

# COMMUNICATIONS

CACM.ACM.ORG

OF THE

# ACM

10/2018 VOL.61 NO.10

## Human-Level Intelligence or Animal-Like Abilities?



Computing within Limits

Transient Electronics Take Shape

Q&A with Dina Katabi

Formally Verified Software  
in the Real World

# Communications of the ACM China Region Special Section

A collection of articles spotlighting how computing is transforming the China region, and the leading-edge industry, academic, and government initiatives currently underway, is coming to CACM this Fall. This section includes articles on Big Trends and Hot Topics by leading practitioners and academics from the China region, including:

- Cloud Infrastructure for World's Largest Consumer Market
- How AI-powered FinTech is Improving People's Lives
- The Future of Artificial Intelligence in China
- Supercomputers as SuperData and SuperAI Machines
- Last-mile Delivery and Autonomous Vehicles
- Quantum Communication
- And much more!

This special section is the first in a series coming to ACM's flagship magazine; each will feature articles authored by the region's leading computing professionals in a particular geographic region, highlighting the most exciting computing advances and innovation.



Association for  
Computing Machinery



# WANT TO LEAD GROUNDBREAKING RESEARCH IN SINGAPORE?

The National Research Foundation Singapore (NRF) invites outstanding researchers in their early stage of research careers to apply for the NRF Fellowship.

We welcome researchers in the following disciplines of science and technology:

- Engineering
- Computer Science, including Infocomm Technology and Interactive & Digital Media
- Natural or Physical Sciences
- Life Sciences

Each Fellow is provided with a research grant of up to S\$3 million. The research grant can be used to cover personnel, equipment and consumables costs. Application is open from December 2018 to February 2019.

Visit [www.nrf.gov.sg/NRFfellowship](http://www.nrf.gov.sg/NRFfellowship) for more details.



## Departments

- 5 **Cerf's Up**  
**The Internet in the 21<sup>st</sup> Century**  
*By Vinton G. Cerf*
- 
- 6 **Letters to the Editor**  
**Hennessy and Patterson on the Roots of RISC**
- 
- 8 **BLOG@CACM**  
**Can We Use AI for Global Good?**  
Amir Banifatemi observes how the AI for Good Summit “allowed us to start a dialogue, find a common frame of reference, and decide how our steps would be smart and structured.”
- 
- 31 **Calendar**
- 
- 114 **Careers**

## Last Byte

- 120 **Q&A**  
**Reaping the Benefits of a Diverse Background**  
Earlier this year, ACM named Dina Katabi of the Massachusetts Institute of Technology's Computer Science and Artificial Intelligence Laboratory recipient of the 2017 ACM Prize in Computing for her creative contributions to wireless systems.  
*By Leah Hoffmann*

## News



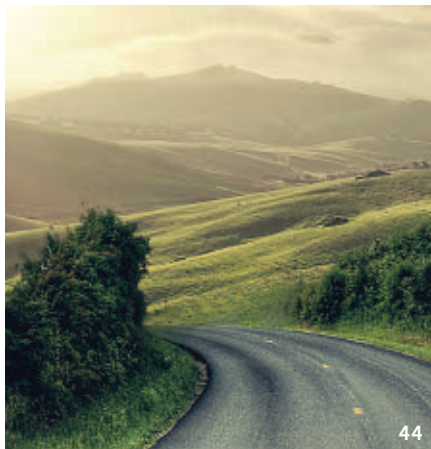
- 11 **Floating Voxels Provide New Hope for 3D Displays**  
In search of holograms that can be viewed from any angle.  
*By Chris Edwards*
- 
- 14 **Transient Electronics Take Shape**  
Advances in materials science and chemistry are leading to self-destructing circuits and transient electronics, which could impact many fields.  
*By Samuel Greengard*
- 
- 17 **The Dangers of Automating Social Programs**  
Is it possible to keep bias out of a social program driven by one or more algorithms?  
*By Esther Shein*

## Viewpoints

- 20 **Technology Strategy and Management**  
**The Business of Quantum Computing**  
Considering the similarities of quantum computing development to the early years of conventional computing.  
*By Michael A. Cusumano*
- 
- 23 **Privacy and Security**  
**A Pedagogic Cybersecurity Framework**  
A proposal for teaching the organizational, legal, and international aspects of cybersecurity.  
*By Peter Swire*
- 
- 27 **Kode Vicious**  
**The Obscene Coupling Known as Spaghetti Code**  
Teach your junior programmers how to read code.  
*By George V. Neville-Neil*
- 
- 29 **Viewpoint**  
**Building the Universal Archive of Source Code**  
A global collaborative project for the benefit of all.  
*By Jean-François Abramatic, Roberto Di Cosmo, and Stefano Zacchiroli*
- 
- Watch the authors discuss their work in this exclusive *Communications* video. <https://cacm.acm.org/videos/building-the-universal-archive-of-source-code>
- 
- 32 **Viewpoint**  
**Are CS Conferences (Too) Closed Communities?**  
Assessing whether newcomers have a more difficult time achieving paper acceptance at established conferences.  
*By Jordi Cabot, Javier Luis Cánovas Izquierdo, and Valerio Cosentino*



Practice



44

- 36 **The Mythos of Model Interpretability**  
In machine learning, the concept of interpretability is both important and slippery.  
*By Zachary C. Lipton*
- 
- 44 **The Secret Formula for Choosing the Right Next Role**  
The best careers are not defined by titles or résumé bullet points.  
*By Kate Matsudaira*

- 47 **Mind Your State for Your State of Mind**  
The interactions between storage and applications can be complex and subtle.  
*By Pat Helland*

**Q** Articles' development led by **acmqueue**  
queue.acm.org



**About the Cover:**  
This month's cover was inspired by a Judea Pearl quote that contends the vision system of an eagle outperforms anything created in the lab, yet the eagle cannot build a telescope or microscope. Adnan Darwiche uses this quote as a jumping-off point to argue what AI is and is not doing today (p. 56). Cover illustration by Hugh Syme.

Contributed Articles



56

- 56 **Human-Level Intelligence or Animal-Like Abilities?**  
What just happened in artificial intelligence and how it is being misunderstood.  
*By Adnan Darwiche*



Watch the author discuss his work in the exclusive *Communications* video.  
<https://cacm.acm.org/videos/human-level-intelligence-or-animal-like-abilities>

- 68 **Formally Verified Software in the Real World**  
Verified software secures the Unmanned Little Bird autonomous helicopter against mid-flight cyber attacks.  
*By Gerwin Klein, June Andronick, Matthew Fernandez, Ihor Kuz, Toby Murray, and Gernot Heiser*

- 78 **The Productivity Paradox in Health Information Technology**  
New York State healthcare providers increased their use of the technology but delivered only mixed results for their patients.  
*By Quang "Neo" Bui, Sean Hansen, Manlu Liu, and Qiang (John) Tu*

Review Articles



86

- 86 **Computing within Limits**  
The future of computing research relies on addressing an array of limitations on a planetary scale.  
*By Bonnie Nardi, Bill Tomlinson, Donald J. Patterson, Jay Chen, Daniel Pargman, Barath Raghavan, and Birgit Penzenstadler*

Research Highlights

- 95 **Technical Perspective**  
**A Control Theorist's View on Reactive Control for Autonomous Drones**  
*By John Baillieul*
- 
- 96 **Fundamental Concepts of Reactive Control for Autonomous Drones**  
*By Luca Mottola and Kamin Whitehouse*
- 
- 105 **Technical Perspective**  
**The Future of MPI**  
*By Marc Snir*

- 106 **Enabling Highly Scalable Remote Memory Access Programming with MPI-3 One Sided**  
*By Robert Gerstenberger, Maciej Besta, and Torsten Hoefler*



*ACM, the world's largest educational and scientific computing society, delivers resources that advance computing as a science and profession. ACM provides the computing field's premier Digital Library and serves its members and the computing profession with leading-edge publications, conferences, and career resources.*

**Executive Director and CEO**  
Vicki L. Hanson  
**Deputy Executive Director and COO**  
Patricia Ryan  
**Director, Office of Information Systems**  
Wayne Graves  
**Director, Office of Financial Services**  
Darren Ramdin  
**Director, Office of SIG Services**  
Donna Cappo  
**Director, Office of Publications**  
Scott E. Delman

#### ACM COUNCIL President

Cherri M. Pancake  
**Vice-President**  
Elizabeth Churchill  
**Secretary/Treasurer**  
Yannis Ioannidis  
**Past President**  
Alexander L. Wolf

#### Chair, SGB Board

Jeff Jortner  
**Co-Chairs, Publications Board**  
Jack Davidson and Joseph Konstan  
**Members-at-Large**

Gabriele Anderst-Kotis; Susan Dumais; Renée McCauley; Claudia Bauzer Medeiros; Elizabeth D. Mynatt; Pamela Samuelson; Theo Schlossnagle; Eugene H. Spafford  
**SGB Council Representatives**  
Sarita Adve; Jeanna Neefe Matthews

#### BOARD CHAIRS

**Education Board**  
Mehran Sahami and Jane Chu Prey  
**Practitioners Board**  
Terry Coatta and Stephen Ibaraki

#### REGIONAL COUNCIL CHAIRS

**ACM Europe Council**  
Chris Hankin  
**ACM India Council**  
Abhiram Ranade  
**ACM China Council**  
Wenguang Chen

#### PUBLICATIONS BOARD

**Co-Chairs**  
Jack Davidson; Joseph Konstan  
**Board Members**  
Phoebe Ayers; Edward A. Fox; Chris Hankin; Xiang-Yang Li; Sue Moon; Michael L. Nelson; Sharon Oviatt; Eugene H. Spafford; Stephen N. Spencer; Divesh Srivastava; Robert Walker; Julie R. Williamson

#### ACM U.S. Public Policy Office

Adam Eisgrau,  
Director of Global Policy and Public Affairs  
1701 Pennsylvania Ave NW, Suite 300,  
Washington, DC 20006 USA  
T (202) 659-9711; F (202) 667-1066

**Computer Science Teachers Association**  
Jake Baskin  
Executive Director

# COMMUNICATIONS OF THE ACM

Trusted insights for computing's leading professionals.

*Communications of the ACM* is the leading monthly print and online magazine for the computing and information technology fields. *Communications* is recognized as the most trusted and knowledgeable source of industry information for today's computing professional. *Communications* brings its readership in-depth coverage of emerging areas of computer science, new trends in information technology, and practical applications. Industry leaders use *Communications* as a platform to present and debate various technology implications, public policies, engineering challenges, and market trends. The prestige and unmatched reputation that *Communications of the ACM* enjoys today is built upon a 50-year commitment to high-quality editorial content and a steadfast dedication to advancing the arts, sciences, and applications of information technology.

#### STAFF

**DIRECTOR OF PUBLICATIONS**  
Scott E. Delman  
cacm-publisher@cacm.acm.org

#### Executive Editor

Diane Crawford  
**Managing Editor**  
Thomas E. Lambert

#### Senior Editor

Andrew Rosenbloom

#### Senior Editor/News

Lawrence M. Fisher

#### Web Editor

David Roman

#### Rights and Permissions

Barbara Ryan

#### Editorial Assistant

Jade Morris

#### Art Director

Andrij Borys

#### Associate Art Director

Margaret Gray

#### Assistant Art Director

Mia Angelica Balaquiot

#### Production Manager

Bernadette Shade

#### Advertising Sales Account Manager

Ilia Rodriguez

#### Columnists

David Anderson; Michael Cusumano;  
Peter J. Denning; Mark Guzdial;  
Thomas Haigh; Leah Hoffmann; Mari Sako;  
Pamela Samuelson; Marshall Van Alstyne

#### CONTACT POINTS

**Copyright permission**  
permissions@hq.acm.org

#### Calendar items

calendar@cacm.acm.org

#### Change of address

acmhelp@acm.org

#### Letters to the Editor

letters@cacm.acm.org

#### WEBSITE

http://cacm.acm.org

#### WEB BOARD

#### Chair

James Landay

#### Board Members

Marti Hearst; Jason I. Hong;  
Jeff Johnson; Wendy E. MacKay

#### AUTHOR GUIDELINES

http://cacm.acm.org/about-communications/author-center

#### ACM ADVERTISING DEPARTMENT

2 Penn Plaza, Suite 701, New York, NY  
10121-0701  
T (212) 626-0686  
F (212) 869-0481

#### Advertising Sales Account Manager

Ilia Rodriguez  
ilia.rodriguez@hq.acm.org

#### Media Kit acmm mediasales@acm.org

#### Association for Computing Machinery (ACM)

2 Penn Plaza, Suite 701  
New York, NY 10121-0701 USA  
T (212) 869-7440; F (212) 869-0481

#### EDITORIAL BOARD

#### EDITOR-IN-CHIEF

Andrew A. Chien  
aic@cacm.acm.org

#### Deputy to the Editor-in-Chief

Lihan Chen  
cacm.deputy.to.aic@gmail.com

#### SENIOR EDITOR

Moshe Y. Vardi

#### NEWS

#### Co-Chairs

William Pulleyblank and Marc Snir

#### Board Members

Monica Divitini; Mei Kobayashi;  
Michael Mitzenmacher; Rajeev Rastogi;  
François Sillion

#### VIEWPOINTS

#### Co-Chairs

Tim Finin; Susanne E. Hambrusch;  
John Leslie King; Paul Rosenbloom

#### Board Members

Stefan Bechtold; Michael L. Best; Judith Bishop;  
Andrew W. Cross; Mark Guzdial; Haym B. Hirsch;  
Richard Ladner; Carl Landwehr; Beng Chin Ooi;  
Francesca Rossi; Loren Terveen;  
Marshall Van Alstyne; Jeannette Wing;  
Susan J. Winter

#### PRACTICE

#### Co-Chairs

Stephen Bourne and Theo Schlossnagle

#### Board Members

Eric Allman; Samy Bahra; Peter Bailis;  
Terry Coatta; Stuart Feldman; Nicole Forsgren;  
Camille Fournier; Jessie Frazelle;  
Benjamin Fried; Tom Killalea; Tom Limoncelli;  
Kate Matsudaira; Marshall Kirk McKusick;  
Erik Meijer; George Neville-Neil;  
Jim Waldo; Meredith Whittaker

#### CONTRIBUTED ARTICLES

#### Co-Chairs

James Larus and Gail Murphy

#### Board Members

William Aiello; Robert Austin; Kim Bruce;  
Alan Bundy; Peter Buneman; Carl Gutwin;  
Yannis Ioannidis; Gal A. Kaminka;  
Ashish Kapoor; Kristin Lauter; Igor Markov;  
Bernhard Nebel; Lionel M. Ni; Adrian Perrig;  
Marie-Christine Rousset; Krishan Sabnani;  
m.c. schraefel; Ron Shamir; Alex Smola;  
Josep Torrellas; Sebastian Uchitel;  
Hannes Werthner; Reinhard Wilhelm

#### RESEARCH HIGHLIGHTS

#### Co-Chairs

Azer Bestavros and Shriram Krishnamurthi

#### Board Members

Martin Abadi; Amr El Abbadi; Sanjeev Arora;  
Michael Backes; Maria-Florina Balcan;  
David Brooks; Stuart K. Card; Jon Crowcroft;  
Alexei Efros; Bryan Ford; Alon Halevy;  
Gernot Heiser; Takeo Igarashi; Sven Koenig;  
Greg Morrisett; Tim Roughgarden;  
Guy Steele, Jr.; Robert Williamson;  
Margaret H. Wright; Nicholai Zeldovich;  
Andreas Zeller

#### SPECIAL SECTIONS

#### Co-Chair

Sriram Rajamani

#### Board Members

Tao Xie; Kenjiro Taura; David Padua

#### ACM Copyright Notice

Copyright © 2018 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 permissions@hq.acm.org or fax (212) 869-0481.

