuous debates in the U.K.’s House of Lords regarding the Regulation of Investigatory Powers Act 2000: “One of the many difficulties I have with the Bill is that, in its strident efforts to be technology neutral, it often conveys the impression that either it is ignorant of the way in which current technology operates, or pretends that there is no technology at all” [9].

Technology-neutral policy is seen as a way to deal with concerns of governments mandating a specific type of technology. While this is favorable in the case

of some policies that affect market developments, technology-neutral lawful access policies may contain hazardous side effects. Indeed, technology-neutral language may be used to ignore the challenges and risks to applying powers to different infrastructures.

### Defining Traffic Data

Intergovernmental organizations, particularly the Group of 8 (G8) and the Council of Europe (CoE), have been working for a number of years to ensure lawful access to traffic data. Both the G8 and the CoE have been criticized by industry and civil society because of their ambiguous and problematic approaches, and the closed nature of their processes. Yet policies continue to be decided there, and brought to national parliaments under the guises of *harmonization* and *international obligations*.

The G8 formed a senior experts group in 1995 to develop an international cooperation regime to address transnational organized crime. This Lyon Group has since been active on high-tech surveillance-related policies, including three meetings with industry representatives throughout 2000 and 2001. Arising from that work, the G8 working-definition of traffic data is “non-content information recorded by network equipment concerning a specific communication or set of communications” [10].

The CoE, a 45-member intergovernmental organization, has convened closed meetings since 1997 to develop a treaty establishing lawful access powers across borders. The CoE Convention on Cybercrime defines traffic data as “any computer data relating to a communication by means of a computer system, generated by a computer system that formed a part in the chain of communication, indicating the communication’s origin, destination, route, time, date, size, duration, or type of underlying service” [2]. In the convention’s Explanatory Report [3], the CoE states that traffic data should be defined so as “to not refer”



to the content of a communication; but this is a non-binding interpretation.

One is left to wonder what is included and what is excluded by these vague definitions. While subject lines in email may be content, uncertainty arises as to whether the name of files requested (for example, HTTP requests), URLs (for example, [www.computer.tld](http://www.computer.tld)), search parameters, TCP, or IP headers, and other such data is considered content or traffic data. A report of a transaction by an individual with server 158.143.95.65 may be considered traffic data; but the name of the Web site(s) run on a server may disclose more information (for example, [aidshelpline.org](http://aidshelpline.org)). Search parameters in the URLs and the name of files accessed may refer to the content of the communications. If we consider the next-generation Internet, mobility bindings or routing information included in the IPv6 extended header will include absolute or relative location information. The location information is part of the mobility signaling protocol and hence fits into the definitions of traffic data discussed here.

Some states have tried to deal with this challenge in their legislative language. The U.K.'s Regulation of Investigatory Powers Act 2000 went through much iteration, particularly in the so-called "Big Browser" debates, before the language was agreed upon. Traffic data is defined in theory as data about the source and destination of a transaction, and data about the routing and the tying of separate packets together. This definition is complemented by the legal definition of *communications data*, that is, data used by the network; or exists within logs; or other data collected by service providers. However the definitions are also quite clear about the extent of information that qualifies: this data does not include URLs per se, and may only include the name of the computers running a service, while the specific resource used qualifies as content, and accorded greater protection. Therefore, the IP address in the U.K. is traffic data, while [www.url.tld/file.html](http://www.url.tld/file.html) is tantamount to content.

Other states have failed to respect this level of technological awareness. Previous U.S. policy differentiated between traffic data from cable and telephone communications. The Cable Act once protected traffic data to a greater degree than telephone traffic data, as viewing habits were considered sensitive. Now that cable infrastructure is also used for Internet communications (which were previously used over telephone lines, and thus traditional laws applied), successive White House administrations worked to erase this cable traffic distinction, finally succeeding with the USA Patriot Act. Rather than deal with the specifics of digital communications

media and services, the changes in U.S. law reduce the protections of traffic data for cable Internet communications to what had previously existed for telephone communications data.

This can be interpreted as a boon to law enforcement. According to U.S. Attorney General John Ashcroft: "Agents will be directed to take advantage of new, technologically neutral standards for intelligence gathering. ...Investigators will be directed to pursue aggressively terrorists on the Internet. New authority in the legislation permits the use of devices that capture senders and receivers addresses associated with communications on the Internet" [1].

Traffic data blurs with the content of communications as new communications infrastructures are encompassed under existing practices. The legal protection of this data is reduced as distinctions applied are based on categorical decisions established under older technologies. The separation of content and traffic remains elusive, even in policy language.

### Categorical Determinants

Traffic data under the POTS as considered derivative, and while informative, it did not necessarily disclose the sensitive details of an individual's life. Under the Cable Act, Congress accepted that the disclosure of individual viewing habits deserved greater protection; but such protections were later deemed unnecessary for the Internet.

Traffic data's constitution differs by communications medium. Here, we present dial-in records, wireless LANs, and search engines to preview what can be accessed by technology-neutral law enforcement powers.

*Dial-in records.* The Remote Authentication Dial-In User Service (RADIUS) is a client/server security protocol, designed to manage dispersed modem pools for large numbers of users. This tends to involve managing a single database of users, allowing for authentication (verifying user name and password) as well as configuration information detailing the type of service to deliver.

Many ISPs are outsourcing the access network to big operators that provide dial-up connectivity worldwide. Internet users dial into a modem pool attached to a Network Access Server (NAS) that operates as a client of RADIUS. The client is responsible for passing user information to designated RADIUS servers (managed by the ISP) and then acting on the response returned.

The RADIUS server stores usage information for dial-in users, often for billing purposes. When the user is authenticated and the session has been configured according to the authorization information, an

## The landscape for lawful access powers REMAINS QUITE FRAGMENTED.

accounting start record is created. When the user's session is terminated, an accounting stop record is created.

The most significant fields of the start/stop records are:

### Start and Stop Timestamps

Timestamp records the "Start" and "Stop" time on the RADIUS accounting host. The duration of a session "Acct-Session-Time" is computed by subtracting the "Start" and "Stop" timestamps.

### Call(ed,ing)-Station-Id

Called-Station-Id records the telephone number called by the user.

Calling-Station-Id records the number the user is calling from.

Figure 1 depicts part of the start and stop RADIUS records.

From this log we can extract a limited amount of information regarding the content of the communications transactions that took place. The user has been identified (aep@somedomain.org), the number of the caller (01223555111, which is a Cambridge number) and the location being called (02075551000, London), IP address assigned (62.188.17.227), the duration (21s), number of bytes and packets sent and received, type of connection, date and time. The traffic data over time identifies the change in location of a user despite the common dialed number. As users roam globally with different access telephone numbers, the user identification remains static. In this sense, the collected traffic data is mildly more

sensitive than traditional telephone data: where POTS traffic data pivots around a given telephone/ID number, RADIUS data pivots around a user ID regardless of location; therefore disclosing location shifts. The ISP now knows everywhere their customers connect from; information that may be useful to other parties.

*Wireless LAN association records.* Such mobility becomes more problematic within wireless environments. In a standard wireless LAN environment using IEEE 802.11b, a radio cell size can vary from hundreds of meters in open air, to a small airport lounge. Before the mobile station (STA) is allowed to send a data message via an access point (AP), it must first become associated with the AP. The STA learns what APs are present and then sends a request to establish an association.

The significant records of a centralized association system log are illustrated in Figure 2 and include:

### time\_GMT

Time when a mobile node associates with a base station

### Cell\_ID

Base station unique identifier in the LAN

### MAC\_ID

A unique identifier of a mobile device.

In our analysis of collected logs we identified moments where two individuals were alone within a cell, and whether they arrived together. It is tempting to analyze these logs by drawing an analogy with the POTS, that is, a registration of a mobile

node with an access point could be seen as the establishment of a phone call between both parties. This analogy is simplistic as it doesn't consider the Cell\_IDs represent places (airport, conference room,

```
Fri Oct 19 11:30:40 2001
User-Name = "aep@somedomain.org"
NAS-IP-Address = 62.188.74.4
Acct-Status-Type = Start
Acct-Session-Id = "324546354"
Acct-Authentic = RADIUS
Calling-Station-Id = "01223555111"
Called-Station-Id = "02075551000"
Framed-Protocol = PPP
Framed-IP-Address = 62.188.17.227

Fri Oct 19 11:31:00 2001
User-Name = "aep@somedomain.org"
NAS-IP-Address = 62.188.74.4
Acct-Status-Type = Stop
Acct-Session-Id = "324546354"
Acct-Authentic = RADIUS
Acct-Session-Time = 21
Acct-Input-Octets = 11567
Acct-Output-Octets = 3115
Acct-Input-Packets = 96
Acct-Output-Packets = 74
Calling-Station-Id = "01223555111"
Called-Station-Id = "02075551000"
Framed-Protocol = PPP
Framed-IP-Address = 62.188.17.227
```

Figure 1. RADIUS  
start and stop  
records.

restaurants) and the registration timestamps can reveal if two nodes are (moving) together. Data mining of association records (registration and deregistration) can provide sufficient information to draw a map of human relationships [4].

*HTTP requests to a search engine.* The media discussed here may involve further traffic data in the form of Internet protocols. The GET and POST methods in the Hypertext Transfer Protocol (HTTP) allow a Web client to interact with a remote server. In the most common search engines, the keywords are included in the HTTP header as part of a GET method. All the Web logs can be transformed to a W3C common log file format that contains the IP address of the client, the connection time, the object requested and its size (see Figure 3).

If traffic data residing in logs is analyzed, a great deal of intelligence can be derived. Observing the logs we can see for example, that 212.164.33.3 has requested (in a short period of time) information about “railway+info+London” and “union+ strike” in two different requests. We may identify not only the patterns of an individual’s movements online, but also interpret an individual’s intentions and plans. Or more dangerously, one could derive false intentions (“child+pornography” may be a search for studies on the effects of pornography on children). Much more can be ascertained with some data mining, even if IP addresses are assigned dynamically, allowing for traceability based on habits and interests; and compounded with location data, previous NAS data, a comprehensive profile can be developed.

## The Shape of Things

Even the COE acknowledges, in passing, that the breadth of possible traffic data may be problematic. “The collection of this data may, in some situations, permit the compilation of a profile of a person’s interests, associates and social context. Accordingly Parties should bear such considerations in mind when establishing the appropriate safeguards and legal prerequi-

sites for undertaking such measures” [3].

However, no such safeguards or prerequisites are mandated nor discussed in detail. The convention text is mute on this matter.

Shifting between infrastructures gives different data; but converging infrastructures is even more worrisome. Mobile communications systems magnify the sensitivity of traffic data; wireless LANs were presented as an indication of the shape of things to come as we encounter new protocols and infrastructures, for example, third-generation wireless systems running IPv6.

Yet governments want access to this information. The collection and access methods currently under consideration are *preservation* (access to specified data of a specific user collected by

```
Time_GMT=20010810010852 Cell_ID=115 MAC_ID=00:02:2D:20:47:24
time_GMT=20010810010852 Cell_ID=115 MAC_ID=00:02:2D:04:29:30
[...]
time_GMT=20010810010854 Cell_ID=129 MAC_ID=00:02:2D:04:29:30
time_GMT=20010810010854 Cell_ID=129 MAC_ID=00:02:2D:20:47:24
[...]
time_GMT=20010810010856 Cell_ID=41 MAC_ID=00:02:2D:04:29:30
time_GMT=20010810010856 Cell_ID=41 MAC_ID=00:02:2D:20:47:24
[...]
time_GMT=20010810010900 Cell_ID=154 MAC_ID=00:02:2D:20:47:24
time_GMT=20010810010900 Cell_ID=154 MAC_ID=00:02:2D:04:29:30
```

Figure 2. Wireless LAN data logs.

```
295.47.63.8 - - [05/Mar/2002:15:19:34 +0000]
"GET /cgi-bin/htsearch?config=htdig&words=startrek HTTP/1.0" 200 2225
295.47.63.8 - - [05/Mar/2002:15:19:44 +0000]
"GET /cgi-bin/htsearch?config=htdig&words=startrek+avi HTTP/1.0" 200 2225
215.59.193.32 - - [05/Mar/2002:15:20:17 +0000]
"GET /cgi-bin/htsearch?config=htdig&words=Modem+HOWTO HTTP/1.1" 200 2045
192.77.63.8 - - [05/Mar/2002:15:20:35 +0000]
"GET /cgi-bin/htsearch?config=htdig&words=conflict+war HTTP/1.0" 200 2225
211.164.33.3 - - [05/Mar/2002:15:21:32 +0000]
"GET /cgi-bin/htsearch?config=htdig&words=railway+info HTTP/1.0" 200 2453
211.164.33.3 - - [05/Mar/2002:15:21:38 +0000]
"GET /cgi-bin/htsearch?config=htdig&words=tickets HTTP/1.0" 200 2453
211.164.33.3 - - [05/Mar/2002:15:22:05 +0000]
"GET /cgi-bin/htsearch?config=htdig&words=railway+info+London HTTP/1.0" 200 8341
212.164.33.3 - - [05/Mar/2002:15:22:35 +0000]
"GET /cgi-bin/htsearch?config=htdig&words=union+strike HTTP/1.0" 200 2009
82.24.237.98 - - [05/Mar/2002:15:25:29 +0000]
"GET /cgi-bin/htsearch?config=htdig&words=blind+date HTTP/1.0" 200 2024
```

Figure 3. Sample search engine traffic data.

service providers for business purposes), *retention* (requiring all logs for all users be stored beyond their business purpose), and *real-time* (government access to real-time data flows). These powers are enshrined in G8 and CoE agreements; and will be appearing in national laws near you, if they are not already there.

The national laws that enshrine these collection and access powers differ remarkably, despite being established under the umbrella/guidance of the G8 and the CoE. Neither organization places requirements on countries to require safeguards such as limitation on access, or specified purposes for collection, and authorizations through judicial warrants. We believe that national deliberation on these matters will be limited because governments will claim, as they already have, that the policy changes are required due to international obligations, or worse, technological imperatives. These international policy

dynamics thus reduce deliberation and our ability to inform policy discourse.

The landscape for lawful access powers remains quite fragmented. U.K. law separates URLs from traffic data, and yet practices very weak access constraints (any number of government agencies may access this data); and later proposed retention regimes for periods ranging from four days (Web cache), six months (RADIUS, SMTP, and IP logs), and seven years [5]. The U.S. recently introduced technology neutrality to its laws thus reducing earlier protections, but the U.S. has stronger access protections than the U.K.; and has no retention requirements.

With over 30 signatory states including the U.S., Canada, Japan, Romania, France, and Croatia, we can rest assured that there will be selective interpretation in implementation of the CoE cybercrime convention. Even among the G8 countries, the protections afforded to citizens' communications in Italy, Germany, the U.S., and Russia vary greatly.

While policies may vary, the sensitive nature of the data produced does not. Traffic data analysis generates more sensitive profiles of an individual's actions and intentions, arguably more so than communications content. In a communication with another individual, we say what we choose to share; in a transaction with another device, for example, search engines and cell stations, we are disclosing our actions, movements, and intentions. Technology-neutral policies continue to regard this transactional data as POTS traffic data, and accordingly apply inadequate protections.

The Canadian government, claiming its intention to ratify the CoE convention, has already proposed to consider all telecommunications services as equivalent. "The standard for Internet traffic data should be more in line with that required for telephone records and dial number recorders in light of the lower expectation of privacy in a telephone number or Internet address, as opposed to the content of a communication" [6]. This is disingenuous: the civil servants who wrote this policy are the same as those who participated in drafting the CoE and the G8 agreements. Select components of these agreements are thus brought home for ratification while the recommended protections are not.

This is not faithful to the spirit of updating laws for new technology. We need to acknowledge that changing technological environments transform the policy itself. New policies need to reflect the totality of the new environment.

These technology policy innovations fail to do so. Governments seek technology-neutral policy, and are

doing so at the international level. This appears to be to the advantage of policy-setters. New powers are granted through technological ambiguity rather than clear debate. International instruments, such as those from the Group of 8 and the Council of Europe, harmonize language in a closed way with little input and debate. These problems will grow as more countries feel compelled to ratify and adopt these instruments; or feel it is in their interests to do so. Implementation, however, will still be fragmented and will likely be in the interests of increasing access powers of the state.

Attempts to innovate policy must be interrogated, lest we reduce democratic protections blindly. **C**

## REFERENCES

1. Ashcroft, J. Testimony of the Attorney General to the Senate Committee on the Judiciary. Washington D.C. Sept. 25, 2001.
2. Council of Europe. Convention on Cybercrime, ETS No.185.
3. Council of Europe. Convention on Cybercrime Explanatory Report (adopted Nov. 8, 2001).
4. Escudero A. Location data and traffic data. Contribution to the EU Forum on cybercrime. (Brussels, Nov. 2001).
5. Gaspar, R. Looking to the future: Clarity on communications data retention law: A national criminal intelligence service. Submitted to the Home Office for Legislation on Data Retention on behalf of ACPO and ACPO(S); HM Customs & Excise; Security Service; Secret Intelligence Service; and GCHQ (Aug. 2000).
6. Government of Canada. Lawful access—Consultation document. Department of Justice, Industry Canada, Solicitor General Canada (Aug. 25, 2002).
7. Podesta, J. National Press Club Speech. (Washington D.C. July 17, 2000).
8. Reno, J. Law enforcement in cyberspace. Address presented to the Commonwealth Club of California, San Francisco, 1996.
9. U.K. Hansard. House of Lords (Committee Stage). Column 1012, June 28, 2000. The Stationery Office Ltd.
10. U.S. Delegation. Discussion Paper for Data Preservation Workshop. G8 Conference on High-Tech Crime. (Tokyo, May 22–24 2001).

---

The data presented here has been obtained with permission from a telephone carrier, an Internet service provider, and a large conference where wireless LAN access was provided. All transactions presented here have been de-identified, and the time logs were altered to reduce the risk of reidentification

---

**IAN (GUS) HOSEIN** (gus@privacy.org) is a fellow in the Department of Information Systems at the London School of Economics; and a fellow at Privacy International; is.lse.ac.uk/staff/hosein.

**ALBERTO ESCUDERO PASCUAL** (aep@it.kth.se) is an assistant professor in the Department of Microelectronics and Information Technology at the Royal Institute of Technology (KTH) in the area of privacy in the next-generation Internet; www.imit.kth.se/~aep/.

---

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

---



By Nikunj P. Dalal, Manjunath Kamath,  
William J. Kolarik, and Eswar Sivaraman

# *Toward an* Integrated Framework *for Modeling* *Enterprise Processes*

Enterprise process modeling is the most important element in the design of next-generation ERP systems.

**I**magine that a large company has spent hundreds of millions of dollars and three years implementing an enterprisewide information system. But when the system goes live, the company discovers the system is incapable of supporting the volume and price structure of its distribution business.

The project fails, and the company suffers monumental losses and is ultimately driven into bankruptcy. The company sues the vendor of the software package, blaming it for its losses. A hypothetical situation? Hardly. This disaster story is as true as it is regrettable and avoidable [4].

Since the mid-1990s, many large- and mid-size enterprises have implemented off-the-shelf enterprise software packages (also called enterprise resource planning, or ERP, systems) to integrate their business activities, including human resource management, sales, marketing, distribution/logistics, manufacturing, and accounting. Enterprise systems promise not only information integration but the benefits of reengineered and radically

improved business processes as well. "The business world's embrace of enterprise systems," according to [4], "may in fact be the most important development in the corporate use of information technology in the 1990s," an assessment that's just as valid today. However, despite a few dramatic successes, many companies still reportedly fail to realize these benefits while incurring huge cost and schedule overruns.



Given these mixed results, what should companies do? Rejecting enterprise software as an enterprise-integration solution altogether would be foolhardy, given the multimillion-dollar investments many companies have made in the software and the partial success many of them have realized. Moreover, next-generation enterprise software from such vendors as SAP, Oracle, and PeopleSoft is evolving rapidly, promising to improve flexibility, implementation, and support for the extended enterprise through modules for customer relationship management, advanced planning systems, supply chain management, and collaborative commerce in a Web-based environment. Consulting firm Gartner Group estimates that by 2005, this next generation of ERP it calls ERP II [11] will replace current ERP systems, thus requiring companies to upgrade. Enterprise process modeling is crucial to the design of ERP II systems.

A common source of difficulty implementing enterprise software involves management's understanding of its own business processes [6]. A business process, which is different from a traditional business function, is typically cross-functional and involves the reciprocal or simultaneous flow of information between two or more functional areas, as well as among the functions within these areas. For example, the order-fulfillment process involves inputs from sales, logistics, manufacturing, and finance, as it progresses from sales order entry, to delivery of the product, to the final step of collecting cash payment from customers. Business processes, including order-fulfillment, procurement, and product development, hold the key to the financial success of an enterprise. In theory, an enterprise system is ready to support business processes because it encapsulates best business practices, or the tried and successful approaches to implementing business processes, and is hence the ideal vehicle for delivering the benefits of an integrated cross-functional approach.

However, "As many companies get ready to implement standard software, they encounter the problem of how to simplify and model the enormous complexity of their business processes" [6]. The result is that companies often face the dilemma of whether to adapt to the software and radically change their business practices or modify the software to suit their specific needs.

Even if they decide to modify the software, they still face maintenance and integration issues. Many enterprise systems today are notably inflexible with respect to process specification and implementation [4]. Moreover, the packages are difficult to change and extend due to their complex proprietary application program interfaces and database schemata—a far cry from proposed open standards of e-commerce [8]. Even if a company were to overcome this barrier, modify its soft-

ware, and painstakingly build complex interfaces with other information systems, its maintenance and integration issues would still not be completely resolved. The modification trauma is reexperienced every time the enterprise software vendor issues a new release of its software. To be sure, leading ERP vendors are working to resolve these issues, though much remains to be done to realize the ERP II vision.

A holistic solution approach garnering considerable researcher attention calls for renewed focus on enterprise process models instead of on technologies alone. It envisions enterprises having the flexibility to redesign enterprise processes—regardless of whether the new processes are derived from clean-sheet process reengineering unhindered by technological considerations or whether industry-standard best practices are incorporated into the software. From this perspective, process models that are easily created, modified, and analyzed greatly aid process-reengineering efforts to realize the promised benefits of enterprise systems. Hence, researchers and managers are increasingly interested in techniques, existing and new, for business-process modeling, specification, implementation, maintenance, and performance improvement [9].

We've developed an enterprise process-modeling framework that can serve as the foundation for next-generation enterprise systems. Here, we outline some limitations of existing enterprise modeling techniques and architectures, describe a prototype implementation of the framework, and conclude with the significance of this work.