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; www.copyright.com.

#### Subscriptions

An annual subscription cost is included in ACM member dues of \$99 (\$40 of which is allocated to a subscription to *Communications*); for students, cost is included in \$42 dues (\$20 of which is allocated to a *Communications* subscription). A nonmember annual subscription is \$269.

#### ACM Media Advertising Policy

*Communications of the ACM* and other ACM Media publications accept advertising in both print and electronic formats. All advertising in ACM Media publications is at the discretion of ACM and is intended to provide financial support for the various activities and services for ACM members. Current advertising rates can be found by visiting <http://www.acm-media.org> or by contacting ACM Media Sales at (212) 626-0686.

#### Single Copies

Single copies of *Communications of the ACM* are available for purchase. Please contact [acmhelp@acm.org](mailto:acmhelp@acm.org).

#### COMMUNICATIONS OF THE ACM

(ISSN 0001-0782) is published monthly by ACM Media, 2 Penn Plaza, Suite 701, New York, NY 10121-0701. Periodicals postage paid at New York, NY 10001, and other mailing offices.

#### POSTMASTER

Please send address changes to *Communications of the ACM*  
2 Penn Plaza, Suite 701  
New York, NY 10121-0701 USA

Printed in the USA.



Association for  
Computing Machinery





Vinton G. Cerf

DOI:10.1145/3275378

# The Internet in the 21<sup>st</sup> Century

After working on DARPA-funded projects from 1967–1982, including the design and implementation of the ARPANET and Internet, I left DARPA to go into the private


sector to design and build MCI Mail. At that time, I handed the architectural reins of the Internet to David D. Clark and Jonathan B. Postel as chief Internet architect and deputy Internet architect, respectively. Since that time, Clark and Postel went on to make deeply significant contributions to the Internet's evolution. Postel as the Internet Assigned Numbers Authority and RFC editor and member of the Internet Architecture Board; Clark as the chairman of the Internet Architecture Board (earlier: Internet Activities Board) and as a leader in articulating Internet design principles. Sadly, Jon Postel passed away 20 years ago, October 16, 1998,<sup>a</sup> just as the Internet Corporation for Assigned Names and Numbers (ICANN) was forming. He was to have been its chief technology officer. More recently, David Clark has produced two wide-ranging and deep books about the Internet. One book will be published this month, *Designing an Internet*,<sup>b</sup> and the other, *International Relations in the Cyberage (The Co-Evolution Dilemma)*, will be published later by MIT Press.

These two works capture the depth and breadth of thought the Internet now demands of us on technical and policy grounds. As new methods for exercising the network arrive (think smartphones and the Internet of Things), we are finding new ways to

apply this global system to our daily challenges. Perhaps more seriously, many people are finding ways to do harmful things through the Internet medium. Headlines highlighting abuses abound: Identity theft; electronic funds transfer and automated teller machine heists; point-of-sale terminal hacks; theft of personal information including credit cards, passwords, and other personal information; malware and denial-of-service attacks; bullying; misinformation; election interference; and the exacerbation of social tensions. The list is longer and would take up the rest of this column.

Responses to these abuses have been sporadic at best. Two-factor authentication would remediate many penetration scenarios but is not widely adopted. Operating system and application software weaknesses are not adequately addressed. Corporate attention to these risks is unevenly applied and incentives to do better are in short supply. The social unrest accompanying deliberate misinformation campaigns is finally reaching policy awareness and is leading to demands for response, but legislators are often poorly equipped to produce implementable regulations. ACM has an active US-ACM Public Policy Committee and other ACM Councils are being drawn into discussions about these problems but there is, as yet, little consensus on effective responses. Varying societal norms and conditions make for a wide range of possible reactions, some of which strike me as excessive and hostile to human rights.

The Secretary-General of the United Nations has commissioned a High-Level Panel on Digital Cooperation. I consider this to be an aptly named effort. The charge to the panel is to consider these matters and to make recommendations to deal with them in an internationally cooperative fashion. It is clearly unlikely the panel will solve the problems in general, but it may be able to surface implementable, international, or transnational actions that would reduce the vulnerabilities currently being exploited by individuals, organizations, and nation states.

At the national level, only a small percentage of businesses and individuals are well equipped to defend themselves in the hazardous online world. People must be trained to detect and reject phishing attacks and be more vigilant about cyber hygiene. More information sharing between the national security apparatus and private-sector enterprises seems called for, especially as vulnerabilities and their remedies become apparent. That such a practice would benefit from international cooperation seems likely but fraught with details about implementation. I am looking forward to reading both of Clark's volumes in the expectation that he and his co-authors will throw light in the dark places that have developed in our 21<sup>st</sup>-century Internet. 

Vinton G. Cerf is vice president and Chief Internet Evangelist at Google. He served as ACM president from 2012–2014.

Copyright held by author.

a <https://tools.ietf.org/html/rfc2468>

b D.D. Clark. *Designing an Internet (Information Policy)*. The MIT Press, Cambridge, MA, Oct. 30, 2018. ISBN-10: 0262038609; ISBN-13: 978-0262038607





**Computer Vision**  
February 4 – May 10, 2019

**Organizing Committee:**

- Y. Amit, University of Chicago
- R. Basri, Weizmann Institute
- A. Berg, University of NC
- T. Berg, University of NC
- P. Felzenszwalb, Brown Univ.
- B. Fux Svaiter, IMPA
- S. Geman, Brown University
- B. Gidas, Brown University
- D. Jacobs, University of MD
- O. Veksler, Univ of W. Ontario

**Program Description:**

Computer vision is an inter-disciplinary topic crossing boundaries between computer science, statistics, mathematics, engineering, and cognitive science. Research in computer vision involves the development and evaluation of computational methods for image analysis.

The focus of the program will be on problems that involve modeling, machine learning and optimization. The program will also bridge a gap between theoretical approaches and practical algorithms, involving researchers with a variety of backgrounds.

**Associated Workshops:**

- Theory and Practice in Machine Learning and Computer Vision (February 18 - 22, 2019)
- Image Description for Consumer and Overhead Imagery (February 25 - 26, 2019)
- Computational Imaging (March 18 - 22, 2019)
- Optimization Methods in Computer Vision and Image Processing (April 29 - May 3, 2019)

**icerm.brown.edu**

Brown University  
121 S. Main Street, 11th floor  
Providence, RI 02903  
info@icerm.brown.edu

DOI:10.1145/3273019

# Hennessy and Patterson on the Roots of RISC

**A**WARDING ACM'S 2017 A.M. Turing Award to John Hennessy and David Patterson was richly deserved and long overdue, as described by Neil Savage in his news story "Rewarded for RISC" (June 2018). RISC was a big step forward. In their acceptance speech, Patterson also graciously acknowledged the contemporary and independent invention of the RISC concepts by John Cocke, another Turing laureate, at IBM, as described by Radin.<sup>1</sup> Unfortunately, Cocke, who was the principal inventor but rarely published, was not included as an author, and it would have been good if Savage had mentioned his contribution.

It is noteworthy that RISC architectures depend on and emerged from optimizing compilers. So far as I can tell, all the RISC inventors had strong backgrounds in both architecture and compilers.

**Reference**

1. Radin, G. The 801 minicomputer. *IBM Journal of Research & Development* (1983), 237-246.

**Fred Brooks**, Chapel Hill, NC, USA

**No Inconsistencies in Fundamental First-Order Theories in Logic**

Referring to Martin E. Hellman's Turing Lecture article "Cybersecurity, Nuclear Security, Alan Turing, and Illogical Logic" (Dec. 2017), Carl Hewitt's letter to the editor "Final Knowledge with

**It is noteworthy that RISC architectures depend on and emerged from optimizing compilers.**

Certainty Is Unobtainable" (Feb. 2018) included a number of misleading statements, the most important that: "Meanwhile, Gödel's results were based on first-order logic, but every moderately powerful first-order theory is inconsistent. Consequently, computer science is changing to use higher-order logic." Computer science is based on logic, mostly first-order logic, and programmers make their coding decisions using logic every day. The most important results of logic (such as Kurt Gödel's Incompleteness Theorems) are taught in theory courses and are the fundamentals on which computer science and software engineering are based. No inconsistencies have ever been found in any of the standard first-order theories used in logic, ranging from moderately powerful to very powerful, and none are believed to be inconsistent.

**Harvey Friedman**, Columbus, OH, USA,  
and **Victor Marek**, Lexington, KY, USA

**Author Responds:**

*Powerful first-order theories of intelligent information systems are inconsistent because these systems are not compact, thus violating a fundamental principle of first-order theories. Meanwhile, the properties of self-proof of inferential completeness and formal consistency in higher-order mathematical theories are the opposite of incompleteness and the self-unprovability of consistency Gödel showed for first-order theories. Differing properties between higher-order and first-order theories are reconciled by Gödel's "I'mUnprovable" proposition's nonexistence in higher-order theories. First-order theories are not foundational to computer science, which indeed relies on the opposite of Gödel's results.*

**Carl Hewitt**, Palo Alto, CA, USA

**More Accurate Text Analysis for Better Patient Outcomes**

David Gefen et al.'s article "Identifying Patterns in Medical Records through

Institute for Computational and Experimental Research in Mathematics





## A disease mention could even lack any meaning at all, as it is just part of a template generated by an electronic health records system of a particular provider's care system.

Latent Semantic Analysis" (June 2018) endorsed the latent semantic analysis (LSA) method of text analysis due to its ability to identify links among mentions of medical terms, including the strengths of their relative associations. In practice, however, a single-keyword mention in a clinical narrative note might not represent the true factual meaning of such a mention. Moreover, a disease may be mentioned in the context of being ruled out as a diagnosis or only in the context of documenting family history. A disease mention could even lack any meaning at all, as it is just part of a template generated by an electronic health-records system of a particular provider's care system. And many clinical-narrative notes include content that has been copied and pasted from other notes, possibly inflating the importance of certain mentions thus incorporated into the applicable machine-learning algorithms.

Even incorporating standard International Classification of Diseases (ICD) codes, as defined and published by the World Health Organization, into text-processing methods, as Gefen et al. discussed, could be misleading.

For a variety of everyday conditions (such as insomnia), such codes do not indicate definitively the existence or nonexistence of a particular condition. Another example of ICDs yielding potentially misleading results for

an inaccurately coded disease concerns nonalcoholic fatty liver disease (NAFLD), a common yet underdocumented disease often mentioned in notes without ICD codes indicated. Given also subjective and idiosyncratic physician billing styles, a patient record might include a code for NAFLD, though the code might indicate just a biopsy, despite greater odds that the patient's liver is functioning normally. Incorporating codes without associated dates likewise limits their true meaning and thus reduces their applicability in association studies based on text. A code in a patient's problem list (a standard record indicating the most important health problems a patient might be facing) has a very different meaning from the same code appearing on the same patient's doctor-noted encounter-diagnosis record.

To improve classification, accuracy of text-processing methods focused on health care (such as LSA, as Gefen et al. explored) would strongly benefit from much more specific representations of keywords to more accurately indicate or negate a condition rather than incorporate only single keywords. For instance, instead of noting "hypertension," a one-keyword mention, as in Gefen et al.'s Figure 1, the methods should use specific non-negated and time-dependent expressions like "Current visit: Hypertension is in excellent control" or in the context of a cardiac-related condition, as in Gefen et al.'s Figure 2, "No evidence of coronary artery disease."

LSA and other advanced techniques have the potential to truly represent the level of strength in the connections among textual concepts. However, to deliver accurate results that most serve the patient, the features within them must be more descriptive. Such features should thus be based on commonly used multi-keyword expressions and their variations.

**Uri Kartoun**, Cambridge, MA, USA

*Communications* welcomes your opinion. To submit a Letter to the Editor, please limit yourself to 500 words or less, and send to [letters@cacm.acm.org](mailto:letters@cacm.acm.org).

© 2018 ACM 0001-0782/18/10

## Coming Next Month in COMMUNICATIONS

### CHINA REGION SPECIAL SECTION

Industry and academic leaders from the region share their insights on many of the big trends and hot topics generating excitement throughout China's computing community.

#### A Look at the Design of Lua

#### Skill Discovery in Virtual Assistants

#### Modern Debugging: The Art of Finding a Needle in a Haystack

#### Software Challenges for the Changing Storage Landscape

#### Corp to Cloud: Google's Virtual Desktops

#### Tracking and Controlling Microservice Dependencies

Plus the latest news about sensing earthquakes with optical fiber, the impact of GDRP, and AI explained.

The *Communications* Web site, <http://cacm.acm.org>, features more than a dozen bloggers in the BLOG@CACM community. In each issue of *Communications*, we'll publish selected posts or excerpts.

twitter

Follow us on Twitter at <http://twitter.com/blogCACM>

DOI:10.1145/3264623

<http://cacm.acm.org/blogs/blog-cacm>

## Can We Use AI for Global Good?

*Amir Banifatemi observes how the AI for Good Summit "allowed us to start a dialogue, find a common frame of reference, and decide how our steps would be smart and structured."*



**Amir Banifatemi**  
**Validating Beneficial AI**

<https://cacm.acm.org/blogs/blog-cacm/229283-validating-beneficial-ai/fulltext>

beneficial-ai/fulltext

July 3, 2018

Can the diverse artificial intelligence (AI) community come together to build an infrastructure to advance the United Nation's sustainable development goals (SDGs, <https://sustainabledevelopment.un.org/sdgs>) around the world? Can global projects be developed that begin to address pressing issues surrounding some of our greatest humanitarian challenges to help all?

Those were the goals of the second annual AI for Good Global Summit, the leading United Nations platform for dialogue on Artificial Intelligence held in Geneva, Switzerland, over three days in May.

The conference was organized by the International Telecommunication Union (ITU), the United Nations' specialized agency for information

and communication technology (ICT), in partnership with the XPRIZE Foundation, the Association for Computing Machinery (ACM), and 32 sister UN agencies. The 500+ attendees consisted of a diverse set of multi-stakeholders with wide-ranging expertise—from the individual UN agencies (including everything from UNESCO and UNICEF to The World Health Organization, The World Bank, and UNHCR), AI researchers, public- and private-sector decision-makers, potential financial partners and sponsor organizations.

The focus of the 2018 edition of the AI for Good Summit was to bring together stakeholders prepared to propose practical projects to tackle topics within the 17 SDGs. Inspired by the XPRIZE incentive model, the goal was to present actual proposals in front of attendees to validate feasibility, timing, and how meaningful next steps can be identified. In short, setting actual solutions in motion.

As part of the summit design, AI innovators in attendance were connected with invited public- and

private-sector decision-makers. Four breakthrough tracks—looking at satellite imagery, healthcare, smart cities, and trust in AI—set out to propose AI strategies with supporting projects to advance sustainable development. Teams were guided in this effort by an expert audience representing industry, academia, government, and civil society. Each track proposed projects, as well as introducing existing and future obstacles to the attendees, who then worked collaboratively to take promising strategies forward.

The results were demonstrative of a strong momentum and multi-stakeholder interest in collaboration to identify AI-based solutions with action at their core. The AI for Good Summit has achieved agreement on a community-oriented approach to support 35 projects, fast-tracked so they can be realized in as quickly as six months through a two- or three-year window. Priority projects coming out of each of the event tracks included:

► **Developing Data and AI Commons:** A transversal effort during the three days of the conference was

designed to capture common core principles and opportunities to build a platform enabling beneficial AI. To provide AI to the masses, there is a need to have usable and shareable data in a common format that everyone can access. General datasets and relevant information useful to machine learning specialists is often spread throughout multiple repositories—there is an opportunity to consolidate them to level the playing field. This, for example, can be domain-specific, such as care, treatment, and outcomes for health researchers, historical weather data, satellite imagery, and landmass/ocean temperature figures for agriculture and climate prediction, or city traffic, lighting, and crime statistics for city planners.

Data Commons would offer assemblies of datasets and supporting usage of AI tools, knowledge, and expertise of AI practitioners to launch new AI projects, scale up fast, and contribute new and improved resources to the AI for Good community. Data Commons would provide a foundation of the AI Commons, a global initiative proposed at the conclusion of the AI for Good Summit. AI Commons would help make access to AI capabilities universal and provide the public a platform to solve challenges with AI and drive inclusion.

The AI Commons is expected to be announced in late Q3 with opportunities for all stakeholders to join and participate in its development and deployment.

► **AI-Powered Analysis of Satellite Imagery:** Satellites transmit the equivalent of approximately two billion one-megapixel photographs every day, and AI is the only thing that can let us see the whole world at once. Beyond recording these images, they can create a global real-time database of the world. Three project proposals are focused on agriculture and use of AI-powered satellite imagery analysis to predict and prevent deforestation, pinpoint and track livestock, and provide data analytics to enable micro-insurance to smallholder family farming—small farms that rely mainly on family labor that are seen as the prime driver of agricultural production in developing countries.

**“We are seeing the AI community working together to create an infrastructure for responsible communication, development, and trust.”**

An additional project proposal looks at creating a global service platform—with associated enabling infrastructure and common capabilities—that would allow developers to establish and support immediate scaling of new satellite data projects.

► **AI and Healthcare:** As one of the fastest-growing economic sectors in many countries, scalable technology surrounding the convergence of health and AI is exciting. Fifteen project proposals are moving forward, including predictive projects surrounding vision loss and osteoarthritis, integration and analysis of medical data, AI and healthcare policy, and responses to disease outbreak as well as other medical emergencies. There was also discussion surrounding the creation of a new, open study platform for stakeholders, supported by ITU and the World Health Organization, that would serve as a repository of use cases of AI in healthcare to identify data formats as well as interoperability mechanisms required to amplify their impact.

► **Building Trust in AI:** To build well-earned trust in the long term, Trustfactory.ai is being established as an incubator to research, source, support, and address key dimensions of trust in AI. The research collective is led by Cambridge University and the University of Padova—and stakeholders see this as a second leg of the infrastructure needed to expand AI usage globally.

► **AI and Smart Cities:** With the goal to identify common repositories of best practices, seven project

proposals focus on the development of AI-driven simulations of city environments and bringing a human-centered approach to each vision. The projects support linguistic diversity within cities; the enabling of blockchain-based, citizen-centered decision making; strategies to combat gender imbalance and violence, and the use of AI to enhance the cultural heritage of each city to ensure that there are as many different definitions of a smart city as there are cities in the world. There is also a project to establish a global network—the ‘Internet of Cities’—to share the data, knowledge, and expertise required to replicate successful smart cities around the world.

With collaborative efforts such as these, we are seeing the AI community working together to create an infrastructure for responsible communication, development, and trust. The foundational work that began at the first AI for Good Summit has allowed us to start a dialogue, find a common frame of reference, and decide how our steps would be smart and structured. Our focus this year was to accelerate progress, launching projects that will show tangible results and provide positive impact in key areas.

The cycle is set to continue. The 2019 summit will take stock of progress and will continue the focus on identifying practical ways to identify and implement AI for Good projects.

This is a time of building infrastructure, guidelines, and kicking off focused development of tangible tools to accelerate the beneficial. Using AI for Good is the mantra that is gaining traction with more participation and conversations that make sense, and the conversation is not going to stop. AI innovations constitute one of the platforms that can bring benefits for everyone, and is a platform that can be a public asset for the common good. There is a great need to extend AI to more people and more places in a responsible way. We believe giving the public a common platform can benefit everyone.

Amir Banifatemi is the Group Lead, AI and Frontier Technologies for XPRIZE.

© 2018 ACM 0001-0782/18/10 \$15.00

# SHAPE THE FUTURE OF COMPUTING. JOIN ACM TODAY.

ACM is the world's largest computing society, offering benefits and resources that can advance your career and enrich your knowledge. We dare to be the best we can be, believing what we do is a force for good, and in joining together to shape the future of computing.

## SELECT ONE MEMBERSHIP OPTION

### ACM PROFESSIONAL MEMBERSHIP:

- Professional Membership: \$99 USD
- Professional Membership plus  
ACM Digital Library: \$198 USD (\$99 dues + \$99 DL)
- ACM Digital Library: \$99 USD  
(must be an ACM member)

### ACM STUDENT MEMBERSHIP:

- Student Membership: \$19 USD
- Student Membership plus ACM Digital Library: \$42 USD
- Student Membership plus Print *CACM* Magazine: \$42 USD
- Student Membership with ACM Digital Library plus  
Print *CACM* Magazine: \$62 USD

- Join ACM-W:** ACM-W supports, celebrates, and advocates internationally for the full engagement of women in computing. Membership in ACM-W is open to all ACM members and is free of charge.

Priority Code: CAPP

### Payment Information

Name \_\_\_\_\_

ACM Member # \_\_\_\_\_

Mailing Address \_\_\_\_\_

City/State/Province \_\_\_\_\_

ZIP/Postal Code/Country \_\_\_\_\_

- Please do not release my postal address to third parties

Email \_\_\_\_\_

- Yes, please send me ACM Announcements via email
- No, please do not send me ACM Announcements via email

### Purposes of ACM

ACM is dedicated to:

- 1) Advancing the art, science, engineering, and application of information technology
- 2) Fostering the open interchange of information to serve both professionals and the public
- 3) Promoting the highest professional and ethics standards

Payment must accompany application. If paying by check or money order, make payable to ACM, Inc., in U.S. dollars or equivalent in foreign currency.

- AMEX
- VISA/MasterCard
- Check/money order

Total Amount Due \_\_\_\_\_

Credit Card # \_\_\_\_\_

Exp. Date \_\_\_\_\_

Signature \_\_\_\_\_

Return completed application to:  
ACM General Post Office  
P.O. Box 30777  
New York, NY 10087-0777

Prices include surface delivery charge. Expedited Air Service, which is a partial air freight delivery service, is available outside North America. Contact ACM for more information.

**Satisfaction Guaranteed!**

## BE CREATIVE. STAY CONNECTED. KEEP INVENTING.



Association for  
Computing Machinery

1-800-342-6626 (US & Canada)  
1-212-626-0500 (Global)

Hours: 8:30AM - 4:30PM (US EST)  
Fax: 212-944-1318

acmhelp@acm.org  
acm.org/join/CAPP



## Floating Voxels Provide New Hope for 3D Displays

*In search of holograms that can be viewed from any angle.*

**F**EW MOVIE SCENES have had such an effect on display-technology research and development as the droid R2D2 projecting a three-dimensional (3D) image of Princess Leia pleading for help in 1977's blockbuster film *Star Wars*. Numerous engineers have wondered just how they might achieve that effect, of an image you can see from any angle, in real life.

Even The Walt Disney Company, which bought Lucasfilm and the distribution rights for the movie franchise in 2012, is among those with engineers working on the idea.

Two years ago, Daniel Joseph and colleagues in entertainment giant Disney's Burbank, CA-based research and development operation filed for a patent on a projector intended to display floating 3D images. The U.S. patent

points to an anticipated implementation of having the 3D image seem to be standing on an illuminated pedestal, similar to the game table on the Millennium Falcon that appears in a scene later in *Star Wars*.

The Disney system suffers from a problem that is shared with similar systems: the image is formed from an array of light sources fed through beam splitters and mirrors some distance



IMAGE © LUCASFILM LTD. & TM. ALL RIGHTS RESERVED.

behind the pedestal, which limits the viewing angle to those looking toward the projection optics, and so cannot emulate the movies.

Daniel Smalley, an assistant professor of electrical and computer engineering at Brigham Young University, says, “Like many in the holography field, I felt that holograms would provide the 3D images of the future, but the annoying issue is you have to be looking in the direction of the screen that generates them. It’s counter to what you expect 3D displays to do in the future.”

Builders of volumetric displays that can be viewed from any angle face their own challenge. “Fundamentally, you have the problem that photons will just keep traveling until they bounce off something,” says V. Michael Bove, principal research scientist and head of the object-based media group at the Massachusetts Institute of Technology.

Systems such as the VX1 built by Australian company Voxon Photonics use a fast-moving sheet to provide a reflective surface for photons. At a high-enough speed, the sheet will seem to disappear, but bright lights bounced off it will persist to the viewer; the result is the illusion of a slightly translucent 3D object floating in space. Bove says the need to move the sheet at high speed makes this an intrinsically noisy option, and one likely to suffer from mechanical wear.

Another option is to disperse particles

**All volumetric displays to date share the same problem, Smalley says. “You don’t have the self-occlusion to make objects that look realistic.”**

into the air and illuminate them. A team led by John Howell, a professor of physics and optics based at the University of Rochester, used cesium vapor to create the voxels in their experimental volumetric display; the cesium atoms glow where the light from two steerable lasers cross. Yet in these displays, moving parts and poisonous particles need to be encapsulated in a transparent dome or sphere.

“What’s of increased interest is not have a display in the table but to interact with it in a meaningful way. Volumetric displays do have this talking-head-in-a-jar character that works against that. You have the sense that this imagery is bottled up,” Bove says.

Smalley also sees interaction as key, citing another Disney movie franchise, *Iron Man*, as additional inspiration for

his move away from holographic technologies. In the first installment of the movie series, protagonist Tony Stark uses a 3D projector not just to visualize the elements of his powered suit, but also to create a virtual gauntlet around his hand.

Smalley’s team overcame the need to encapsulate their display by trapping and moving a single dust-sized particle. The prototype uses an ultraviolet laser taken from a Blu-ray player to capture and move the piece of dust. A visible-light source tracks and illuminates it. Physicists have yet to develop a theory that fully explains the process of such photophoretic trapping, but it appears to rely on local heating from being struck by photons. Gas molecules hitting the hotter surface acquire more kinetic energy as they bounce off, pushing the particle away.

Says Smalley, “On average it doesn’t work very well at all, but in the [statistical] tails you see incredible behavior. The particle just stays there. You can even blow on it gently. We had one particle trapped in there for 15 hours. It could have stayed for longer: we had to switch the machine off.”

The particle’s composition seems to be crucial. Smalley’s team settled on black liquor—a by-product of the paper-making process—after trying numerous candidates. “I do not believe we can say this is definitively the

## ACM News

# Hijacking the Cryptomine

The gold rush in cryptocurrencies has led cybercriminals to adopt new tactics.

Cybersecurity provider Symantec says the profitability of ransomware dropped in 2017 from an average \$1,017 in 2016 to \$522 per ransomware event. That’s why many cybercriminals have shifted to using coin miners, software designed to mine cryptocurrencies.

Infecting the computing devices of others in order to amass the processing power needed to mine cryptocurrencies is called cryptojacking. Symantec recently reported the detection of coin miners on endpoint computers had increased 8,500% in 2017.

The risk of being caught cryptojacking is minimal; it is difficult to trace because of the anonymity of cryptocurrencies. Also, cryptojacking scripts do not damage computers or data, and nothing is stolen (except processing power), so there is little incentive to follow up when an attack is discovered.

A common method of cryptojacking involves executing a JavaScript in a browser, stealing resources from the user’s CPU, which are pooled with resources from other cryptojacked devices to mine cryptocurrencies.

Browser-based cryptojacking doesn’t require a download, starts instantly, and works efficiently and

surreptitiously in the background; usually, until the browser session is closed. Sometimes hackers will launch a stealth “pop-under” window or a tiny one-pixel browser to continue illicitly accessing a device’s processing power.

Victims might be unaware they have been cryptojacked. The effects are mostly performance-related, and include lags in computers’ execution of commands, slower performance, and overheating.

Most antivirus software and ad blockers can now detect coin-mining software. Browser extensions like No Coin or minerBlock, and JavaScript blockers like NoScript, can be installed to

defeat cryptojacking.

Legitimate uses of cryptojacking are beginning to appear online. For instance, digital media outlet Salon started a beta test early this year, using Coinhive to mine the open source cryptocurrency Monero as an alternative to online advertising as a revenue stream. If a visitor has an ad blocker turned on when visiting Salon.com, they might see a prompt to either disable the ad blocker or select a “suppress ads” option. The latter choice allows Salon to put readers’ unused computing power to use mining Monero while they are visiting the site.