## Enterprise Process Modeling Techniques

Many techniques for modeling enterprise processes, including Data Flow Diagrams (DFDs), Integration Definition for Function Modeling (IDEF0), and activity diagrams in the Unified Modeling Language, have their roots in process modeling for software development. In 1992, [3] reported "Process modeling work is still young, and the span of the research agenda is still being formulated. Nevertheless, work to date holds promise for benefits in management, process-driven environments, and process reengineering." However, this promise has been only partially realized with the evolution of a number of techniques, architectures, and frameworks focusing on modeling the enterprise in general and business processes in particular. Such approaches focus on modeling the enterprise in order to reengineer or redesign business processes with the help of information technology. In a broader context, the technology itself is but a part of a greater whole, including the enterprise, the supply chain, and entire groups of related industries. Techniques include the Computer Integrated Manufacturing Open System

Architecture business process modeling approach, the Integrated Enterprise Modeling approach, the Purdue Enterprise Reference Architecture, the predicate-logic-based Toronto Virtual Enterprise method, Baan's Dynamic Enterprise Modeling method, and SAP's adaptation of the Event-driven Process Chain method (part of the Architecture of Integrated Information Systems). Each provides a basic set of constructs to model enterprise functionality [5] but also involve four major gaps:

*Need for a theory base.* Existing process models are descriptive but lack prescriptive capabilities. That is, they do not provide business modelers or system architects a formal theoretical base from which business processes can be analyzed in a rigorous, quantitative manner. This gap is serious; formal analysis is essential for learning the effect of changes in process logic and parameters on business performance measures in the interests of making better business decisions. Moreover, an underlying formalism would help the enterprise system architect generate multiple, whole-process views of the enterprise at various levels of abstraction. Such a capability is essential for managing the enterprise.

*Need for modeling and implementing distributed computing.* Many existing process modeling techniques do not explicitly incorporate the distributed computing paradigm of the Internet and lack the syntax and semantics necessary for modeling the distributed enterprise and for designing and implementing the process model in an Internet-based environment. Still needed are modeling techniques compatible with the distributed infrastructure of the Internet. Companies using mid- and large-scale ERP systems are dispersed geographically, making it imperative that their users be able to collaborate in creating, modifying, and analyzing process models from any location at any time. The need for distributed access to process models is even greater for next-generation extended enterprises and virtual organizations. The process models in the enterprise software toolkit have not kept pace with other developments in the computing paradigm.

*Need for new process redesign semantics.* Many process modeling techniques, especially those originally designed for software development, including DFDs, are general-purpose by design. As a result, they lack explicit semantics for enterprise-oriented concepts like cost and time. While enterprise modeling architectures

and workflow software for process redesign allow for these concepts, they are not generally tied to enterprise software. Moreover, information on cost drivers and process performance measures, including time, quality, and efficiency, are not readily captured in existing enterprise modeling systems. The challenge for the architect is to create a simple and usable process-modeling technique that also represents enterprise-oriented semantics.

*Need to link business and engineering processes.* Conventional wisdom points to the fact that business results are tied to physical processes, whereby resources are converted to products satisfying market demand. However, contemporary process modeling approaches do not adequately reflect the

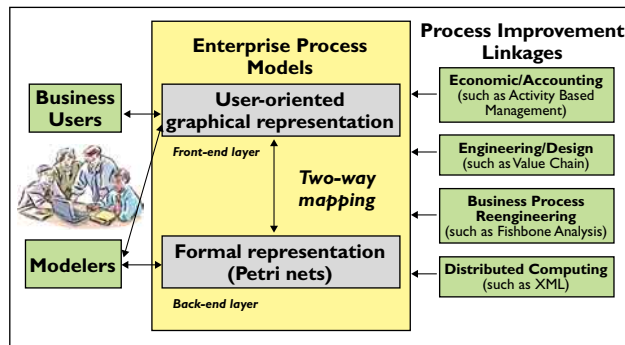


Figure 1. Conceptual model of the framework.

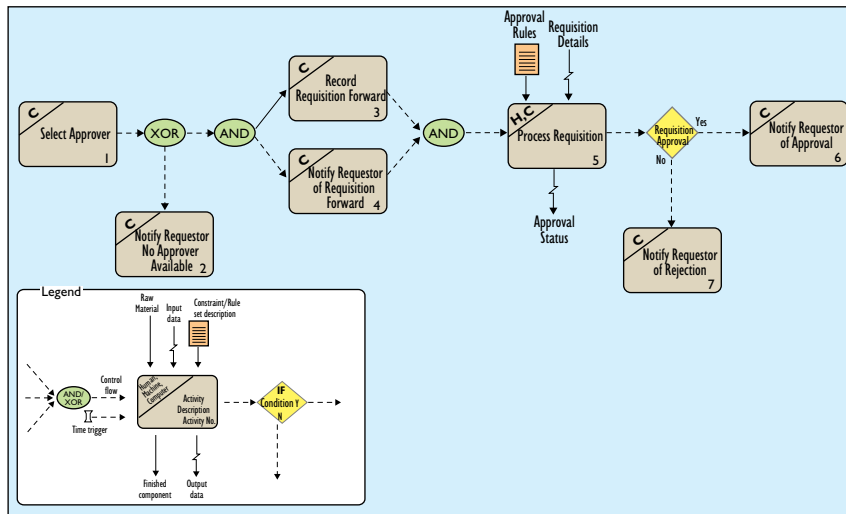
interrelationships between business and production engineering. Engineering approaches generally focus on physical conversion at one end of the process spectrum, while business approaches focus on market and financial strategies at the other end. To be effective, process design, control, and improvement demand the use of modeling methods with scalable and dynamic properties providing seamless links between business and technical process issues.

## Holistic Management

Together these four needs support the case for an integrated process modeling framework. Our framework thus takes an interdisciplinary approach, with inputs from information systems, accounting, computer science, industrial engineering, and business disciplines (see Figure 1).

Any modeling approach must keep the user in the loop. The strengths of popular process modeling techniques, including DFDs and IDEF0, reflect their simplicity; even novice end users readily understand the associated graphical symbols and terminology. The framework emphasizes business users and specialized modelers who create, modify, analyze, and use enterprise process models.

These models have at least two layers: front-end graphical and back-end formal. A theoretical base is established by well-defined mappings between the user-oriented graphical model at the front end and its corresponding formal representation at the back end. The mapping is two-way; a formal representation can be generated from a user's graphical model and vice versa. Note that the mappings are more than translations. Analysis performed at the back end may provide inputs



**Figure 2. EPML model of requisition processing.**

to modify the front-end graphical model and vice versa. The framework entails a process modeling language incorporating various process improvement methods, as shown in Figure 1.

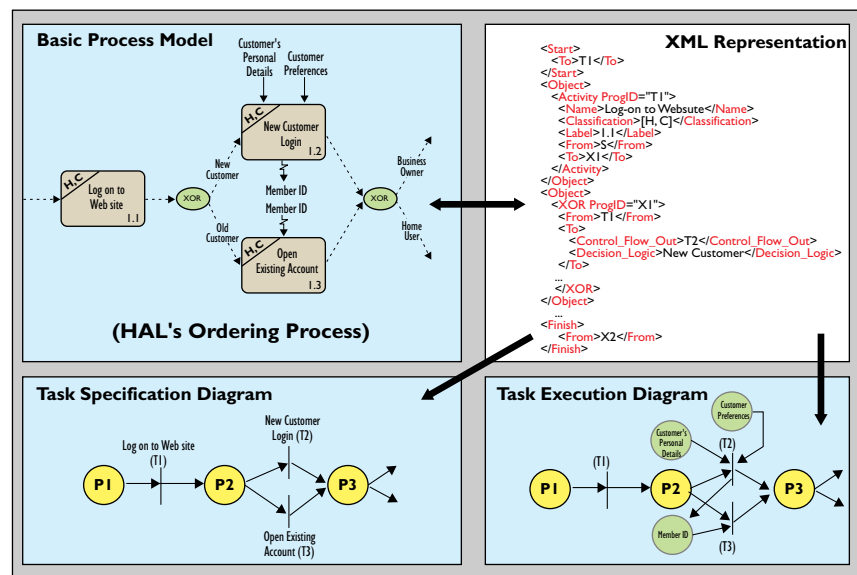
**Implementation.** We demonstrated the framework's feasibility with a proof-of-concept implementation at the Center of Computer Integrated Manufacturing at Oklahoma State University with four components: a graphical process modeling technique; Petri net theory providing a formal theory base; XML for mapping the front-end graphical layer to the formal Petri net layer; and a Web-based software prototype.

The front-end modeling layer is based on our newly developed graphical process modeling language—the Enterprise Process Modeling Language, or EPML [1]—which builds on such existing process modeling techniques as DFDs, IDEF techniques, and SAP's Event-driven Process Chain technique. Figure 2 outlines a model created in EPML involving approval of purchase requisitions. A requisition is first assigned to an approver; if no approver is available, the requestor is prompted to take appropriate action. If an approver is available, the requestor is informed, and a record is maintained of the status of the requisition. Once the approver processes the requisition, the requestor is notified of the rejection or approval outcome.

The graphical models created through EPML are then formalized as Petri net models. Although Petri net theory is used here, the framework is general-purpose,

allowing for the coexistence of other formalisms as well. Petri nets provide a strong mathematical foundation for modeling and analyzing concurrency, choice, asynchronous completion [7], state transitions, and other aspects of business processes [2, 9]. Petri net-based analysis results in such quantitative summary measures as throughput and response time. Petri nets have also been used for modeling workflows and verifying the correctness of control flow in business processes [2, 9, 10].

The translation between the front-end EPML model and the back-end Petri net representation is achieved through an XML-based markup language; Figure 3 outlines this approach using a subset of a larger model we created for a representative next-generation enterprise involved in the direct selling of computers to end customers.



**Figure 3. The mapping process.**

Mapping a graphical process model to Petri net representation is achieved in two steps, as shown in the figure. First, the elements necessary for specifying flow of control in a business process—the tasks, their sequencing, and the logical transitions among them—are represented using a Petri net model that maintains nearly one-to-one correspondence with its equivalent EPML graphical model; we call this model the Task Specification Diagram, or TSD, which is used to verify the correctness of the control flow specification. Verification is essential for automated control and coordination of business processes, because the control flow drives the scheduling, sequencing, resource assignment, and allo-

cation needed to create the final product or service. Once correctness of control flow is established, it is necessary to ensure the correctness of the resource assignment policies and input-output specifications. This second step involves enrichment of the TSD with the

related areas must collaborate to develop effective tools, techniques, and methods for ERP II. Architectural issues underscored by this work include: building holistic process models that link business and technical parameters; integrating—semantically, logically, and physically—process submodels created by distributed users; linking descriptive models to underlying formal analytic models; and linking process models with the overall logic of enterprise systems. ■

Type of Analysis	Representative Analysis Issues	Sample Questions
<b>Design.</b> To verify the correctness of the flow of control in the process definition.	<p><b>1. Termination.</b> Can a process attain all terminal state(s)?</p> <p><b>2. Deadlock.</b> Can a process attain a non-terminal state from where the flow of control ceases, that is, is the process deadlocked?</p> <p><b>3. I-Boundedness.</b> Is the task specification diagram safe, or I-bounded?</p>	<p>Can the customer assemble all valid system configurations?</p> <p>Can the user's dialogue reach undesirable intermediate dead-end states from which there are no more options to proceed?</p> <p>Is it possible for the system to charge a customer twice?</p>
<b>Performance.</b> To evaluate the execution correctness and run-time performance of the process.	<p><b>1. Resource assignment policies.</b> Can the assignment of resources to various tasks lead to a deadlocked state?</p> <p><b>2. Summary measures.</b> Can important business questions be addressed using summary measures derived from the task execution diagram (TED)? Metrics, including resource utilization levels, response times, and estimated costs, can be derived either from TEDs or from additional queuing/simulation models based on the TED.</p>	<p>Can the order-fulfillment process get stuck indefinitely in some intermediate process?</p> <p>What are the bottleneck resources? What is the average time to complete a customer order? What are the odds the customer order is delivered in two weeks?</p>

#### Representative Petri-net-based analysis.

details of each task's input, output, and resource requirements; we call this resulting Petri net representation a Task Execution Diagram, or TED. The table here includes sample analysis questions addressable through the TSD and the TED. The analysis questions relate to correctness of control flow (studied using the TSD) and execution correctness and run-time performance metrics (studied using the TED).