—John Delaney is a freelance writer based in Queens, NY, USA.

best material. It seems unlikely that it is,” he says.

It is possible to produce freestanding volumetric images without injecting particles into the air. More than a decade ago, Hidei Kimura, founder and CEO of Japanese company Burton Inc., and Taro Uchiyama of Keio University found that when focused on specific points, microsecond bursts of high-intensity infrared light could cause air molecules to become glowing plasma. Kimura envisaged the technology being used to create levitating signs above head height for use in emergencies; the bursts would be intense enough to burn the hand of a user foolish enough to try to touch the glowing voxels.

Much shorter pulses could yield a safer system. Yoichi Ochiai of the University of Tsukuba and Kota Kumagai of the University of Utsunomiya in Japan showed at the ACM SIGGRAPH conference in 2015 the results of a prototype based on lasers that fire bursts no more than 100 femtoseconds long. According to Ochiai, users would simply get a tingling sensation from touching the plasma voxels, though users would need to be careful to not let their eyes get too close to the images, as retinal damage is a distinct possibility. Robert Stone, professor of interactive multimedia systems at the University of Birmingham in the U.K., says he has concerns over the eye forming strong afterimages because of the brightness of the plasma.

The plasma projector has the advantage of being far more resistant to disturbance by moving hands than the particle-based option. However, all volumetric displays to date have a common problem, Smalley says: “It is like taking a bunch of fireflies and organizing them into patterns. Everything looks like a ghost. You don’t have the self-occlusion to make objects that look realistic.

“We want to be able to take a point and have it shine light in only one direction. That would mean it begins to look solid.”

The lack of self-occlusion in the optical-trap display is, for the moment, a secondary issue. It is difficult to move the single particle that flies around the Brigham Young display any faster than is possible today; that limits its cover-

**“The general public has for 40 years been seeing cinematic depictions of physically impossible things, and when they do see what’s possible, they’re disappointed.”**

age to a volume the size of a ping-pong ball, and the results demonstrated so far are based on long-exposure images that took up to a minute to generate.

Says Barry Blundell, senior lecturer in computing at the University of Derby in the U.K. and a researcher into volumetric displays since the late 1980s, “With the optical-trap display, I would have to see images generated a lot faster. The only way to do that is parallelism; you’ve got to have more lasers surrounding the display, and more particles. The problem could be that you need to have so much physical apparatus that you lose the viewing freedom.”

Smalley claims the technology exists to drive and illuminate a collection of particles in the shape of the spatial light modulator, the same kind of device as that used to research holographic displays and optical computers. Bove argues the laser and light-modulator components needed for scaled-up displays are now relatively cheap.

Still, expectations may be set too high.

“The general public has for 40 years been seeing cinematic depictions of physically impossible things, and when they do see what’s possible, they are disappointed,” says Bove.

Smalley concedes, “At this stage, you don’t have to be an expert to realize that this isn’t the Princess Leia display you are looking for. But, if given the opportunity to be developed further, I don’t think you would be disappointed.”

Researchers may be trying too hard to make fact out of fiction. “What some of the people working on volumetrics haven’t realized is that the key ele-

ments are complex movement and dynamics, not super-high resolution,” Blundell argues.

Smalley envisages applications where the user needs to inspect the shape closely and move around it. The ability to produce mid-air streamers in fluid-dynamics simulations and models of organs to help with planning medical operations seem good examples. “A lot of 3D technologies can’t give you a strong spatial sense when you get up close. With ours, you can,” he says.

Bove says by looking closely at requirements for target applications and working with user-interface designers, the developers of volumetric displays can move from experiment to market more easily. “Can it be behind a transparent barrier? Is it important that it be viewable from any angle or is 90 degrees OK? Is it acceptable for it to have moving parts?” he suggests as questions to be asked.

Developing volumetric technologies for specific applications may lead to the problem of no individual market being large enough to support research and development, but such displays look more technologically feasible, Bove says. “The problem with the Leia display is that it needs all of the boxes to be ticked.”

#### Further Reading

*Smalley, D.E. et al*

**A Photophoretic-Trap Volumetric Display, *Nature*, 553, pp486–490 (25 January 2018), doi:10.1038/nature25176**

*Ochiai, Y., Kumagai, K., Hoshi, T., Rekimoto, J., Hasegawa, S., and Hayasaki, Y.*

**Fairy Lights in Femtoseconds: Aerial and Volumetric Graphics Rendered by Focused Femtosecond Laser Combined with Computational Holographic Fields, *ACM Transactions on Graphics*, Volume 35, Issue 2, (May 2016), doi:10.1145/2850414**

*Blundell, B.*

**On the Uncertain Future of the Volumetric 3D Display Paradigm, *3D Research*, 8 (2) p11, doi:10.1007/s13319-017-0122-2**

*Joseph, D.M., Smoot, L.S., Smithwick, Q.Y., and Ilardi, M.J.*

**Retroreflector Display System for Generating Floating Image Effects, U.S. Patent Application 2018/0024373 A1 (25 January 2018)**

**Chris Edwards** is a Surrey, U.K.-based writer who reports on electronics, IT, and synthetic biology



# Transient Electronics Take Shape

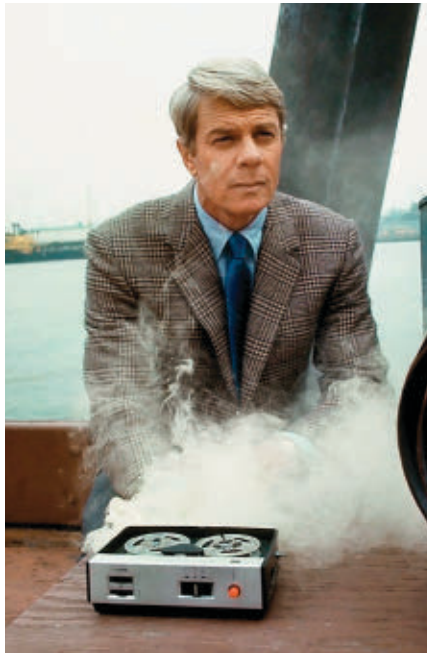
*Advances in materials science and chemistry are leading to self-destructing circuits and transient electronics, which could impact many fields.*

ONE OF THE intriguing aspects of the popular 1960s television show “Mission Impossible” was the opening sequence of every episode, which featured a secret agent listening to a recorded message about an upcoming mission. At the end of the recording each week, the tape would sizzle, crackle, and disintegrate into a heap of smoke and debris, ensuring no one else could access the top-secret information it contained.

Until recently, self-destructing electronic systems remained within the realm of science fiction, but advances in chemistry, engineering, and materials science are finally allowing researchers to construct circuits that break down on their own timetable. This includes systems that rely on conventional complementary oxide semiconductor (CMOS) technology.

“The goal is to develop functional circuits that can operate for a period of time and then vaporize,” says Amit Lal, Robert M. Scharf 1977 Professor of Engineering in the Electrical and Computer Engineering Department at Cornell University in Ithaca, NY, and director of the university’s SonicMEMS lab. “It’s the Biblical ashes-to-ashes concept applied to electronics.”

The technology could reshape numerous fields, including medicine, agriculture, and the military. It could also reduce environmental damage caused by materials in semiconductors and electronics, which require recycling and too often wind up in landfills and water supplies. Already, Lal and a team at Cornell have obtained a patent for water-soluble circuits that biodegrade without leaving toxic materials behind. Other researchers at Northwestern University and the University of Houston have built self-



**The self-destructing audio tape of the “Mission Impossible” television show anticipated by decades the advent of self-destructing electronics.**

destructing circuits that could be used in smartphones, drones, and even inside the human body.

While the technology is still in the early stages of development, it could have a commercial impact within a few years. For now, the biggest obstacles revolve around perfecting transient electronics and self-destructing circuits and scaling them for mass use. There’s also a need to gain a deeper understanding of polymers and composite materials, and to ensure these systems fully vaporize without leaving traces of toxic chemicals. As Lal explains, “It’s not easy to design a circuit that works perfectly and delivers a high level of performance for a period of time, and then make it vaporize in the desired situation or at a precise moment, and within a relatively short period of time.”

## Mission: Possible

There are myriad possible uses for self-destructing electronic circuits. For example, the technology would allow farmers to place monitoring devices in a field and not have to worry about removing them later. Using different materials or combinations of materials that avoid toxic residue ranging from Tungsten to formulated polymers, the circuits would simply disintegrate at a certain point. The remaining material would have little or no impact on the environment.

The same technology also would let doctors insert biomedical devices into the human body to dispense medicine in a controlled way; in some cases, such as with chemotherapy, such micro-targeting of cells could dramatically reduce side effects, and there would be no need to surgically remove the device at the end of treatment.

Transient electronics could allow the military to deploy drones, robots, and other electronic devices into the field without the worry adversaries could recover them and benefit in any way.

The environmental benefits of self-destructing circuits are also obvious, considering tens of millions of tons of e-waste are generated every year, and toxic substances including mercury, lead, cadmium and arsenic are not always recycled, or completely destroyed during incineration. In some cases, e-waste winds up in landfills, particularly in developing nations. The resulting toxins that leach into the soil, air, and water create health hazards that can result in neurological damage, reproductive disorders, and cancers.

New types of designs and encapsulation layers will allow electronic systems formed with specialized materials to operate in a stable, high-performance manner for a prescribed period and



then to degrade and disappear completely, at a molecular level, to biocompatible and environmentally compatible end products. “The ability to reduce even some electronic waste could be highly beneficial,” explains Ved Gund, a senior process engineer at Intel who collaborated with Lal on the development of a destructible circuit while he was a graduate student at Cornell.

The common denominator among all transient electronics is an ability to make a device physically vanish through a controlled process, often triggered by events based on external environmental cues. These could take the form of electronic signals, light, temperature, shock or pressure changes, and chemical processes (including enzymes released by the human body). It may mean programming different functions into a device at different stages—essentially physically morphing a system through an evolutionary process—or creating different devices within a device for a specific purpose.

“You can achieve physical transience in many different ways,” explains John Rogers, Louis Simpson and Kimberly Querrey Professor of Materials Science and Engineering, Biomedical Engineering, and Neurological Surgery at Northwestern University in Evanston, IL. This is important, he adds, because it allows transient electronics and destructible circuits to be used in many different ways and in many different environments, ranging from harsh industrial conditions to inside the human body.

Although the idea of producing transient electronics is nothing new, the technology began to emerge over the last decade, and Rogers is one of the pioneers in the field. In 2007, as a member of the U.S. Defense Department’s Defense Science Research Council, he began collaborating with the Defense Advanced Research Projects Agency (DARPA) on ways to produce electronics that can adopt a transient physical form. The thinking at the time was simple, even if executing on the concept was extraordinarily difficult. “Ideally, you flip a switch or push a button remotely and the device simply melts away, disintegrates, or vanishes, rather than falling into the hands of an adversary,” he explains.

In 2009, Rogers published an aca-

## Self-destructing technology could reshape numerous fields, as well as reducing environmental damage from materials in electronics.

demically paper outlining how a partially transient system with substrates built atop a thin and fragile electronic circuit could be water soluble. The research started Rogers and fellow scientists down a path toward building more sophisticated circuits and devices using environmentally benign end-products. Their focus has revolved primarily around military and medical applications, with the goal of developing circuits and other electronics that self-destruct and leave no trace of their component materials.

In 2017, Rogers and colleagues announced more advanced ways to build state-of-the-art silicon complementary metal-oxide-semiconductor (CMOS) foundries to produce high-performance, water-soluble forms of electronics.

### Short Circuits

Researchers have continued to push the boundaries of transient and self-destructive electronics and circuits. For instance, Cunjiang Yu, Bill D. Cook Assistant Professor of Mechanical Engineering at the University of Houston in Texas, along with researchers in China, have developed self-destructive electronics with copper, magnesium oxide, and indium gallium zinc oxide supported on a polyanhydride substrate. Water vapor breaks down the polymer substrate and eventually causes the electronic materials to dissolve. Yu’s research is significant because it is the first known approach that directly utilizes the substrate as the mechanism triggering the dissolution of the electronics.

## ACM Member News

### USING BIG DATA TO IMPROVE LIVES



When she had her first class in programming in college, “I thought I was the only person who had never

programmed before,” recalls Nuria Oliver, director of Data Science Research at multinational telecom company Vodafone. Undeterred, she wound up at the top of her class. That experience, she says, motivates her to this day.

Oliver is passionate about inspiring more women to study science, technology, engineering, and math (STEM) topics in school and pursuing careers in research technology, as well as encouraging them to persevere and not quit jobs in these fields. “Anyone—even if you have no experience—can do anything if you apply yourself,” she says, and offers her experience as proof.

Throughout her career, Oliver has been interested in using artificial intelligence and machine learning to better understand human behavior, with the goal of building technology that is meaningful in peoples’ lives.

She received her undergraduate degree in electrical engineering and computer science from the Universidad Politécnica de Madrid, Spain, in 1994. After earning her Ph.D. in Perceptual Intelligence from the Massachusetts Institute of Technology in 2000, Oliver spent seven years with Microsoft Research in Redmond, WA, until she was offered the opportunity to become the first female scientific director at Telefonica R&D, in Barcelona, Spain, modeling human behavior from mobile data.

In late 2016, Oliver was also named chief data scientist for DataPop Alliance, an international non-profit organization devoted to leveraging big data to improve the world. Early last year, she joined Vodafone to lead its global research agenda to analyze mobile data to better understand what people want and need from their mobile phones.

—John Delaney

The Cornell group, in conjunction with Honeywell Aerospace, has explored self-destructing technology by experimenting with a number of different approaches, including systems that use liquids and signals to trigger the disintegration process. In one instance, they created a circuit with microscopic cavities of novel polymers containing sodium bifluoride and rubidium. Exposing the shell to radio waves of a specific frequency triggers graphene-on-nitride micro-valves in the shell to open, allowing the alkali metals to oxidize and produce a thermal reaction that causes an already thinned-out chip to disintegrate and vaporize rapidly. “The technique uses the metals in the chips as an energy source. They are attached to a special polymer that reacts to the heat,” Lal explains.

The disintegration process is triggered by a tiny block that measures 0.04 inches wide. After the electronics disintegrate, the result is a fine powder consisting of cesium and rubidium oxides, sand-like particles from the silicon chip, and tiny flakes of carbon from the graphene, along with the remaining battery (the research team is also working on a way to make the battery vaporizable, too). “The project requires ongoing research into polymers and how to optimize both mechanical and materials functions,” Intel’s Gund says. One area of particular interest is how to use flexible layers of a material substrate to produce a circuit that operates like conventional silicon electronics, while using plastics and other materials that can also be broken down or recycled using the vaporization process.