## Conclusion

Enterprise integration remains a challenging problem for many organizations. Enterprise systems, once viewed as a technological panacea for dealing with information fragmentation, have produced mixed results. Building better enterprise systems requires putting the enterprise back into enterprise systems [4], along with enterprise process modeling. The framework we developed addresses several modeling concerns relating to theory, distributed computing, process semantics, and links between business and engineering processes. With Petri net theory underlying the framework's graphical process models, managers and designers get both ease-of-use of graphical process modeling and the ability to perform rigorous quantitative and qualitative performance analysis. An XML-based markup language for mapping from the front end to the back end (and vice-versa) enables a standard layer for communicating with the existing systems of customers, partners, and suppliers.

Still needed is a comprehensive theoretical foundation to drive the design and construction of next-generation enterprise systems. Practitioners and researchers from computing, business management, engineering, and

1. Chaugule, A. *A User-Oriented Enterprise Process Modeling Language*. Masters Thesis, Oklahoma State University, Stillwater, OK, 2001.
2. Cichocki, A., Helal, A., Rusinkiewicz, M., and Woelk, D. *Workflow and Process Automation: Concepts and Technology*. Kluwer Academic Publishers, Boston, MA, 1998.
3. Curtis, B., Kellner, M., and Over, J. Process modeling. *Commun. ACM* 35, 9 (Sept. 1992), 75–90.
4. Davenport, T. Putting the enterprise into the enterprise system. *Harvard Bus. Rev.* (July-Aug. 1998), 121–131.
5. Kamath, M., Dalal, N., Chaugule, A., Sivaraman, E., and Kolarik, W. A review of enterprise process modeling techniques. In *Scalable Enterprise Systems: An Introduction to Recent Advances*, V. Prabhu, S. Kumara, and M. Kamath, Eds. Kluwer Academic Publishers, Boston, MA, 2003, 1–32.
6. Keller, G. and Detering, S. Process-oriented modeling and analysis of business processes using the R/3 reference model. In *Modeling and Methodologies for Enterprise Integration*, P. Bernus and L. Nemes, Eds. Chapman & Hall, London, 1996, 69–87.
7. Murata, T., 1989. Petri nets: Properties, analysis, and applications. *Proceedings of the IEEE* 77, 4 (Apr. 1989), 541–580.
8. Radding, A. *ERP, Componentization, and E-commerce*, 1999; see [itmanagement.earthweb.com/erp/print/0,,11981\\_616621,00.html](http://itmanagement.earthweb.com/erp/print/0,,11981_616621,00.html).
9. Salimifard, K. and Wright, M. Petri net-based modeling of workflow systems: An overview. *Europ. J. Operat. Res.* 134, 3 (2001), 664–676.
10. van der Aalst, W. and van Hee, K. *Workflow Management: Models, Methods, and Systems*. MIT Press, Cambridge, MA, 2002.
11. Zrimsek, B. *ERP II: The Boxed Set*. Gartner Group, Stamford, CT, Mar. 4, 2002; see [www3.gartner.com/pages/story.php?id.2376.s.8.jsp](http://www3.gartner.com/pages/story.php?id.2376.s.8.jsp).

**NIKUNJ P. DALAL** (nik@okstate.edu) is an associate professor of information systems in the Department of Management Science and Information Systems in the College of Business at Oklahoma State University in Stillwater.

**MANJUNATH KAMATH** (mkamath@okstate.edu) is a professor in the School of Industrial Engineering and Management and director of the Center for Computer Integrated Manufacturing at Oklahoma State University in Stillwater.

**WILLIAM J. KOLARIK** (kolarik@okstate.edu) is a professor and head of the School of Industrial Engineering and Management at Oklahoma State University in Stillwater.

**ESWAR SIVARAMAN** (esivaram@gnu.edu) is a visiting assistant professor in the Department of Systems Engineering and Operations Research at George Mason University in Fairfax, VA.

This research was supported by a grant from the National Science Foundation (Award No. 0075588) under the Scalable Enterprise Systems Initiative; the research team included M. Kamath, N. Dalal, W. Kolarik, and A. Lau.



# HOW TO MANAGE YOUR SOFTWARE PRODUCT LIFE CYCLE WITH

# MAUI

BY LUIGI SUARDI

*MAUI monitors multiple software engineering metrics and milestones to analyze a software product's development life cycle in real time.*

Software vendors depend on writing, maintaining, and selling quality software products and solutions. But software product conception, planning, development, and deployment are complex, time-consuming, and costly activities. While the market for commercial software demands ever higher quality and ever shorter product development cycles, overall technological complexity and diversification keep increasing. One result is that the rate of project failure is increasing, too. Software products consist of bytes of data, functions, files, images; companies' exclusive resources are human beings. Development organizations thus require ways to measure human code-writing productivity and quality-assurance processes to guarantee the continuous improvement of each new product release.

The following steps outline a hypothetical software product life cycle:

*Customer data.* The sales force collects and enters customer data into a Siebel Systems' customer relationship management system.

*Product requirements.* The customer data is converted by product architects into product

requirements and entered into a Borland CaliberRM collaborative requirements management system for high-level product definition.

*Development.* Project management tools (such as MS Project, ER/Studio, and Artemis) are used by product development managers during product design and engineering. At the



same time, source code development is supported by the Concurrent Versioning System [2], allowing programmers to track code changes and support parallel development.

*Testing.* Concluding the coding phase, the quality-assurance team uses various testing tools (such as Purify, PureCoverage, and TestDirector) to isolate defects and perform integration, scalability, and stress tests, while the build-and-packaging team uses other tools to generate installable CD images and burn CDs. In this phase of product development the Vantive system tracks product issues and defects. Testers open Vantive tickets, or descriptions of problems, that are then examined and eventually resolved by product developers. Robohelp and Documentum support their documentation efforts.

*Release and maintain.* The software product itself is finally released to the market, where its maintenance process begins a more simplified customer-support life cycle with the help of such tools as MS Project, Concurrent Versions System (CVS), and Vantive.

What happens when something goes wrong, as it inevitably does, milestones slip, or productivity, quality, or customer satisfaction falls off? How does the development company address and solve these problems? Critical questions the product developer should be able to answer include:

- How much does product development, maintenance, and support cost the company? How does quality relate to cost and stability to milestones?;
- Why does a particular product's schedule keep slipping while other product schedules stay on track? Is resource allocation adequate and management stable?;
- How many worker-hours have gone into a particular product? How are they apportioned among requirements management, planning, design, implementation, testing, and support? How long does it take to resolve customer complaints?;
- How stable is each software artifact? What is the appropriate level of quality? What is the extent of the quality assurance team's test coverage?; and
- What can be done to minimize the rate of failure and the cost of fixing defects, improving milestone accuracy and quality and optimizing resource allocation? (see Figure 1).

Those struggling to find answers include product-line and quality-assurance managers, process analysts, directors of research, and chief technology officers.

One approach they might take is to change the product development process by adopting a more

formal product life cycle [4], possibly introducing an integrated product development suite (such as those from Rational Software, Starbase, and Telelogic), leveraging embedded guidelines, integration, and collaboration functions. It may be their only option when the degree of software product failure is so great that the software product development life cycle must be completely reengineered. However, this approach is too often subjective and politically driven, producing culture shock yet still not solving problems in such critical areas as project management and customer support.

Another approach is to maintain the current process, acquire a much better understanding of life cycle performance, and take more specific actions targeting one or more phases of the life cycle. The resulting data is used to plan and guide future changes in a particular product's life cycle. It is highly objective, involving fewer opportunities for subjective decision making while minimizing change and culture shock and promising extension to other phases of the product life cycle.

### Measuring Critical Features

Software product life cycle management starts by measuring the critical features of the product and the activities performed in each phase of its life cycle. Useful life cycle metrics [1, 3] include:

- Changes of scope;
- Milestone delays;
- Rate of functional specification change;
- Percent of test coverage;
- Defect resolution time;
- Product team meeting time;
- Customer problem response time; and
- Critical systems availability.

Ideally, the data is aggregated and summarized in a product life cycle data warehouse where it is then available for generating views, automated alerts, and correlation rules from day-to-day operations, product status summaries, and top-level aggregated and executive data. These views, alerts, and rules support a variety of users, from the product line manager to the process analyst, and from the software test engineer to the chief technology officer. Figure 2 outlines an agent-based data acquisition, alert, and correlation architecture for the hypothetical life cycle described earlier.

Much of this work is performed by software agents autonomously mining the data and providing automated alerts based on thresholds and correlation rules. Phase-specific alerts are generated in, for exam-

ple, engineering, when fixing defects would take too long or require too many new lines of code. Global alerts are generated when, for example, the research and development expenses are not proportional to the sales levels for a specific product or when new

stones are likely to slip. Further analysis shows the entire development staff is busy addressing current release problems. An analyst prepares a detailed report to alert company executives, who might then decide to: assign an expert product architect to assess the situation and propose a recovery plan; notify customers the next release is delayed (quantified based on the assessment); and review the product team's technical and management skills to determine whether and which actions (such as training and adjusting responsibilities) are needed to increase product quality and customer satisfaction.

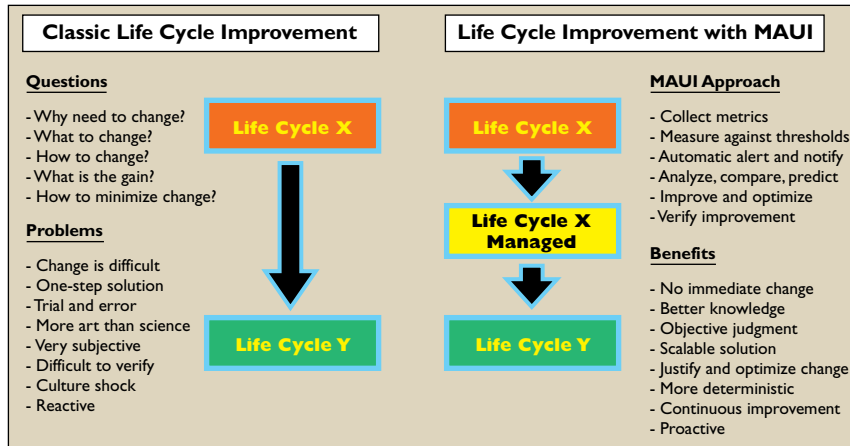


Figure 1. Software product life cycle improvement scenarios.

a system might alert development managers to the following situations:

*Too much time to resolve product defects.* The managers drill into details provided by the system and notice that some components keep changing, prompting them to organize a code review of the components and identify and order improvements to their design and modularity. As a result, the product becomes more stable, and the time to resolve defects decreases.