Meanwhile, Rogers and his research group have focused on engineering a system that could wirelessly deliver programmable drug doses to a specific part of the body, then naturally degrade and disappear. This technology might be used to deliver medication post-surgery, for example. The challenge of this approach, Rogers says, “is that we have to build a device that is very stable over a relevant time period but then is ultimately completely unstable, in the sense that it eventually vanishes without a trace.” The team is working to perfect a silicon, magnesium, magnesium oxide, and silk circuit that dissolves in the body in much the same way that absorbable sutures vanish after minor

## Rogers and his research group are designing a system to deliver programmable drug doses to a specific part of the body, then naturally degrade and disappear.

surgeries. This involves using mixtures of chemicals and polymers that cause disintegration and packing them into layers with electrodes that will trigger the destruction process.

### Materially There

Developing new types of circuits and electronics that self-destruct requires rethinking and redesigning semiconductors that have never been engineered for anything other than maximum performance over a desired lifespan, Gund says. Adding to the task: the design and engineering process can vary greatly, depending on the desired performance and results. A biomedical device may require 10 weeks of high-performance operation before it is made to degrade and dissolve into the body, while a military device might be required to disintegrate in a matter of seconds. What’s more, depending on the device and how it used, the trigger mechanism might vary.

Researchers continue to explore how different combinations of chemicals and substances interact to produce a desired result, and how they can get to the point where there is little or no trace of the circuit or electronic component. So, far, most of the research has been conducted through trial and error and testing different combinations of materials together. In the future, Yu says, machine learning might also serve as a valuable tool for sorting through growing mountains of data and discovering combinations that can be used for different types of circuits and in different situations.

“These projects require an interdisciplinary approach and experimentation with a lot of different chemicals and materials,” Yu explains. “We are only beginning to understand how to build these self-destructive electronics and engineer the desired systems.”

Nevertheless, the field continues to advance and commercialization of the technology could take place within the next few years. In the future, inexpensive and disposable circuits could also introduce new types of devices and systems used within the Internet of Things (IoT). Low power requirements could support vast networks of connected circuits that could operate for years and pose no environmental hazard.

Says Rogers: “Transient electronics are beginning to take shape in a tangible way. The technology will almost certainly impact a wide range of areas in the years to come.”

### Further Reading

- Gund, V., Ruyack, A., Camera, K., Ardanuc, S., Ober, C., and Lal, A. (2015). **Multi-modal graphene polymer interface characterization platform for vaporizable electronics**. 2015. 873-876. 10.1109/MEMSYS.2015.7051098. [https://www.researchgate.net/publication/283633594\\_Multi-modal\\_graphene\\_polymer\\_interface\\_characterization\\_platform\\_for\\_vaporizable\\_electronics](https://www.researchgate.net/publication/283633594_Multi-modal_graphene_polymer_interface_characterization_platform_for_vaporizable_electronics)
- Gund, V., Ruyack, A., Camera, K., Ardanuc, S., Ober, C., and Lal, A. (2016). **Transient Micropackets for Silicon Dioxide and Polymer-Based Vaporizable Electronics**. 1153-1156. 10.1109/MEMSYS.2016.7421840. [https://www.researchgate.net/publication/301709792\\_Transient\\_micropackets\\_for\\_silicon\\_dioxide\\_and\\_polymer-based\\_vaporizable\\_electronics](https://www.researchgate.net/publication/301709792_Transient_micropackets_for_silicon_dioxide_and_polymer-based_vaporizable_electronics).
- Chang, J., Fang, H., Bower, C.A., Song E., Yu, X., and Rogers, J.A. **Materials and processing approaches for foundry-compatible transient electronics**. *Proceedings of the National Academy of Sciences* Jul 2017, 114 (28) E5522-E5529; DOI: 10.1073/pnas.1707849114. <http://www.pnas.org/content/114/28/E5522>
- Gao, Y., Zhang, Y., Wang, X., Sim, K., Liu, J., Chen, J., Feng, X., Xu, H., and Yu, C. **Moisture-triggered physically transient electronics**. *Science Advances*, Sept. 1, 2017: Vol. 3, no. 9, e1701222 DOI: 10.1126/sciadv.1701222. <http://advances.sciencemag.org/content/3/9/e1701222.full>.

Samuel Greengard is an author and journalist based in West Linn, OR, USA.

© 2018 ACM 0001-0782/18/10 \$15.00

# The Dangers of Automating Social Programs

*Is it possible to keep bias out of a social program driven by one or more algorithms?*

**A**SK POVERTY ATTORNEY Joanna Green Brown for an example of a client who fell through the cracks and lost social services benefits they may have been eligible for because of a program driven by artificial intelligence (AI), and you will get an earful.

There was the “highly educated and capable” client who had had heart failure and was on a heart and lung transplant wait list. The questions he was presented in a Social Security benefits application “didn’t encapsulate his issue” and his child subsequently did not receive benefits.

“It’s almost impossible for an AI system to anticipate issues related to the nuance of timing,” Green Brown says.

Then there’s the client who had to apply for a Medicaid recertification, but misread a question and received a denial a month later. “Suddenly, Medicaid has ended and you’re not getting oxygen delivered. This happens to old people frequently,” she says.

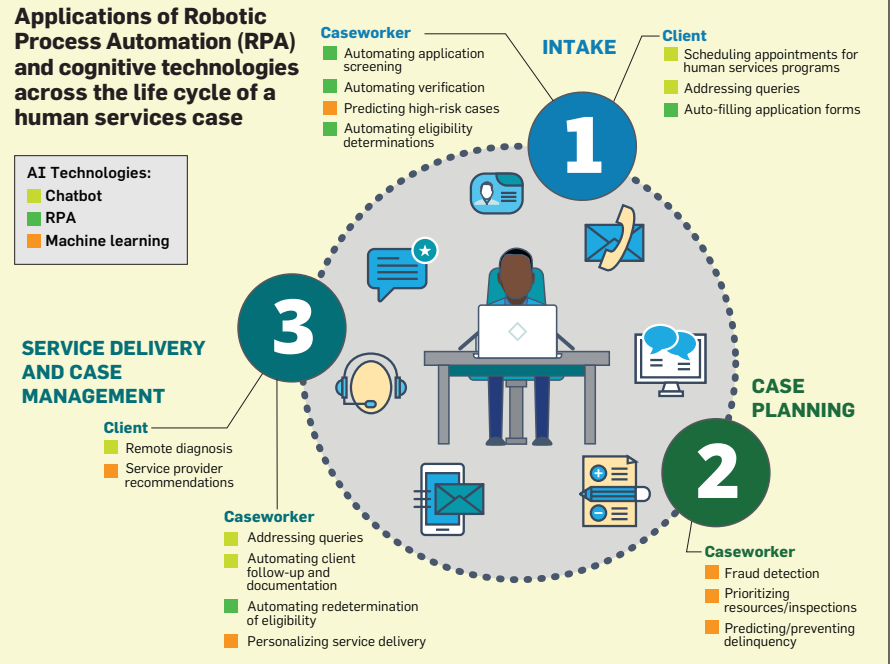
Another client died of cancer that Green Brown says was preventable, but the woman did not know social service programs existed, did not have an education, and did not speak English. “I can’t say it was AI-related,” she notes, “but she didn’t use a computer, so how is she going to get access to services?”

Such cautionary tales illustrate what can happen when systems become automated, the human element is removed, and a person in need lacks a support system to help them navigate the murky waters of applying for government assistance programs like Social Security and Medicaid.

There are so many factors that go into an application or appeals process for social services that many people just give up, Green Brown says. They can also lose benefits when a line of questioning ends in the system, but

## Applications of Robotic Process Automation (RPA) and cognitive technologies across the life cycle of a human services case

**AI Technologies:**  
 ■ Chatbot  
 ■ RPA  
 ■ Machine learning



which may not tell their whole story. “The art of actual conversation is what teases out information,” she says. A human can tell something isn’t right simply by observing a person for a few minutes; determining why they are uncomfortable, for example, and whether it is because they have a hearing problem, or a cognitive or psychological issue.

“The stakes are high when it comes to trying to save time and money versus trying to understand a person’s unique circumstances,” Green Brown says. “Data is great at understanding who the outliers are; it can show fraud and show a person isn’t necessarily getting all benefits they need, but it doesn’t necessarily mean it’s correct information, and it’s not always indicative of eligibility of benefits.”

There are well-documented examples of bias in automated systems used to provide guidelines in sentencing criminals, predicting the likeli-

hood of someone committing a future crime, setting credit scores, and in facial recognition systems. As automated systems relying on AI and machine learning become more prevalent, the trick, of course, is finding a way to ensure they are neutral in their decision-making. Experts have mixed views on whether they can be.

AI-based technologies can undoubtedly play a positive role in helping human services agencies cut costs, significantly reduce labor, and deliver faster and better services. Yet taking the human element out of the equation can be dangerous, agrees the 2017 Deloitte report “AI-augmented human services: Using cognitive technologies to transform program delivery.”

“AI can augment the work of caseworkers by automating paperwork, while machine learning can help caseworkers know which cases need urgent attention. But ultimately, humans are the users of AI systems, and these sys-



tems should be designed with human needs in mind,” the report states. That means they first need to determine the biggest pain points for caseworkers, and the individuals and families they serve. Issues to factor in are what are the most complex processes; can they be simplified; what activities take the most time and whether they can be streamlined, the report suggests.

Use of these systems is in the early stages, but we can expect to see a growing number of government agencies implementing AI systems that can automate social services to reduce costs and speed up delivery of services, says James Hendler, director of the Rensselaer Institute for Data Exploration and Applications and one of the originators of the Semantic Web.

“There’s definitely a drive, as more people need social services, to bring in any kind of computing automation and obviously, AI and machine learning are offering some new opportunities in that space,” Hendler says.

One of the ways an AI system can be beneficial is in instances in which someone seeking benefits needs to access cross-agency information. For example, if someone is trying to determine whether they can get their parents into a government-funded senior living facility, there are myriad questions to answer. “The potential of AI and machine learning is figuring out how to get people to the right places to answer their questions, and it may require going to many places and piecing together information. AI can help you pull it together as one activity.”

One of the main, persistent problems these systems have, however, is inherent bias, because data is input by biased humans, experts say.

Just like “Murphy’s Law,” which states that “anything that could go wrong, will,” Oren Etzioni, chief executive officer of the Allen Institute for Artificial Intelligence, says there’s a Murphy’s Law for AI: “It’s a law of unintended consequences, because a system looks at a vast range of possibilities and will find a very counter-intuitive solution to a problem.”

“People struggle with their own biases, whether racist or sexist—or because they’re just plain hungry,” he says. “Research has shown that there are [judicial] sentencing differences based on the time of day.”

Machines fall short in that they have no “common sense,” so if a data error is input, it will continue to apply that error, Etzioni says. Likewise, if there is a pattern in the data that is objectionable because the data is from the past but is being used to create predictive models for the future, the machine will not override it.

“It won’t say, ‘this behavior is racist or sexist and we want to change that’; on the contrary, the behavior of the algorithm is to amplify behaviors found in the data,” he says. “Data codifies past biases.”

Because machine learning systems seek a signal or pattern in the data, “we need to be very careful in the application of these systems,” Etzioni says. “If we are careful, there’s a great potential benefit as well.”

To make AI and machine learning systems work appropriately, many cognitive technologies need to be trained and retrained, according to the Deloitte report. “They improve via deep learning methods as they interact with users. To make the most of their investments in AI, agencies should adopt an agile approach [with software systems], continuously testing and training their cognitive technologies.”

David Madras, a Ph.D. student and machine learning researcher at the University of Toronto (U of T), believes if an algorithm is not certain of something, rather than reach a conclusion, it should have the option to indicate uncertainty and defer to a human.

Madras and colleagues at U of T developed an algorithmic model that includes fairness. The definition of fairness they used for their model is based on “equalized odds,” which they found in a 2016 paper, “Equality of Opportunity in Supervised Learning,” by computer scientists from Google, the University of Chicago, and the University of Texas, Austin. According to that paper, Madras explains, “the model’s false positive and false negative rates should be equal for different groups (for example, divided by race). Intuitively, this means the types of mistakes should be the same for different types of people (there are mistakes that can advantage someone, and mistakes that can disadvantage someone).”

The U of T researchers wanted to examine the unintended side effects of machine learning in decision-making

**“Humans are better than computers at exploring those grey areas around the edges of problems. Computers are better at the black-and-white decisions in the middle.”**

systems, since a lot of these models make assumptions that don’t always hold in practice. They felt it was important to consider the possibility that an algorithm could respond “I don’t know” or “pass,” which led them to think about the relationship between a model and its surrounding system.

“There is often an assumption in machine learning that the data is a representative sample, or that we know exactly what objective we want to optimize.” That has proven not to be the case in many decision problems, he says.

Madras acknowledges the difficulty of knowing how to add fairness to (or subtract unfairness from) an algorithm. “Firstly, unfairness can creep in at many points in the process, from problem definition, to data collection, to optimization, to user interaction.” Also, he adds, “Nobody has a great single definition of ‘fairness.’ It’s a very complex, context-specific idea [that] doesn’t lend itself easily to one-size-fits-all solutions.”

The definition they chose for their model could just as easily be replaced by another, he notes.

In terms of whether social services systems can be unbiased when the algorithm running them may have built-in biases, Madras says that when models learn from historical data, they will pick up any natural biases, which will be a factor in their decision-making.

“It’s also very difficult to make an algorithm unbiased when it is operating in a highly biased environment; especially when a model is learned



from historical data, the tendency is to repeat those patterns in some sense,” Madras says.

Etzioni believes an AI system can be bias-free even when bias is input, although that is not an easy thing to achieve. An original algorithm tries to maximize consistency with data, he says, but that past data may not be the only criteria.

“If we can define a criterion and mathematically describe what it means to be free of bias, we can give that to the machine,” he says. “The challenge becomes describing formally or mathematically what bias means, and secondly, you have to have some adherence to the data. So there’s really a tension between consistency with the data, which is clearly desirable, and being bias-free.”

People are working so both consistency and being bias-free can be supported, he adds.

For AI to augment the work of government case workers and make social programs more efficient is to couple the technical progress being made with educating people on how to use these programs, Etzioni says.

“Part of the problem is when a human just blindly adheres to the recommendations of the system without trying to make sense of them, and the system says, ‘It must be true,’ but if the machine’s analysis is one output and a sophisticated person analyzes it, we find ourselves in the best of both worlds.”

AI, he says, really should stand for “augmented intelligence,” where technology plays a supporting role, he says.

“Humans are better than computers at exploring those grey areas around the edges of problems,” agrees Hendler. “Computers are better at the black-and-white decisions in the middle.”

The issue of transparency of algorithms and bias was discussed at a November 2017 conference held by the Paris-based Organization for Economic Cooperation and Development (OECD). Although several beneficial societal use-cases of AI were mentioned, researchers said the solution lies in addressing system bias from a policy perspective as well as a design perspective.

“Right now, AI is designed so as to optimize a given *objective*,” the

researchers stated. “However, what we should be focusing on is designing AI that delivers *results* that are in line with peoples’ well-being. By observing human reactions to various outcomes, AI could learn through a technique called ‘cooperative inverse reinforcement learning’ what our preferences are, and then work towards producing results consistent with those preferences.”

AI systems need to be held accountable, says Alexandra Chouldechova, an assistant professor of statistics and public policy at Carnegie Mellon University’s Heinz College of Information Systems and Public Policy.

“Systems fail to achieve their purported goals all the time,” Chouldechova notes. “The questions are: Why? Can it be fixed? Could it have been prevented in the first place?”

“By being clear about a system’s intended purpose at the outset, transparent about its development and deployment, and proactive in anticipating its impact, we can hopefully reach a place where there will be fewer adverse unintended consequences.”

For the foreseeable future, Hendler believes humans and computers working together will outperform either one separately. For the partnership to work, a human must be able to understand the decision-making of the AI system, he says.

“We currently teach people to take the data and feed it into AI systems to get an ‘unbiased answer.’ That unbiased answer is used to make predictions and help people find services,” Hendler says. “The problem is, the data coming in has been chosen in various ways, and we don’t educate computer or data scientists how to know the data in your database will model the real world.”

This is certainly not a new problem. Hendler recalls the famous case of Stanislov Petrov, a Soviet lieutenant-colonel whose job was to monitor his country’s satellite system. In 1983, the computers sounded an alarm indicating the U.S. had launched nuclear missiles. Instead of launching a counterattack, Petrov felt something was wrong and refused; it turned out to be a computer malfunction. AI scientists, says Hendler, should learn from Petrov.

“The real danger is people over-trusting these ‘unbiased’ AI systems,” he says. “What I’m afraid of is most people don’t understand these issues ... and just will trust the system the way they trust other computer systems. If they don’t know these systems have these limitations, they won’t be looking for the alternatives that humans are good at.”

#### Further Reading

Madras, D., Creager, E., Pitassi, T., and Zemel, R. **Learning Adversarially Fair and Transferable Representations**, 17 Feb. 2018, Cornell University Library, <https://arxiv.org/abs/1802.06309>

Buolamwini, J. and Gebru, T. **Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification**, *Proceedings of Machine Learning Research*, 2018, Conference on Fairness, Accountability and Transparency. <http://proceedings.mlr.press/v81/buolamwini18a/buolamwini18a.pdf>

Dovey Fishman, T., Eggers, W.D., and Kishnani, P. **AI-augmented human services: Using cognitive technologies to transform program delivery**, Deloitte Insights, 2017, <https://www2.deloitte.com/insights/us/en/industry/public-sector/artificial-intelligence-technologies-human-services-programs.html>

Zhao, J., Wang, T., Yatskar, M., Ordonez, V., and Chang, K.. **Men Also Like Shopping: Reducing Gender Bias Amplification using Corpus-level Constraints**, University of Virginia. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2979–2989 Copenhagen, Denmark, Sept. 7–11, 2017. [https://pdfs.semanticscholar.org/566f/34fd344607693e490a636cddf3b92f74f976.pdf?\\_ga=2.37177120.1400811332.1523294823-1569884054.1523294823](https://pdfs.semanticscholar.org/566f/34fd344607693e490a636cddf3b92f74f976.pdf?_ga=2.37177120.1400811332.1523294823-1569884054.1523294823)

Tan, S., Caruana, R., Hooker, G., and Lou, Y. **Auditing Black-Box Models Using Transparent Model Distillation With Side Information**, 17 Oct. 2017, Cornell University Library, <https://arxiv.org/abs/1710.06169>

O’Neil, C. **Weapons of Math Destruction**. 2016. Crown Random House.

Hardt, M., Price, E., and Srebro, N. **Equality of Opportunity in Supervised Learning** October 11, 2016 <https://arxiv.org/pdf/1610.02413.pdf>

Esther Shein is a freelance technology and business writer based in the area of Boston, MA, USA.



DOI:10.1145/3267352

Michael A. Cusumano

# Technology Strategy and Management

## The Business of Quantum Computing

*Considering the similarities of quantum computing development to the early years of conventional computing.*

**I**N 1981, NOBEL Laureate Richard Feynman challenged the computing community to build a quantum computer. We have come a long way. In 2015, McKinsey estimated there were 7,000 researchers working on quantum computing, with a combined budget of \$1.5 billion.<sup>20</sup> In 2018, dozens of universities, approximately 30 major companies, and more than a dozen startups had notable R&D efforts.<sup>a</sup> Now seems like a good time to review the business.

### *How do quantum computers work?*

Quantum computers are built around circuits called quantum bits or qubits. One qubit can represent not just 0 or 1 as in traditional digital computers, but 0 or 1 or both simultaneously—a phenomenon called “superposition.” A pair of qubits can represent four states, three qubits eight states, and so on.  $N$  qubits can represent  $2^N$  bits of information, and even 300 qubits can

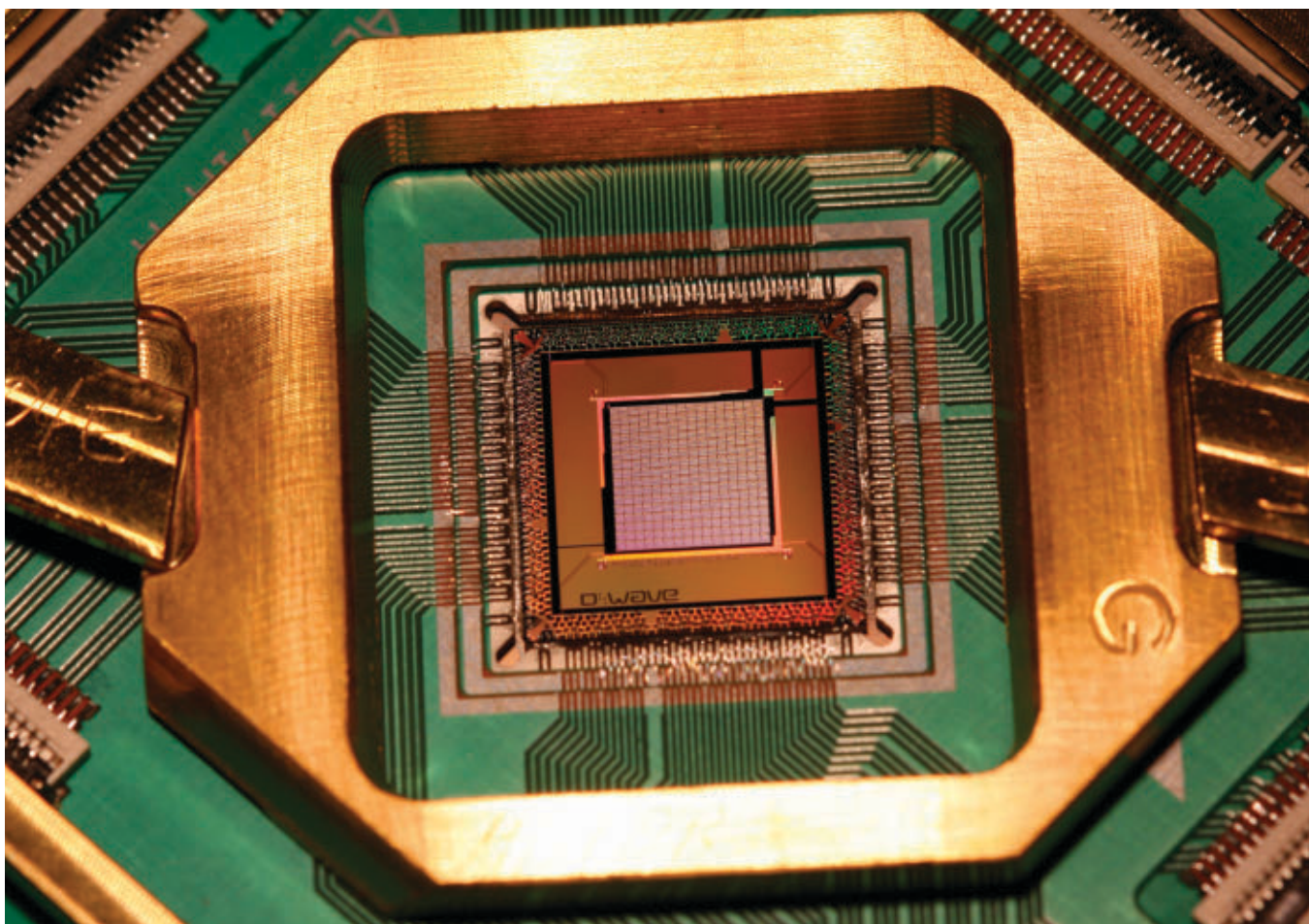
represent information equal to the estimated number of particles in the known universe.<sup>21</sup> To perform calculations, qubits exploit superposition and “entanglement.” This refers to when two quantum systems (such as an electron or a nucleus), once they interact, become connected and retain a specific correlation in their spin or energy states (which represent combinations of 0 and 1), even if physically separate. Entanglement makes it possible for quantum bits to work together and represent multiple combinations of values simultaneously, rather than represent one combination at a time. Once a calculation is finished, you observe the qubits directly as 0 or 1 values to determine the solution, as with a classical computer.

**What are the technical hurdles?** Qubits resemble hardwired logic gates usually made of atomic particles and superconductor materials chilled to near-absolute zero. A one-qubit system is not so difficult to build, but a quantum

computer needs multiple qubits to do calculations, and at least 50 qubits to do anything useful.<sup>14</sup> We might need 4,000 to 8,000 entangled qubits to surpass current encryption technology using very large integers.<sup>3</sup> Programming the devices also requires specialized hardware design skills, not conventional software programming skills.<sup>3</sup>

Entangled qubits are difficult to use and scale because of another phenomenon called “decoherence.” The specific correlations between quantum states can dissipate over time, thus destroying the ability of qubits to explore multiple solutions simultaneously. A useful analogy is to think of qubit outputs like smoke rings blown from a cigar.<sup>14</sup> The rings can represent information but disintegrate (lose their “coherence”) quickly. Since entangled qubits have a small probability of taking on different values due to external interactions, the computations require another process to detect and correct errors.

<sup>a</sup> <https://bit.ly/2OXEA5n>



The D-Wave 2000Q chip, designed to run quantum computing problems, increases from 1,000 qubits to 2,000 qubits, allowing larger problems to be run—increasing the number of qubits yields an exponential increase in the size of the feasible search space.

**How many different ways are there to build quantum computers?** There are several competing technologies. D-Wave was founded in 1999 to accumulate patent rights in exchange for research grants.<sup>17</sup> It has been funded mainly by venture capital, corporate investors such as Goldman Sachs, and more recently, Jeff Bezos and the CIA.<sup>13</sup> The company has focused on “adiabatic quantum computing,” also known as “quantum annealing.” D-Wave used this approach to build a 28-qubit device in 2007 and has been marketing a 2,000-qubit device since 2017. Each D-Wave qubit is a separate lattice contained within a magnetic field of Josephson Junctions (logic circuits made of superconductor materials that exploit quantum tunneling effects) and couplers (which link the circuits and pass information). You program the device by loading mathematical equations into the lattices. The processor then explores all possible solutions simultaneously, rather than one at a

time. The answer that requires the lowest energy represents the optimal solution.<sup>10</sup> However, some critics note that D-Wave qubits do not all seem to work together or exhibit quantum entanglement, and may not operate faster than conventional computers.<sup>4</sup>

Google and IBM, as well as startups such as Quantum Circuits and Righetti Computing, deploy a different logic-gate approach, using entangled electrons or nuclei.<sup>19</sup> Xanadu, a Toronto startup, uses photons.<sup>b</sup> Microsoft’s design relies on quasi-particles called anyons. Arranged into “topological qubits,” these resemble braided knots on a string, with (theoretically) high levels of stability and coherence. Microsoft plans to build a device within five years and make it commercially available via the cloud.<sup>1,16</sup>

**Who leads in the patent race?** Patent-related publications have increased from a handful in the 1990s to

more than 400 per year in 2016–2017. The U.S. leads with approximately 800 total patents, three to four times the numbers from Japan and China. The company with the largest portfolio is D-Wave, followed by IBM (which started research in 1990) and then Microsoft. IBM leads in annual patent filings. At universities, the leaders in patent applications are MIT, Harvard, Zhejiang (China), Yale, and Tsinghua (China).<sup>2</sup>

**What are some applications where quantum computers should excel?**

Experts list mathematical problems that require massive parallel computations such as in optimization and simulation, cryptography and secure communications, pattern matching and big-data analysis, and artificial intelligence and machine learning.

D-Wave computers seem to generate “good enough” solutions to complex combinatorial optimization problems with many potential solutions. For example, in 2012, Harvard researchers

b <https://bit.ly/2B04tP1>



used a D-Wave computer to do complex simulations of protein molecule unfolding (useful in drug discovery).<sup>22</sup> Since 2013, NASA and Google, along with several universities, have been using D-Wave computers in their joint Quantum AI Lab.<sup>7</sup> The Lab has explored Web search, speech recognition, planning and scheduling, and operations management.<sup>9</sup> Since 2014, Northrup-Grumman has been using D-Wave to simulate large-scale software systems behavior (useful for error detection).<sup>4</sup> Volkswagen, BMW, and Google are relying on D-Wave to analyze the huge amounts of data needed for self-driving cars. In 2017, Volkswagen used a \$15-million D-Wave computer accessed via the cloud to optimize the airport routes of 10,000 taxis in Beijing. The machine processed GPS data in seconds that would normally take a computer 45 minutes. The programming took six months, however, and some experts doubt the results, which have not been published in a scientific journal.<sup>6,11</sup>

Perhaps the “killer app” will be quantum encryption and secure communications. These applications utilize an algorithm discovered in 1994 by Peter Shor, formerly of Bell Labs and now at MIT. Shor demonstrated how to use a quantum computer to factor very large numbers. Entanglement also makes it possible to have unbreakable cryptographic keys across different locations. Governments (the U.S. and China in particular) as well as companies (AT&T, Alibaba, BT, Fujitsu, HP, Huawei, Mitsubishi, NEC, Raytheon, and Toshiba, among others) have been pursuing these applications.<sup>c</sup> China seems especially advanced.<sup>18</sup>

**Do quantum computers represent a new general-purpose computing “platform?”** No. Quantum computers are special-purpose devices that exploit quantum phenomena for massively parallel computations. They are not suited to everyday computing tasks that require speed, precision, and ease of use at low cost. The competing technologies also seem useful for different applications, and so multiple types of quantum computers may persist, splitting potential application ecosystems. D-Wave computers

## Perhaps the “killer app” will be quantum encryption and secure communications.

tackle optimization and simulation problems. They cannot run Shor’s algorithm, and so may not be not useful for cryptography or quantum communications. IBM, Google, and Microsoft, as well as several startups, are designing more general-purpose devices, but these are still theoretical, experimental, or small scale.