*Too many defects.* The system reports that many more defects are generated for product X than for the other, say, eight products for which the quality assurance managers are responsible. After analyzing the current resource allocation with their direct reports they move resources from the most stable products to product X and notify the development organization of the situation. Focusing on the right products and quickly reacting to alerts increases overall product quality.

*Missed milestones.* The system reports the correlation of metrics (such as rate of defect generation, time needed to resolve a support issue, and overall stability and quality) indicates a product's next release mile-

requirements crop up toward the end of the development cycle. Such

management approach called Measure, Alert, Understand, Improve, or MAUI, to manage several problematic software projects. Focusing on the

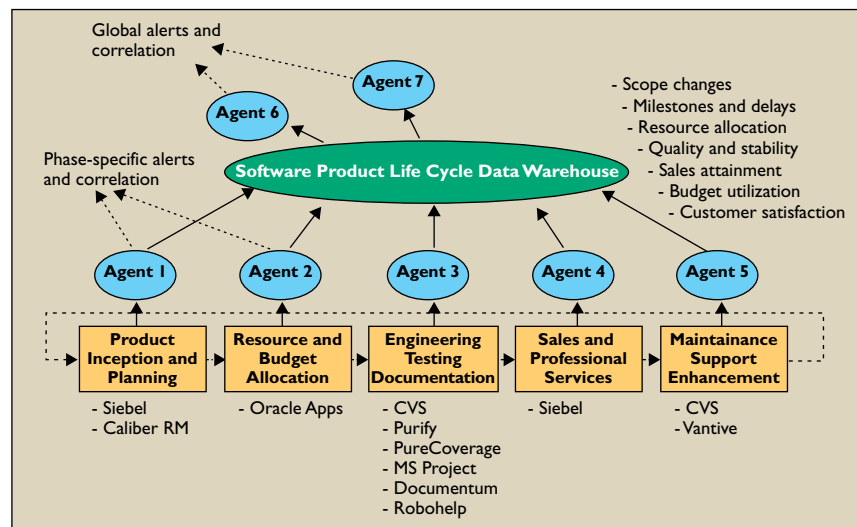


Figure 2. Managed software product life cycle.

engineering phase of the software product life cycle, it was designed to monitor development activities carried out through CVS and Telelogic's Synergy Configuration Management (SCM) system. Several months of daily monitoring revealed trends and patterns in the metrics and parameters of these projects' life cycles. The tables here summarize this data, along with the advice we generated for a number of critical BMC projects and teams.

The metrics in Table 1 help the development team analyze project activities. The software engineering monitor (SEM) agent collects them daily at the file level, aggregates them into directories and projects

Metric	Definition	
<b>Stability index</b>	Last week percent change – 70%	percent of the code that has changed in a week
	Last week active users – 10	number of developers that made changes in a week
	Avg. LOC per active user – 10	average number of LOC per developer
	Avg. active user life span – 10	average time spent by a developer on the project
<b>Quality index</b>	Avg. defect line changes – 40	average number of LOC needed to fix a defect
	Documentation level – 30	number of LOC per every line of comment
	Active defects – 20	number of defects
	Avg. routine length – 10	average length of a routine
<b>Documentation level</b>	Number of LOC per every line of comment	
<b>Weekly turmoil</b>	Percent of the source code that changes in a week	
<b>Defect complexity</b>	Average LOC needed to fix a defect	
<b>Average time spent by developers on project</b>	Average length of time a developer has been working on the project or component.	
<b>Average lines of code per developer</b>	How many LOC each developer on the team is responsible for (total LOC/team size)	
<b>Team size</b>	Number of developers active in the last six months	

**Table 1. Metrics definitions; LOC = lines of code.**

according to the hierarchies defined in the underlying SCM tools, and stores them in a metric history database for later use [5]. The SEM agent generates alerts and events and notifies development team members

Overall status	Alarm	
<b>Stability</b>	6.8	average + no trend
<b>Quality</b>	6	average + trend to lower quality
<b>Documentation level</b>	2	good (two LOC for every line of comment)
<b>Weekly turmoil</b>	0.55%	very low but higher on average
<b>Total lines of code</b>	25,098 LOC	small project size
<b>Defect complexity</b>	170 LOC	high + trend to higher (avg. 170 LOC per fix)
<b>Last month line changes</b>	2,083 LOC	trend to more changes
<b>Average lines of code per developer</b>	2,091 LOC	very low
<b>Average time spent by developers on project</b>	92 days	low (about three months)
<b>High complexity fixes</b>	278,618, 283,336,	1,183 LOC 12/20/01
	294,694	2,135 LOC 08/15/01
	246,419	

Data acquired with SEM. Observation window: 200 days; monitoring period: 90 days.

**Table 2. Project Jupiter engineering activities monthly analysis, Jan. 2002.**

automatically when metric thresholds are crossed. For any given metric collection cycle the observation window is 200 days, so all activities older than 200 days from collection time are ignored by the SEM agent. The indexes are special metrics defined by the development team with BMC's project managers in light of their own criteria for stability and quality. The indexes are defined as weighted sums of basic

Overall status	OK	
<b>Stability</b>	8	good + trend to higher stability
<b>Quality</b>	6	average + trend to higher quality
<b>Documentation level</b>	4	good (four LOC for every line of comment)
<b>Weekly turmoil</b>	0.29%	low and stable
<b>Total lines of code</b>	94,585 LOC	small project size
<b>Defect complexity</b>	57 LOC	average + trend to lower (avg. 57 LOC per fix)
<b>Last month line changes</b>	694 LOC	trend to fewer changes
<b>Average lines of code per developer</b>	31,528 LOC	very good
<b>Average time spent by developers on project</b>	167 days	good (more than five months)
<b>High complexity fixes</b>	none	

Data acquired with SEM. Observation window: 200 days; monitoring period: 200+ days.

**Table 3. Project Saturn engineering activities monthly analysis, June 2002.**

metrics and calculated using a formula normalizing their value from 0 to 10.

Table 2 indicates the SEM agent has reported an overall alarm status for a project called Jupiter due to the low number of lines of code (LOC) per developer, thus supporting the following analysis:

- The significant level of code turmoil suggests the product is unstable and therefore not yet ready for testing;
- The defect-complexity trend suggests immaturity and a related lack of well-defined components;
- The low number of LOC per developer suggests too many developers may still be changing the code; and
- Low average time spent by developers on the project could indicate a lack of ownership within the development team or the reshuffling of resources.

This analysis allows project managers to proactively review resource allocation and task assignments and perform targeted code reviews of the aspects of the product that change most often and

that involve an unacceptably high number of defects. Long-term savings of time and money in customer support and maintenance are potentially significant.

Table 3 indicates that the SEM agent has reported an overall OK status for another project, this one called Saturn, supporting the following analysis:

- The product is fairly stable and evolving toward even greater stability;
- Quality is improving, and the documentation level is good;
- Decreased defect complexity and lack of complexity fixes suggest the product is becoming mature and will soon be ready for release to customers; and
- High numbers of LOC per developer and time spent by developers on the project indicate project ownership and resource allocation are well defined.

This analysis shows that project managers are doing a good job. Further analysis might also suggest these managers would probably be comfortable releasing the product earlier than predicted, even beating the schedule. Benefits from reinvesting immediate revenue into product improvements are potentially significant.

These early experiments in MAUI real-time development monitoring demonstrate the value of continuously measuring software engineering metrics (see Figure 3). The MAUI prototype provides a real-time feedback loop that helps teams and managers quickly identify problem areas and steer software development projects in the right direction.

utilities, and other components would be adjusted to generate more meaningful alerts and provide tighter control of critical components; and *Determine how the MAUI approach might be used with standard framework metrics.* Monitoring metrics from such frameworks as the Capability Maturity Model developed by the Carnegie Mellon Software Engineering Institute and the ISO 9000 software quality standard developed by the International Organization for Standardization would help software development teams make better decisions about the critical steps needed to ensure delivery of software products to market.

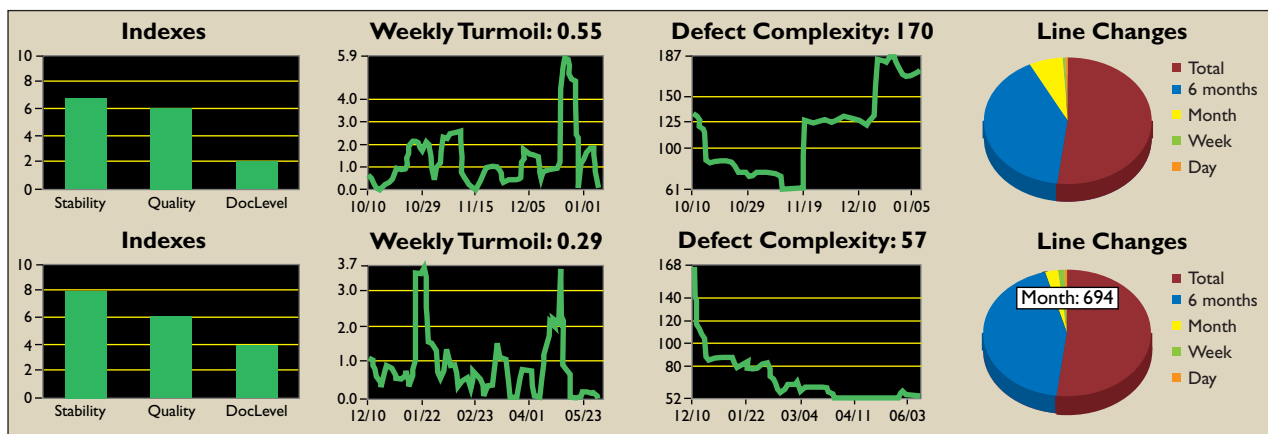


Figure 3. Project Jupiter and project Saturn trends viewed through the SEM Web interface.

BMC's adoption of MAUI has been limited by three main concerns:

- No instant results.* Monitoring must be done for at least two to three months before meaningful patterns emerge;
- Novelty of the approach.* Most people still feel other more important things must be done first; and
- Misuse of data.* Some people see the risk of a Big Brother syndrome, where both managers and programmers fear their work is constantly being monitored, evaluated, and criticized.

Future MAUI improvements include:

- Pattern identification.* New project behavior could be compared to previous project behavior, helping pinpoint required changes in direction;
- Adjustable metric thresholds depending on current project phase.* Integration of components, testing, and maintenance would make it possible for MAUI to generate better alerts;
- Adjustable metric thresholds depending on type of product or component.* Core APIs, user interfaces,

## Conclusion

Most of the value of life cycle management follows from automating data acquisition, providing alerts and correlation rules, identifying bottlenecks, increasing quality, optimizing critical processes, saving time, money, and resources, and reducing risk and failure rate. Key benefits include:

- Improved product-development processes.* The knowledge, repeatability, and performance of the product-development process already being used can be improved without dramatically changing the process itself;
- Automated alerts.* Data is acquired from various phases of the product life cycle, monitored against predefined thresholds and correlation rules, and used to automatically alert users to problems;
- Proportional scaling.* The complexity and cost of the management infrastructure needed to solve problems is proportional to the difficulty of solving the problems;
- Improved evaluation.* Having numeric and quantitative data is a good starting point for producing a more objective evaluation of the software being developed while helping justify and optimize

changes to the process;

*Improved predictability of results.* Analyzing historical data helps reveal patterns and predict future results.

*Happier people.* Because life cycle management makes it possible to accomplish tasks through best-of-breed solutions and tools, the people responsible for the tasks are happier;

*Critical tool availability.* Uptime of critical tools is monitored to ensure tool availability; and

*Continuous monitoring.* The product development process is continuously monitored, measured, and improved.

A software product life cycle that is stable, predictable, and repeatable ensures timely delivery of software applications within budget. A predictable life cycle is achievable only through continuous improvement and refinement of the processes and their tools. The real-time MAUI approach to product life cycle monitoring and analysis promises the continuous improvement and refinement of the soft-

ware product life cycle from initial product concept to customer delivery and beyond. ■

## REFERENCES

1. Florac, W., Park R., and Carleton A. *Practical Software Measurement: Measuring for Process Management and Improvement*. Software Engineering Institute, Carnegie Mellon University, Pittsburgh, 1997.
2. Fogel, K. *Open Source Development with CVS*. Coriolis Press, Scottsdale, AZ, 1999.
3. Grady, R. *Practical Software Metrics for Project Management and Process Improvement*. Prentice Hall, Inc., Upper Saddle River, NJ, 1992.
4. Jacobson, I., Booch, G., and Rumbaugh, J. *The Unified Software Development Process*. Addison-Wesley Publishing Co., Reading, MA, 1999.
5. Spuler, D. *Enterprise Application Management With PATROL*. Prentice Hall, Inc., Upper Saddle River, NJ, 1999.

---

**LUIGI SUARDI** (luigi\_suardi@bmc.com) is manager of the Integration Blueprint and OpenPortal teams at BMC Software, Inc., in Houston, TX.

---

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

---

© 2004 ACM 0002-0782/04/0300 \$5.00



# THE ROLE OF PERSONALITY in Web-based Distance Education Courses

BY EYONG B. KIM AND MARC J. SCHNIEDERJANS

*Before investing in a Web-based education program, managers should consider how effective it will be when paired with employee personality traits.*

**THE MIS FIELD** requires its participants to engage in continuous education to remain up to date. One approach to meeting this demand is through the use of a Web-based virtual learning environment (VLE). One type of VLE is totally Web-based education (TWE), which requires no class meetings and little instructor contact. When we consider the conveniences of TWE courses to today's globally expanding firms, the ever-increasing bandwidth of the Internet leading to improvements in TWE course interactivity and quality, and the pennies-to-dollars cost of operating TWE course offerings, it is obvious that TWE courses are necessary, and will be more pervasive with each passing year [10].

The recent evolution in the delivery of education has resulted in much debate on the effectiveness of distance education via TWE courses [1, 3, 9]. While both pro and con positions of this debate raise important issues, in our view, both sides of this debate have overlooked an important factor, one that involves the personality characteristics of prospective students. Research dating back to the 1960s demonstrates that an individual's personality characteristics are good predictors of future training and learning performance [11]. Inspired by this research, we believe that the personality characteristics

that contribute to student learning should be assessed in order to determine who is most certain to benefit from TWE courses. Just as students must qualify for college, so too should prospective students be qualified to take TWE courses by their personality characteristics.

As a study by Mount and Barrick [8] illustrates, much research has been directed at establishing relationships between individual personality characteristics and learning performance. These researchers found that characteristics such as "stability," "openness," "conscientiousness," "agreeableness," and "extraversion" (referred to as the "big five" personality characteristics) are all related to some type of learning performance.

Most of the prior research on VLE learning is focused on issues such as learner satisfaction, interface design, or attitude of learners [2]. However, some researchers suggest that effective learning in a VLE, compared to traditional classrooms, has been observed for mature and motivated learners while less motivated learners tend to suffer [5]. A VLE, especially Web-based courses, has inherent problems in settings where users lack the abilities required to take full advantage of the medium [4]. But no research to date has directly examined whether students' personality characteristics caused them to have

Wonderlic Personality Characteristics Inventory (PCI) Scales	Definitions	Relationship with Grade Performance (Statistical significance)	Corresponding Subscales	Subscale Relationship with Grade Performance (Statistical significance)
Agreeableness	A tendency to be courteous, helpful, trusting, good-natured, cooperative, tolerant, and forgiving.	High ( $p<0.001$ )	Cooperation Consideration	Moderate ( $p=0.039$ ) High ( $p<0.002$ )
Extraversion	A tendency to be sociable, gregarious, talkative, assertive, adventurous, active, energetic, and ambitious.	Moderate ( $p=0.044$ )	Sociability Need for recognition Leadership orientation	Low ( $p=0.332$ ) Low ( $p=0.441$ ) High ( $p<0.001$ )
Conscientiousness	A tendency to be hardworking, dependable, efficient, and achievement striving.	Moderate ( $p=0.014$ )	Dependability Achievement striving Efficiency	Low ( $p=0.586$ ) High ( $p<0.003$ ) Low ( $p=0.456$ )
Stability	A tendency to handle stress, to maintain an even temperament, and to have a high degree of composure and self-confidence across most situations.	High ( $p<0.001$ )	Even-temperament Self-confidence	Moderate ( $p=0.029$ ) Moderate ( $p=0.040$ )
Openness	A tendency to be imaginative, cultured, curious, polished, original, broadminded, intelligent, and artistically sensitive.	High ( $p<0.001$ )	Abstract thinking Creative thinking	Low ( $p=0.616$ ) High ( $p<0.001$ )

**Table 1. Personality characteristics and their relationships with grade performance.**

such as teamwork, integrity, commitment to work, and learning orientation (see Table 2). All of the scales and subscales are based on an index score ranging from 0 to 100%, where 100% represents a high degree of the particular personality characteristic. The PCI instrument's scales are designed to be used individually or together in models for predicting an individual's potential for handling particular jobs or training. Comparisons with the PCI instrument scales and subscales are the basic measures of personality characteristics used in this study.

To obtain measures of learning performance, we selected junior-

trouble with a TWE course. If there is a significant performance difference among TWE course students depending on personality, it may be desirable to evaluate personality characteristics in advance to determine if a TWE program is the most appropriate educational format for them.

## Testing Personality

Is there a relationship between personality characteristics and grade performance in TWE courses? If so, can the relationship be used to discriminate between high-performing and low-performing students for the purpose of selecting candidates for TWE courses?

To answer these questions it was necessary to have measures for comparison on both subject personality characteristics and on learning performance. For this job we selected the Wonderlic Personality Characteristics Inventory (PCI) survey assessment instrument [12], a measuring instrument whose 50-year history of applications would be unquestionably reliable and valid to academic researchers. This instrument was also easily attainable (and affordable) for commercial use by practitioners. The PCI has been widely used among psychology researchers to provide measurement scales of the "big five" personality characteristics dimensions shown in Table 1. The PCI also provides a further breakdown of the characteristics with a series of 12 subscales measures (see Table 1), each providing a unique glimpse of components of an individual's personality. The survey instrument consists of 150 questions, covering the "big five" scale, the 12 subscales, and an additional measurement scale for "success" characteristics

level college students taking their first introductory MIS course as the sample for this research. We felt these students represented a less biased sample since they had no previous MIS education or experience to cause learning dissonance. The course was a TWE self-paced course and was delivered to students via the Web using the Blackboard Educational Software System. Students were required to use a textbook [6], its supplementary materials (which included a CD-based workbook that provided sample test questions, interactive learning simulators, and additional problem solutions), and an instructor-prepared Web-based MS PowerPoint slide presentation on each chapter as preparation for the exams. The slide presentation provided extensive textbook content with animated diagrams structured in a programmed-learning format to make it easy for students to learn on their own. The exams were available to be taken at the convenience of the student online through the Blackboard Web delivery system during the semester. The exam scores were combined into a percentage as a learning achievement measure used to gauge their learning performance. Achievement reflected in grades is important because the Web-course effectiveness of learning is measured in terms of students' achievement and satisfaction [7]. These learning achievement scores are hereafter referred to as "grade performance" measures, and are used in statistical comparisons with the personality characteristic measures from each student.

A total of 140 students were randomly selected as the subjects from a single class of over 200 who participated in the survey. These students ranged in age from

20 to 30 years, with a mean of 22.5 years. Participation and use of the PCI instrument by the students met all the requirements suggested by the Wonderlic Organization for a valid and reliable survey experience. Also, Wonderlic consistency statistics built into the survey instrument confirmed the validity of the student's answers.

## Survey Results

Is there a relationship between personality characteristics and grade performance in TWE courses? To answer this question correlations coefficients were computed between the personality characteristic measures and the grade performance measures for all 140 students. Having met all the necessary and sufficient conditions for an accurate statistical application, the resulting significant relationships (a value of  $p < 0.05$ ) between the five personality characteristics in Table 1 are listed as either "high" or "moderate." All "low" relationships represent statistically insignificant or non-relationships (a value of  $p > 0.05$ ). Since all five of the characteristic measures turned out to be significantly related to grade performance but had differing degrees of significance (high to moderate), we computed the correlations with each of the 12 subscales to see if any subscale personality characteristics might differentiate grade performance.

Interestingly, but not surprisingly, the subscales measuring personality characteristics that tend to support the learning climate in a traditional classroom setting, such as "sociability" and the "need for recognition" appear not to be related to grade performance. Also, the "abstract thinking" subscale is logically not required in TWE courses, since the content was presented in a more "programmed learning" format. The subscale personality characteristics of "dependability" and "efficiency" also dropped out. Since the TWE in this study was self-paced with only one deadline for all exam grades, the usual need to be efficient and dependable in meeting multiple semester deadlines on exams and homework typical of the traditional classroom appears not to be a personality characteristic requirement in a successful prospective TWE student in our study.

We further examined the use of the PCI "success" scales to determine if these personality characteristics show any relationship with grade performance. As shown in Table 2, we found that while, logically, "commitment to work" and "learning orientation" were

highly related to grade performance, the personality characteristics of "teamwork" and "integrity" were not. The TWE course in this study was performed by individual effort, not requiring team effort, and it is virtually impossible to show altruistic behavior in the TWE course environment of this study. Those students

Wonderlic PCI Success Scales	Definitions	Relationship with Grade Performance
Teamwork	The tendency of how well an individual might work with others and cooperate in groups.	Low ( $p=0.456$ )
Integrity	The tendency of altruistic behavior.	Low ( $p=0.881$ )
Commitment to work	The tendency to remain on a job for a long time, and not be undependable, irresponsible, impulsive, disorganized, or lack persistence.	High ( $p<0.001$ )
Learning orientation	The tendency of an individual to be willing to engage in activities to acquire knowledge, skills, and behaviors and to learn new methods and procedures to improve job effectiveness, how interested they are in developing themselves, seek opportunities to learn new and different ways of doing things, and enrolled in training programs that they are likely to be active and fully engaged participants.	High ( $p<0.001$ )

**Table 2. Success scales and their relationship with grade performance.**

whose high scores centered on the characteristics of "teamwork" and "integrity" might need these types of personality expression as outlets for their educational growth, but

they may not be ideal prospective TWE course students in our introductory MIS course.

In summary, we found a student's grade performance

	High Performer Identified	Low Performer Identified	Total Comparisons
Actual High Performer	125 (99.2%)	1 (00.8%)	126 (100.00%)
Actual Low Performer	2 (01.6%)	124 (98.4%)	126 (100.00%)

**Table 3. Accuracy of personality characteristics in discriminating between high and low performers.**

of this TWE course does show a significant relationship with a number of personality characteristics. The ideal prospective TWE student for the introductory MIS course in this study is compliantly cooperative, considerate, even-tempered, self-confident, a creative thinker, committed to work, shows leadership, needs to achieve, and has a positive learning orientation.