For the business to progress faster, more people need access to bigger quantum computers so they can build better programming tools and test real-world applications. Toward this end, IBM has made small quantum computers available via the cloud and is heading toward bigger devices; users have already run approximately 300,000 experiments.<sup>12,15</sup> Google has made its D-Wave computer available to researchers as a cloud service.<sup>8</sup> Google is also designing bigger machines with a different technology. Microsoft announced in 2017 that it would offer up to 40 qubits via a simulator on the Azure cloud. Microsoft has also created a quantum programming language called Q# and integrated this with Visual Studio.<sup>3,4</sup> However, Microsoft has not yet built physical devices and the programming language may be completely specific to its architecture.<sup>5</sup>

In short, quantum computing still resembles conventional computing circa the late 1940s and early 1950s. We have laboratory devices and some commercial products and services, but mostly from one company. We have incompatible architectures still in the research stage, with different strengths and weaknesses. All the machines require specialized skills to build and program. Companies

still work closely with universities and national laboratories. There is no consensus as to what is the best technology or design. D-Wave led the first generation but its computers are technically limited and scientifically controversial. Although D-Wave should survive as a niche player, IBM and Google seem more likely to dominate the next generation, with Microsoft and maybe a startup or two close on their heels.<sup>12</sup> **C**

### References

1. Bisson, S. Inside Microsoft’s quantum computing world. *InfoWorld* (Oct. 17, 2017).
2. Brachman, S. U.S. leads world in quantum computing patent filings with IBM leading the charge. *IP Watchdog* (Dec. 4, 2017).
3. Bright, P. Microsoft makes play for next wave of computing with quantum computing toolkit. *Ars Technica* (Sept. 25, 2017).
4. Brooks, M. Quantum computers buyers’ guide: Buy one today. *New Scientist* (Oct. 15, 2014).
5. Campbell, F. Microsoft’s quantum computing vaporware. *Forbes.com* (Dec. 18, 2017).
6. Castellanos, S. Companies look to make quantum leap with new technology. *The Wall Street Journal* (May 6, 2017).
7. Choi, C. Google and NASA launch quantum computing AI lab. *MIT Technology Review* (May 16, 2013).
8. Condon, S. Google takes steps to commercialize quantum computing. *ZDNet* (July 17, 2017).
9. D-Wave Systems, Inc. D-Wave 2000Q system to be installed at quantum artificial intelligence lab run by Google, NASA, and Universities Space Research Association. Press Release (Mar. 13, 2017).
10. D-Wave Systems, Inc. Introduction to the D-Wave quantum hardware; <https://bit.ly/2FzstKS>
11. Ewing, J. BMW and Volkswagen try to beat Google and Apple at their own game. *The New York Times* (June 22, 2017).
12. Grossman, L. Quantum leap. *Time* (Feb. 17, 2014).
13. Guedim, Z. 11 Companies set for quantum leap in computing. *EdgyLabs* (Oct. 12, 2017).
14. Hardy, Q. A strange computer promises great speed. *The New York Times* (Mar. 21, 2013).
15. Knight, W. Serious quantum computers are finally here. What are we going to do with them? *MIT Technology Review* (Feb. 21, 2018).
16. Lee, C. How IBM’s new five qubit universal quantum computer works. *Ars Technica* (May 4, 2016).
17. Linn, A. The future is quantum: Microsoft releases free preview of quantum development kit. (Dec. 11, 2017); <https://bit.ly/2C3fxv3>
18. MacCormack, A., Agrawal, A., and Henderson, R. D-Wave systems: Building a quantum computer. Harvard Business School Case #9-604-073 (Apr. 2004), Boston, MA.
19. Matthews, O. How China is using quantum physics to take over the world and stop hackers. *Newsweek* (Oct. 30, 2017).
20. Metz, C. Yale professors race Google and IBM to the first quantum computer. *The New York Times* (Nov. 13, 2017).
21. Palmer, J. Here, there, and everywhere: Quantum technology is beginning to come into its own. *The Economist* (May 20, 2018).
22. Veritasium. How does a quantum computer work? (June 17, 2013); <https://bit.ly/1ApDtkj>
23. Wang, B. Dwave adiabatic quantum computer used by Harvard to solve protein folding problems. *Next Big Future* (Aug. 16, 2012).

**Michael A. Cusumano** ([cusumano@mit.edu](mailto:cusumano@mit.edu)) is a professor at the MIT Sloan School of Management and founding director of the Tokyo Entrepreneurship and Innovation Center at Tokyo University of Science.

The author thanks Ganesh Vaidyanathan for his comments.

Copyright held by author.

c <https://bit.ly/2OXEA5n>

d <https://bit.ly/2B4SMFg>



► Carl Landwehr, Column Editor

## Privacy and Security

# A Pedagogic Cybersecurity Framework

*A proposal for teaching the organizational, legal, and international aspects of cybersecurity.*

**R**EAL<sup>a</sup> CYBERSECURITY TODAY devotes enormous effort to non-code vulnerabilities and responses. The Cybersecurity Workforce Framework<sup>a</sup> of the National Initiative for Cybersecurity Education lists 33 specialty areas for cybersecurity jobs. Ten of the specialty areas primarily involve coding, but more than half primarily involve non-code work (15 areas, in my estimate) or are mixed (eight areas, per my assessment).

This column proposes a Pedagogic Cybersecurity Framework (PCF) for categorizing and teaching the jumble of non-code yet vital cybersecurity topics. From my experience teaching cybersecurity to computer science and other majors at Georgia Tech, the PCF clarifies how the varied pieces in a multidisciplinary cybersecurity course fit together. The framework organizes the subjects that have not been included in traditional cybersecurity courses, but instead address cybersecurity management, policy, law, and international affairs.

The PCF adds layers beyond the traditional seven layers in the Open Systems Interconnection model (“OSI model” or “OSI stack”). Previous writers have acknowledged the possibility of a layer or layers beyond seven, most commonly calling layer 8



the “user layer.”<sup>b</sup> The framework proposed here adds three layers—layer 8 is organizations, layer 9 is governments, and layer 10 is international. This column explains how the new

framework would benefit cybersecurity students, instructors, researchers, and practitioners. Layers 8–10 classify vulnerabilities and mitigations that are frequently studied by non-computer scientists, but are also critical for a holistic understanding of the cybersecurity ecosystem by computing professionals.

<sup>b</sup> Varying previous definitions of higher layers of the OSI Model are available at [https://en.wikipedia.org/wiki/Layer\\_8](https://en.wikipedia.org/wiki/Layer_8).

<sup>a</sup> <https://bit.ly/2McPRB3>

**Table 1. Vulnerabilities at each layer of the expanded OSI stack.**

As discussed in the column, for layers 8–10, “A” refers to vulnerabilities and risk mitigation arising within the organization or nation; “B” refers to vulnerability and risk mitigation in relation with other actors at that level; and “C” refers to other limits created by actors at that level.

Layer	Vulnerability
1. Physical	Cut the wire; stress equipment; wiretap
2. Data link	Add noise or delay (threatens availability)
3. Network	DNS and BGP attacks; false certificates
4. Transport	Man in the middle
5. Session	Session splicing (Firesheep); MS SMB
6. Presentation	Attacks on encryption; ASN-1 parser attack
7. Application	Malware; manual exploitation of vulnerabilities; SQL injection; buffer overflow
8. Organization	A: Insider attacks; poor training or policies B: Sub-contractors with weak cybersecurity; lack of information sharing C: Weak technical or organizational standards
9. Government	A: Laws prohibiting effective cybersecurity (for example, limits on encryption); weak laws for IoT or other security B: Badly drafted cybercrime laws (for example, prohibiting security research) C: Excessive government surveillance
10. International	A: Nation-state cyberattacks B: Lack of workable international agreements to limit cyberattacks C: Supranational legal rules that weaken cybersecurity (for example, some International Telecommunications Union proposals)

**Table 2. The pedagogic cybersecurity framework.**

Layer of the Expanded OSI Stack	A: Risk Mitigation Within an Organization or Nation	B: Relations with Other Actors	C: Other Limits from This Level	Protocol Data Unit
<b>8: Organization</b>	<b>8A: Internal policies or plans of action</b> to reduce risk within an organization (for example, incident response plans).	<b>8B: Vulnerability management in contracts with other entities,</b> like vendors (for example, cyber-insurance).	<b>8C: Standards and limits originating from the private sector</b> (for example, PCI DSS standard, led by the PCI Cyber Security Standards Council).	Contracts
<b>9: Government</b>	<b>9A: Laws</b> that govern what an individual or organization can or must do (for example, HIPAA Security Rule).	<b>9B: Laws</b> that govern how organizations and individuals interact (for example, Computer Fraud and Abuse Act).	<b>9C: Government limits on its own actions</b> (for example, Fourth Amendment, limits on illegal searches).	Laws
<b>10: International</b>	<b>10A: Unilateral actions by one government directed at one or more other nations</b> (for example, U.S. Cyber Command launching a cyberattack on a hostile nation).	<b>10B: Formal and informal relationship management with other nations</b> (for example, the Budapest Convention's provisions about cybercrime and Mutual Legal Assistance).	<b>10C: Limits on nations that come from other nations</b> (for example, the United Nations and international law).	Diplomacy

## The Abstraction Layers of the OSI Model

The PCF builds on the Open Systems Interconnection model (OSI) stack familiar to most computer scientists. It treats the stack primarily as a conceptual framework for organizing how we understand computing systems, particularly in the security domain. The OSI model describes abstraction layers that enable the student or practitioner to focus on where a problem may exist, such as the physical, network, or application layer. While retaining the abstraction layers from the OSI model, the PCF does not emphasize the role of the OSI model as a standardizing model. Instead, it broadens students' understanding by focusing attention on the critical domains that introduce well-documented and well-understood risks from management, government, and international affairs. I provide supplemental materials online that further discuss the relationship of the PCF to the OSI model and expand other points made in this column.<sup>c</sup>

As a conceptual framework for understanding computer systems, the seven traditional layers apply intuitively to cybersecurity risks, as discussed by Glenn Surman in his 2002 article “Understanding Security Using the OSI Model.”<sup>2</sup> Surman concluded: “The most critical thing you should take from this paper is that for every layer there are attacks being created, or attacks awaiting activation as a result of poor defence.” Bob Blakley from Citicorp assisted with these illustrations of vulnerabilities that exist at each of the seven layers, and I have added vulnerabilities existing at layers 8, 9, and 10.

As a way to introduce layers 8 through 10, each horizontal layer highlights important types of cybersecurity vulnerabilities. At layer 8, organizations face a wide range of cyber-risks, and take many actions to mitigate such risks. At layer 9, governments enact and enforce laws—good laws can reduce cybersecurity risks, while bad laws can make them worse. At layer 10, the international realm, no one nation can impose its laws, but treaties or discussions with Russia and China, for instance, may improve cybersecurity. As shown in Table

<sup>c</sup> Supplementary materials on the framework are available at <https://bit.ly/2MJCrZq>

1, the vulnerabilities in these new layers are further organized by institutional form—whether the vulnerability arises within the organization (or nation), between organizations (or nations), or from other institutions at that layer.

In addition to categorizing vulnerabilities, the PCF builds on another aspect of the OSI model, the “protocol data unit,” such as bits for the physical layer, packets for the network layer, and data for the application and other top layers. These protocol data units “describe the rules that control horizontal communications,” within a single layer of the OSI stack.<sup>d</sup>

At layer 8, for organizations, I suggest the controlling rules come from contracts. The much-cited law and economics scholars Jensen and Meckling have defined corporations as a “nexus of contracts.”<sup>1</sup> Contracts are the governance structure for relations between corporations, such as data-use agreements between an organization and its contractors. Less intuitively for non-lawyers, contracts also govern arrangements within a corporation, governing the roles and actions of the board of directors, management, and employees. Contracts are thus the protocol data unit for layer 8, providing the rules within that layer.

At layer 9, the controlling rules for government—the protocol data units—are laws. Governments enact and enforce laws, requiring actions from the organizations within the government’s jurisdiction. The international realm of layer 10 operates where no binding law applies. Actors at layer 10 interact through diplomacy (or lack of diplomacy), such as negotiating a cyber-related treaty, and sometimes through declared or undeclared war.

Put another way, the traditional seven layers concern protocols expressed in machine language; layers 8 to 10 concern protocols (contracts, laws, diplomacy) expressed in natural language. The layers operate in a way familiar from the OSI stack: organizations at layer 8 select the applications at layer 7. Governments at layer 9 set laws to govern organizations. Actions at layer 10 affect the governments at layer 9, and apply when no single government can set the law.

<sup>d</sup> <https://bit.ly/2x40Aoj>

**I have often encountered practitioners (and researchers) who believe “real” cybersecurity involves writing code.**

### The 3x3 Institutional Matrix

Universities have traditionally studied the three non-code layers in different departments. In general, business schools focus on managing companies and other organizations. Law schools are the experts in law. International relations programs study international affairs. These different university departments are organized based on the institutions they primarily study: companies, laws, and transnational institutions.

By contrast, my experience is that computer scientists often group all of these issues into the general term “policy.” Traditionally in computer science, this soft realm of “policy” is the generic term for everything not expressed in machine language. But public policy departments do not intensively cover all aspects of management, law, and international relations, so the computer science use of “policy” creates confusion for the other departments that increasingly teach and research on cybersecurity. The proposed framework matches the typical departmental organization in universities, and provides a visual representation of the key dimensions for what computer scientists have often simply called “policy.”