Our findings may be limited in terms of other types of distance education pedagogy. That is, TWE courses that use other types of educational methodologies, like interactive chat sessions, where a characteristic like "teamwork" may be a significant factor, might show differing statistical results. Since this study is the first of its kind dealing with TWE and personality characteristics, we sought only to establish the connection personality characteristics and student grade performance, rather than to define the relationship with all possible educational pedagogy. Indeed, due to the interactive effects of differing pedagogy on the learning experience of students and the differing types of educational content that can be taught in TWE, it may be impossible to

determine which pedagogy and personality characteristics are the best combination for improving grade performance.

Can the relationship between personality characteristics and grade performance be used to distinguish between high- and low-performing students for the purpose of selecting candidates for TWE courses? A simple comparative test is used to determine discrimination, or selection, accuracy. We took the 14 students whose grade performance was in the top 10% of the sample and designated them as "high performers" for this test and we took the 14 students whose grade performance was in the bottom 10% and designated them as "low performers." We then computed the mean subscale scores for all seven of the significant personality characteristics in Table 1 and the mean of the two success scale scores in Table 2, based on the student sample of 140. If the personality characteristics do in fact discriminate between the high- and low-performers, then each of the high-performing students should have all nine (7 + 2) scores fall above average, and the low performers should have their nine scores fall below average.

The results of the 126 comparisons (14 students times 9 scores each) for both the high and low performers (a grand total of 252 comparisons) are presented in Table 3. With little exception, the resulting scale scores distinguished exactly as they should—those students who can expect to have high grades in a TWE course from those who will have low grades. Clearly in this study, the personality characteristics of the students can easily discriminate for purposes of selection between the highest performing and lowest performing students in the TWE course.

While the age range of the undergraduate students in this study was limited to young adults, the potential market for TWE courses clearly includes older adults. In response to this limiting factor of our study, a small sampling of 23 students taking a TWE quantitative methods graduate MBA course was examined. These students ranged in age from 28 to 52 years, with a mean of 34.7 years. In this TWE MBA course, the education materials were slightly different, and included Web-based reading material (no textbook), with supporting technology like pop-up windows for tables and animated diagrams and figures. All other educational materials and environmental factors, including the means of testing this older group sample, were similar to the undergraduate study. Running the same set of statistics on this subsequent group resulted in the same five personality characteristics being significant, although not as statistically significant (at the moderate level of  $p < 0.040$ ) as the larger undergraduate result. Given the smaller sample size, the loss of statistical significance is quite understandable.

We feel this smaller subsequent sample of a graduate class helps to confirm the importance the five personality characteristics found in the undergraduate study, and supports the reliability and validity of our study's claims.

## Conclusion

While the single class sample of this study limits its generalizability, the obvious strengths of these analyses still demonstrate that a student's personality characteristics, as measured by the PCI survey instrument, can be a strong indicator of resulting grade achievement. As such, we feel that personality characteristics are worth measuring as qualifying criteria to take TWE courses.

Using the methodology presented here for other TWE courses requires a careful analysis of the learning skills desired, and their relationship with the scales and subscales of the PCI survey instrument. Our statistics presented here can be used as a beginning point for that analysis. Such an analysis might help prospective students get some idea of their future learning performance in a TWE course, and provide companies who fund the TWE courses some idea as to how well these courses might warrant their investment of time and money. ■

## REFERENCES

1. Bruckman, A. The future of e-learning communities. *Commun. ACM* 45, 4 (Apr. 2002), 60–63.
2. Cardler, J. Summary of current research and evaluation of findings on technology in education, Working Paper, Educational Support Systems, San Mateo, CA, (1997).
3. Decker, T., et al. Debating distance learning. *Commun. ACM* 43, 2 (Feb. 2000), 11–15.
4. Heller, R. The role of hypermedia in education: A look at research issues. *Journal of Research on Computing in Education* 22, 4 (1990), 431–441.
5. Hiltz, S. *The Virtual Classroom: Learning without Limits Via Computer Networks*. Albex Publishing, Norwood, NJ, 1993.
6. Laudon, K., and Laudon, J. *Management Information Systems*, 7th ed. Prentice-Hall, Upper Saddle River, NJ, 2002.
7. Maki, R., Maki, W., Patterson, M., and Whittmaker, P. Evaluation of a Web-based introductory psychology course: Learning and satisfaction in on-line versus lecture courses. *Behavior Research Methods, Instruments, and Computers* 32, 2 (2000), 230–239.
8. Mount, M., and Barrick, Murray R. Five reasons why the 'big five' article has been frequently cited. *Personnel Psychology* 51, 4 (Winter 1998), 849–857.
9. Phoha, V. Can a course be taught entirely via email? *Commun. ACM* 42, 9 (Sept. 1999), 29–30.
10. Tsichritzis, D. Reengineering the university. *Commun. ACM* 42, 6 (June 1999), 93–100.
11. Wiggins, N., Blackburn, M., and Hackman, J. The prediction of first-year graduate success in psychology. *Journal of Educational Research* 63, (1969).
12. Wonderlic, Inc. 2002 Resource Guide; [www.wonderlic.com](http://www.wonderlic.com).

**EYONG B. KIM** ([ekim@hartford.edu](mailto:ekim@hartford.edu)) is an assistant professor of MIS, Management Department, Barney School of Business, University of Hartford, West Hartford, CT.

**MARC J. SCHNIEDERJANS** ([Mschniederjans1@unl.edu](mailto:Mschniederjans1@unl.edu)) is the C. Wheaton Battey Distinguished Professor of Business, Department of Management, College of Business Administration, University of Nebraska-Lincoln, Lincoln, NE.



# WHAT CAUSES STRESS IN INFORMATION SYSTEM PROFESSIONALS?

*Job stress can lead to burnout and turnover, costing IT organizations countless dollars in replacement costs, and making methods for measuring and minimizing stress a business benefit.*

**S**tress among information system (IS) professionals is long recognized as a key factor affecting IS productivity and turnover and leading to substantial associated costs. It is estimated that, on average, IS employees work 50 hours per week; almost half work an average of six hours on Saturdays and Sundays; and about 70% have worked while sick [2]. It has also been recently proposed that high stress levels affect the productivity of IS employees. In a recent survey of 16,000 international technology professionals in 28 nations, the productivity of U.S. programmers was shown to be on average 7,700 lines of code, compared to 16,700 lines for non-U.S. programmers [1]. One reason cited for this difference is job stress from "putting in 70-hour work weeks to meet business pressures and deliver IT projects faster." Some employees are opting to switch careers as a result of job stress. It is estimated the average replacement cost for an IS employee runs between \$32,000 and \$34,000. Stress and the

turnover it can cause may thus be a costly problem for an organization.

It is becoming increasingly clear that steps must be taken to address the problem of high stress because of its effect on employee productivity and turnover. However, before such strategies can be articulated, it is essential to examine the key sources of stress for IS employees. This article reports the results of a series of studies we have conducted to understand primary stressors for IS employees. We conducted in-depth interviews and collected data using an open-ended questionnaire.

This resulted in a list of 33 stressors cited as most common. We categorized the stressors based on their association with one of seven factors, described as follows:

- *Training.* Two stressors were associated with this factor, involving the need for appropriate training and skills development to complete tasks.
- *Deadlines.* Five stressors were associated with the pressures of meeting time dead-

ILLUSTRATION BY JOHN UELAND



lines and the need to complete projects within schedule.

- *Coworkers.* Five stressors were associated with this factor, involving the pressures of working with coworkers and elements of power struggles and conflicts that may result from working with others.
- *Performance evaluations.* Three stressors were associated with performance evaluations.
- *Job security.* Six stressors were associated with this factor, which involves concerns about job loss due to downsizing, mergers, or other factors.
- *Career development.* Four stressors were associated with the needs of IS employees to keep up with developments in the field and pressures that result from continuing skill development.
- *User demands.* Eight stressors were associated with pressures put on IS staff by users, such as dealing with the IS user interface.

The averages of each factor (see the table on the next page) show their relative intensity on a 1–7 scale, with 1 implying a nonstressor and 7 associated with high stress. As the table illustrates, the pressure to meet specific deadlines was rated as the single most important factor associated with stress, closely followed by user demands. In addition, all 33 items were combined into a single scale, referred to as the Stress measurement and determination inventory (SMDI). Its average value for the current sample is 4.3.

## Stress Variations

In order to understand whether stress levels are equal across various organizational characteristics, we examined seven demographic variables:

- Number of employees in the IS group,
- Years worked in the IS field,
- Years employed at current job,
- Number of IS jobs held previously,
- Hierarchical level of the respondent,

- Age group of respondent, and
- Gender of respondent.

We split the sample in quartiles for three of the demographic variables: number of employees, years worked in IS, and age. For the remaining demographic variables, we split the sample into three equal groups. Next, we conducted an ANOVA followed by a Scheffe's test (for pair-wise comparisons) to examine how stress levels (as indicated by the SMDI) differed from the intensity of individual stressor components. The results are shown in Figure 1.

Each demographic variable axis in Figure 1 is divided into seven equal segments, representing the seven stressors described

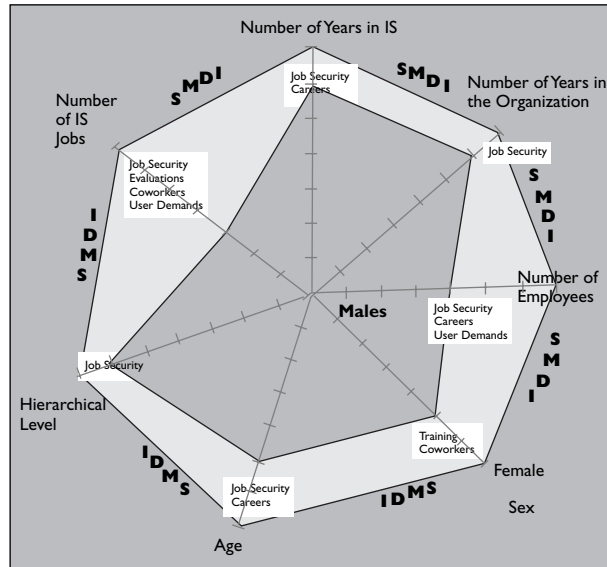


Figure 1. The organizational stress band.

previously. In general, we found that increases in the demographic variables resulted in higher SMDI scores. We also found SMDI scores to be higher in female employees. These higher scores appear to be mediated by stressors involving coworkers and adequacy of training. Female respondents who worked in organizations with more than 180 employees reported higher stress scores than those with fewer employees, for example. Respondents employed at their current jobs for more than 11 years had significantly higher stress scores in the area of job security than those employed for shorter periods. In general, as the number of years in their current employment and their number of

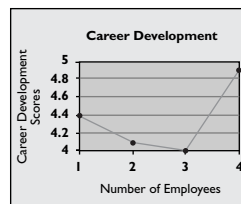


Figure 2. The organizational stress dip effect.

years in IS increased, respondents became increasingly concerned about job security. With an increasing number of IS jobs held, the stress drivers were job security, evaluations, coworkers, and user demands. Upper-level employees were more likely to express job security concerns than employees working at lower hierarchical levels. Increasing age was also associated with stress over job security, as well as career development stress.

Connecting the stress drivers for the demographic variable axes resulted in the shaded areas of Figure 1 that we call the organizational stress band, which illustrates the regions where stressors are reported to be high and significantly different within different

# CLEAR COMMUNICATIONS

## ABOUT PERFORMANCE AND REWARD EXPECTATIONS WERE NOTED BY MANY AS USEFUL IN REDUCING STRESS LEVELS.

segments of each demographic variable. It shows that higher values of each demographic variable are associated with higher SMDI scores, and specific components of the SMDI vary across different levels of the demographic variable. Finally, this analysis demonstrates that respondents with the following characteristics report higher stress levels: those working in organizations with a large number of employees (>180), those employed at the organization for more than 11 years, those who have held more than 15 jobs previously, those at upper levels of the organizational hierarchy, and those who are more than 40 years old and female. The stress band also shows the differentiating characteristics of stress components.

We also discovered an interesting trend in the reported career development stress scores within the demographic variable of number of employees. We found stress levels to be higher at the low and high levels of the variable than at the intermediate levels. This is also shown in Figure 2 and is called the stress dip. We found similar trend lines in other demographic variables: number of years with the organization, number of IS jobs held previously, number of years in IS, and age.

### Stress and Burnout

We investigated whether the stressors identified in this study were related to negative organizational consequences such as burnout, job satisfaction, and intention-to-quit. We measured burnout by using the emotional exhaustion subscale of the Maslach Burnout Inventory (MBI) [3]. We also collected data to measure job satisfaction and an individual's inten-

tion to quit his or her job.

The correlation coefficients for these variables are shown in the table here. We had expected a high negative correlation between the SMDI (and its components) and burnout, job satisfaction, and intention-to-quit. All of the coefficients were as expected with the exception of those between satisfaction and user demands and between deadlines and intention-to-quit.

### Preventive Strategies

We designed the studies reported here with two fundamental objectives:

to develop an instrument that can assist in an examination of stressors for IS employees; and through the development of this instrument, provide directions

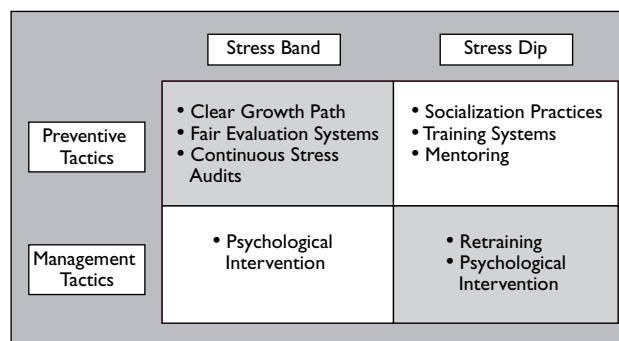


Figure 3. Stress intervention strategies.

Variable	Means	Correlation Coefficients			Percentile Scores for SMDI	SMDI Score
		Emotional Exhaustion	Satisfaction	Intention-to-quit	Percentile	
SMDI	4.3	0.37*	-0.22*	0.20*	10	73
Training	4.2	0.21*	-0.13*	0.19*	20	87
Deadlines	4.8	0.36*	-0.14*	0.07*	30	99
Coworkers	4.0	0.22*	-0.14*	0.13*	40	109
Work Evaluation	4.3	0.16*	-0.15*	0.13*	50	121
Job Security	2.8	0.14*	-0.23*	0.17*	60	129
Career Development	4.3	0.25*	-0.27*	0.17*	70	140
User Demands	4.6	0.32*	-0.05*	0.13*	80	151
		*Coefficients above 0.12 are significant at 0.05.			90	165

Means, correlation coefficients, and percentile scores for the stressors.

for appropriate intervention strategies [5].

In this regard, the cross-sectional data used in this study may be used to develop tentative standards. Percentile scores for the SMDI and its factors are shown in the table. These

# WHILE THE MEASUREMENT AND DETERMINATION OF STRESS FACTORS IS THE FIRST STEP, THE INCORPORATION OF RESULTS INTO ACTIVE INTERVENTION PLANS IS THE NECESSARY SECOND PHASE OF ANY STRESS MANAGEMENT STRATEGY.

statistics may be used in a more precise evaluation of stress measurement. The data reported in the table represents pooled results from all studies.

While the measurement and determination of stress factors is the first step, the incorporation of results into active intervention plans is the necessary second phase of any stress management strategy. Both the stress band and the stress dip can be integrated into an organizational intervention strategy that includes both prevention and management tactics (see Figure 3).

Recalling the message of the stress band—that higher levels of variables such as age, tenure, and size of the organization are associated with higher stress in areas such as employment security, performance evaluation, and career development—preventive strategies may include developing clear growth paths within the IT field, establishing open evaluation systems, and committing to continuous stress audits within the organization. Planning ahead for employee transfer in case of outsourcing or mergers may be useful in addressing the core stressor of job security. Helpful approaches may include sessions with employees to explain the possibility of organizational changes, avoiding surprise news, and the providing a clear schedule of forthcoming changes.

The stress dip indicates that stress levels are high when variables such as age, tenure, and size of the organization are at their lowest. For example, new employees generally exhibit higher stress levels. Preventive tactics may include the use of formal socialization programs to allow new employees to learn the “rules of the game.” The primary stressors for new employees involve concerns about performance and career development. Building a suitable socialization program where clearly defined opportunities for advancement are explained can be useful. Clear communications about performance and reward expectations were noted by many of our respondents as useful in reducing stress levels.

In addition, creating formal retention strategies built around training can address both the stress band and stress dip. Higher stress levels in older

employees are not a new phenomenon in the IT world. Scalet [6] reports that only about 6.8% of the IT workforce in the U.S. is over the age of 55, compared to 11.7% of the general work force, and associations such as the Information Technology Association of America (ITAA) have received numerous reports of age discrimination. One reason for this is that the search for newer technical skills makes hiring younger employees more attractive. However, a formal and intensive retraining program may be used to leverage the business experience of older employees and reduce their stress levels. Finally, psychological intervention and counseling can assist employees who have been diagnosed with high stress levels.

## Conclusion

There can be no doubt of the need to adequately measure, determine the causes of, and manage the stress syndrome. Falling somewhere between a measurement and a diagnostic instrument, the SMDI can prove useful in isolating some of the stressors that may negatively affect IS employees. ■

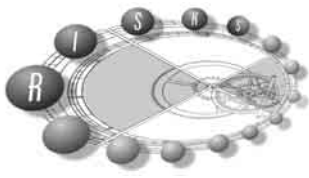
## REFERENCES

1. Hoffman, T. Are U.S. programmers slackers? *ComputerWorld* 33, 15 (Apr. 1999), 1–3.
2. King, J. Stress rattles ‘help!’ desks. *Computerworld* 29, 11 (Mar. 1995) 1, 16.
3. Maslach, C. *Burnout: The Cost of Caring*. Prentice Hall, Englewood Cliffs, NJ, 1982.
4. Ouellette, C. Living with pain. *ComputerWorld* 33, 16 (Apr. 1998), 9–10.
5. Quick, J.C., Quick, J.D., Nelson, D.L., and Hurrell, J.J. *Preventive Stress Management in Organizations*. American Psychological Associates, Washington, DC, 1997.
6. Scalet, S.D. The high price of age discrimination. *CIO* 23, 5 (May 2001), 136–140.

**VIKRAM SETHI** (vsethi@world.std.com) is chair and professor in the Department of Information Systems and Operations Management, Wright State University, Dayton, OH.

**RUTH C. KING** (ruthking@uiuc.edu) is an assistant professor of information systems in the College of Business, University of Illinois at Urbana-Champaign, IL.

**JAMES CAMPBELL QUICK** (jquick@uta.edu) is the director of the doctoral program in business administration at the University of Texas at Arlington.



## Risks of Monoculture

**T**he W32/Blaster worm burst onto the Internet scene in August of 2003. By exploiting a buffer overflow in Windows, the worm was able to infect more than 1.4 million systems worldwide in less than a month. More diversity in the OS market would have limited the number of susceptible systems, thereby reducing the level of infection. An analogy with biological systems is irresistible.

When a disease strikes a biological system, a significant percentage of the affected population will survive, largely due to its genetic diversity. This holds true even for previously unknown diseases. By analogy, diverse computing systems should weather cyber attacks better than systems that tend toward monoculture. But how valid is the analogy? It could be argued that the case for computing diversity is even stronger than the case for biological diversity. In biological systems, attackers find their targets at random, while in computing systems, monoculture creates more incentive for attack because the results will be all the more spectacular. On the other hand, it might be argued that cyber-monoculture has arisen via natural selection—providers with the best security products have survived to dominate the market. Given the dismal state of computer security today, this argument is not particularly persuasive.

Although cyber-diversity evidently provides security benefits, why do we live in an era of relative computing monoculture? The first-to-market advantage and the ready availability of support for popular products are examples of incentives that work against diversity. The net result is a “tragedy of the (security) commons” phenomenon—the security of the Internet as a whole could benefit from increased diversity, but individuals have incentives for monoculture.

It is unclear how proposals aimed at improving computing security might affect cyber-diversity. For example, increased liability for software providers is often suggested as a market-oriented approach to improved security. However, such an approach might favor those with the deepest pockets, leading to less diversity.

Although some cyber-diversity is good, is more diversity better? Virus writers in particular have used diversity to their advantage; polymorphic viruses are currently in vogue. Such viruses are generally encrypted with a weak cipher, using a new key each time the virus propagates, thus confounding

signature-based detection. However, because the decryption routine cannot be encrypted, detection is still possible. Virus writers are on the verge of unleashing so-called metamorphic viruses, where the body of the virus itself changes each time it propagates. This results in viruses that are functionally equivalent, with each instance of the virus containing distinct software. Detection of metamorphic viruses will be extremely challenging.

Is there defensive value in software diversity of the metamorphic type? Suppose we produce a piece of software that contains a common vulnerability, say, a buffer overflow. If we simply clone the software—as is standard practice—each copy will contain an identical vulnerability, and hence an identical attack will succeed against each clone. Instead, suppose we create metamorphic instances, where all instances are functionally equivalent, but each contains significantly different code. Even if each instance still contains the buffer overflow, an attacker will probably need to craft multiple attacks for multiple instances. The damage inflicted by any individual attack would thereby be reduced and the complexity of a large-scale attack would be correspondingly increased. Furthermore, a targeted attack on any one instance would be at least as difficult as in the cloning case.

Common protocols and standards are necessary in order for networked communication to succeed and, clearly, diversity cannot be applied to such areas of commonality. For example, diversity cannot help prevent a protocol-level attack such as TCP SYN flooding. But diversity can help mitigate implementation-level attacks, such as exploiting buffer overflows. As with many security-related issues, quantifying the potential benefits of diversity is challenging. In addition, metamorphic diversity raises significant questions regarding software development, performance, and maintenance. In spite of these limitations and concerns, there is considerable interest in cyber-diversity, both within the research community and in industry; for an example of the former, see [www.newswise.com/articles/view/502136/](http://www.newswise.com/articles/view/502136/) and for examples of the latter, see the Cloakware.com Web site or Microsoft's discussion of individualization in the Windows Media Rights Manager. ■

**MARK STAMP** ([stamp@cs.sjsu.edu](mailto:stamp@cs.sjsu.edu)), an assistant professor of computer science at San Jose State University, recently spent two years working on diverse software for MediaSnap, Inc.