As an additional way to organize the many non-code cybersecurity-concerns, the PCF employs a 3x3 matrix that refines which institutions are involved in each area of cyber-vulnerability or response. Table 2 portrays the matrix. In Figure 2, each layer (row) is defined by the institutions that make decisions affecting cybersecurity. Layer 8 applies to organizations facing cyberattacks. Layer 9

applies to governments writing and enforcing laws about cybersecurity. Layer 10 applies where there is no government to issue laws. Study of layer 10 thus includes both state and non-state actors that have transborder effects.

In the matrix, each of the three columns refines the sorts of institutions making the decisions. For each layer, column A contains issues arising within the institution—the organization or nation. Each “issue” identifies cyber vulnerabilities or mitigating activities. Column B contains issues defined by relations with other actors at that level. Column C contains issues where other limits arise from actors at the same layer of the stack.

This three-column approach becomes clearer as applied to layer 8, the organizational layer. Column A includes cybersecurity activities within a single organization. A company (or other organization that faces cybersecurity attacks) takes numerous actions to reduce cyber-risk. It develops incident response plans and other internal policies, and trains its employees. One way to conceptualize cell 8A is to think of the responsibilities of a CISO in managing cyber-risk within the organization.

Column B in layer 8 (cell 8B) concerns the organization’s relations with other actors. First, a company creates data-use agreements and other contracts with vendors and other entities. Flawed management of these relations can expose a company to risk, such as if it hires a subcontractor to manage systems or data and the contractor does so badly. Another much-discussed aspect of cybersecurity is information sharing between organizations, such as through an Information Sharing and Analysis Center.

The third column, cell 8C, concerns other limits that originate in the private sector. The PCI DSS standard is a well-known example, governing security at the point of sale. This standard has a powerful effect on the cybersecurity of millions of merchants. The contractual standard originates in the private sector, led by the PCI Security Standards Council. If the standard is designed and implemented well, then cybersecurity improves; if done badly, cyber-risks and costs increase.



Looking at layer 8 as a whole, the simple point is that overall cybersecurity significantly depends on how well an organization handles risk within its organization (8A), its contracts and relations with other actors (8B), and standards and norms that come from the private sector (8C).

Governments, for purposes of the PCF, create laws. Cell 9A contains laws that govern what an individual or organization can do. For instance, using U.S. examples for illustration, the HIPAA Security Rule sets requirements for medical providers. As a different example, consider legislation that would prohibit the use of strong encryption or require a backdoor. I have opposed such legislation, but it illustrates how a government law, applying to each organization, can affect cybersecurity risk.

Cell 9B contains laws that govern how organizations and individuals interact. Some of the HIPAA requirements fit here, such as the business associate requirements of HIPAA that govern contracts with outside parties. An important example in cell 9B is the Computer Fraud and Abuse Act, the anti-hacking law that defines when it is criminal to access computer systems without authorization.

Whereas cells 9A and 9B primarily concern government laws affecting the private sector, cell 9C applies to government limits on government action. The limit on illegal searches in the Fourth Amendment is one example. More broadly, cell 9C concerns the controversial topic of government surveillance. Surveillance sometimes aids security, such as when a criminal is detected, and sometimes hurts security, such as when government actions create backdoors or other vulnerabilities.

The international layer applies to actions taken within one nation that are intended to have cyber effects in other nations. Cell 10A concerns unilateral actions by one government, such as the U.S. The government, for instance, may decide that U.S. Cyber Command should launch a cyberattack on a hostile nation.

Cell 10B involves relations with other nations, which is the main task of diplomacy. There are formal treaties that affect cybersecurity, such

## The PCF provides a parsimonious way to identify and develop a response to a growing number of non-code cybersecurity risks.

as the Budapest Convention's provisions about cybercrime and Mutual Legal Assistance. More generally, cell 10B applies to the range of possible cooperation with other nations on cyberattack or defense.

Finally, cell 10C applies to limits on nations that come from other nations. For instance, some countries have proposed to set cybersecurity rules through the International Telecommunications Union, associated with the United Nations. If such rules are implemented, then supranational laws could govern cyber actions that have transborder effects.

### Applying the Framework

Adding layers 8, 9, and 10 to the OSI stack in the PCF brings important advantages to the study and practice of cybersecurity. I have personally experienced the framework's usefulness in teaching cybersecurity at my own institution: my cybersecurity classes cover every topic mentioned in this column. The PCF provides students with invaluable context for how all the issues fit together, to ensure they understand the "big picture." The framework also clarifies the scope of a cyber-curriculum. Some classes, for instance, focus primarily on how a CISO or company should manage a company's risks (layer 8). Others are mostly about international affairs (layer 10), perhaps with discussion of national cybersecurity laws (cell 9A). The PCF enables program directors and students to concisely describe the coverage of a cybersecurity class or curriculum.

The 3x3 matrix clarifies a research agenda for those seeking to identify and mitigate non-code cyber problems. For example, cell 8B raises legal

and management issues of how to design and manage cybersecurity contracts: How should cybersecurity be treated in outsourcing or insurance contracts? Cell 9A concerns legal and political science issues of how laws get drafted and implemented. Cell 10C calls on international relations expertise to discuss the role of supranational institutions. Few individuals are expert in all of this literature. Researchers can develop an issue list for each cell, along with canonical readings to assign in general examinations.

For cybersecurity practitioners, I have often encountered practitioners (and researchers) who believe "real" cybersecurity involves writing code, perhaps with some vague acknowledgment of the need for "interdisciplinary" study. The sheer volume of issues identified in the 3x3 matrix emphasizes the growing significance of non-code issues—bad decisions in any part of the matrix can negatively affect cybersecurity. As with the existing seven layers of the stack, organizations can identify their vulnerabilities by systematically examining layers 8 to 10. Organizations can then better identify and mobilize expertise for these non-code cyber issues.

In sum, the PCF provides a parsimonious way to identify and develop a response to the growing number of non-code cybersecurity risks. The 3x3 matrix visually categorizes and communicates the range of non-code cybersecurity issues. No longer can "real" cybersecurity refer only to technical measures. Instead, a large and growing amount of cyber-risk arises from problems at layers 8, 9, and 10. Extending the stack to these 10 layers results in an effective mental model for identifying and mitigating the full range of these risks. ■

### References

1. Jensen, M.C. and Meckling, W.H. Theory of the firm: Managerial behavior, agency costs and ownership structure. *Journal of Financial Economics* 3, 4 (Oct. 1976), 305-360.
2. Surman, G. Understanding security using the OSI model. GSEC Practical Version 1.3 (Mar. 29, 2002); <https://bit.ly/2BaJGrV>.

**Peter Swire** (Peter.Swire@scheller.gatech.edu) is the Elizabeth & Tommy Holder Chair of Law and Ethics in the Scheller College of Business and Associate Director for Policy in the Institute for Information Security and Privacy at Georgia Institute of Technology in Atlanta, GA, USA.

Copyright held by author.





## Kode Vicious

# The Obscene Coupling Known as Spaghetti Code

*Teach your junior programmers how to read code.*

**Dear KV,**

Forgive me, for my ACM membership has lapsed, and for my sins I have been saddled with mentoring a spaghetti coder.

I am working on a piece of new software—greenfield for once—but with stiff reliability requirements. My helper, a young, self-proclaimed “devop,” aims to improve as a programmer, and, unfortunately, this person got stuck with me.

No matter how hard I constrain the work I dole out, I just cannot stop this helper from the obscene coupling known as spaghetti code, all masquerading under obsessive, perfect syntax. We cannot even get into the hard reliability aspects of the software, because tangled messes that lint perfectly and break opaquely just keep piling up.

After many approaches, each one narrower in scope than the last, I have come down to doling out work units that are constrained to writing single, well-defined functions in a Python library, but even then I am failing to keep this person from needlessly chaining functions, silently mixing and transparently passing data through multiple layers of interfaces, and, most painfully, burying important error output in ways we all know too well as spaghetti code.

Assuming this apprentice is willing and eager, how can one go about breaking this fundamental coupling



mentality in implementation and open this person’s mind to engage the actual problem at hand—what the software does!

I do not want to botch this and produce the next Darth Vader!

**Mr. Function Defines Form**

**Dear Function,**

Well, at least you didn’t mention goto, the root of much of the spaghetti code of my well-spent youth. Yes, KV was once young, but because of programmers such as your ward, he has never looked young or beautiful.

Once upon a time, spaghetti code was defined by the fact that it jumped all over the place without any rhyme or reason, but, as you say, you have someone, who even when given a constrained contract such as single functions, is still able to make a plate of pasta of it.

Perhaps it is time to introduce the idea of narrative to your Padawan. Code, as I have pointed out countless times, is a form of communication between the people who write and maintain it and is only incidentally executable on a machine, which we call a computer. I cannot seem to say



## Advertise with ACM!

Reach the innovators and thought leaders working at the cutting edge of computing and information technology through ACM's magazines, websites and newsletters.



Request a media kit with specifications and pricing:

**Ilia Rodriguez**  
+1 212-626-0686  
acmm mediasales@acm.org



## The concept of simple narrative can be applied to code.

this often enough, clearly, because I say it a lot. Someday I will lose my voice, and the people I am screaming at will finally think they will get some peace; but if that ever happens, I have a recorded version I can play through a megaphone.

Communication is just a fancy word for storytelling, something that humans have probably been doing since before we acquired language. Unless you are an accomplished surrealist, you tell a story by starting at the beginning, then over the course of time expose the reader to more of the details, finally arriving at the end where, hopefully, the reader experiences a satisfying bit of closure. The goal of the writer (or coder) is to form in the mind of the reader the same image the writer had. That is the process of communication, and it does not matter if it is prose, program, or poetry—at the end of the day, if the recipient of our message has no clue what we meant, then all was for naught.

Of course, as many brilliant writers have proven over time, clear narrative is not entirely necessary, but let's just stick with the clear narrative metaphor for code, rather than claiming we should write an accounting system based on *Naked Lunch*. I mean, I would enjoy it, but would it work? Only the Mugwumps would know.

The concept of simple narrative can be applied to code in the following way. We are trying to write down the steps that are required to do a particular job with a machine in such a way that when other readers come upon the narrative (code)—which is usually thrust upon them with a bug list as long as a baby's arm—they are able to pick up the story wherever they choose. For a short program, something less than 100 lines, the narrative

can all be in one `main()` function. I recommend that you find a few such programs—well written, well commented, and that do one thing and do it well. Then make your Padawan read them and explain them to you.

KV has extolled the virtues of reading good code as a way of learning to write good code, and for young readers, short programs are best. Even though you are working on greenfield code—a rarity in our industry—there must be some scripts or code lying about that you do not hate and that extol the virtues you wish to instill in this apprentice. The most important part of any of these programs is that they do one thing, they do it clearly, and it is obvious to even the most inexperienced programmer what is going on. Find that code, explain its beauty, and then make them extend and maintain it.

Since you both are working on the same code base, you also have ample opportunity for leadership by showing this person how you code. You must do this carefully or the junior programmer will think you are pulling rank, but, with a bit of gentle show and tell, you can get your Padawan to see what you are driving at. This human interaction is often difficult for those of us who prefer to spend our days with seemingly logical machines. Mentorship is the ultimate test of leadership and compassion, and I really hope you do not wind up sliced in half on the deck of a planet-smashing space station.

**KV**

### Related articles on [queue.acm.org](https://queue.acm.org)

#### Human-KV Interaction

*Kode Vicious*

<https://queue.acm.org/detail.cfm?id=957782>

#### Reading, Writing, and Code

*Diomidis Spinellis*

<https://queue.acm.org/detail.cfm?id=957782>

#### A Conversation with Steve Bourne, Eric Allman, and Bryan Cantrill

<https://queue.acm.org/detail.cfm?id=1413258>

**George V. Neville-Neil** ([kv@acm.org](mailto:kv@acm.org)) is the proprietor of Neville-Neil Consulting and co-chair of the *ACM Queue* editorial board. He works on networking and operating systems code for fun and profit, teaches courses on various programming-related subjects, and encourages your comments, quips, and code snips pertaining to his *Communications* column.

Copyright held by author.

## Viewpoint

# Building the Universal Archive of Source Code

*A global collaborative project for the benefit of all.*

**S**OFTWARE IS BECOMING the fabric that binds our personal and social lives, embodying a vast part of the technological knowledge that powers our industry and fuels innovation. Software is a pillar of most scientific research activities in all fields, from mathematics to physics, from chemistry to biology, from finance to social sciences. Software is also an essential mediator for accessing any digital information.

In short, a rapidly increasing part of our collective knowledge is embodied in, or dependent on, software artifacts. Our ability to design, use, understand, adapt, and evolve systems and devices on which our lives have come to depend relies on our ability to understand, adapt, and evolve the source code of the software that controls them.

Software source code is a precious, unique form of knowledge. It can be readily translated into a form executable by a machine, and yet it is human readable: Harold Abelson wrote “Programs must be written for humans to read,”<sup>1</sup> and source code is the preferred form for modification of software artifacts by developers.<sup>3</sup> Quite differently from other forms of knowledge, we have grown accustomed to use version-control systems that trace source code development, and provide precious insight into its evolution. As Len Shustek puts it, “Source code provides a view into the mind of the designer.”<sup>4</sup>

And yet, we have not been taking good care of this precious form of knowledge.



Source code is spread around a variety of platforms and infrastructures that we use to develop and/or distribute it, and software projects often migrate from one to another: there is no universal catalog that tracks it all.

Software can be deleted, corrupted, or misplaced. What’s even more worrying, in recent years we have seen major code forges shut down, endangering hundreds of thousands of publicly available software projects at once.<sup>6</sup>

We clearly need a universal archive of software source code.

The deep penetration of software in all aspects of our world brings along failures and risks whose potential impact is growing. Users now understand the need for an organized

attention to software safety, security, reliability, and traceability. But unlike other scientific fields, we lack large-scale research instruments for enabling massive analysis of all the available software source code.

As computer scientists and professionals, it is our duty, responsibility, and privilege to build a shared infrastructure that answers these needs. Not just for our community, not just for the technical and scientific community, but for society as a whole.

Software Heritage<sup>a</sup> is an initiative launched at Inria—the French Institute for Research in Computer Science and Automation—precisely to take up this

<sup>a</sup> See <https://www.softwareheritage.org>



mission. While a full article detailing our approach is available online,<sup>2</sup> we focus here on the challenges raised by the three main goals: collecting, preserving, and sharing the source code of all the software ever written.

### Collection

There are various kinds of source code. Some is current, actively developed, and technically easy to make available; some other is legacy source code that must be painfully retrieved from offline media. Some is open, and free for all to read and reuse; some is closed behind proprietary doors. Software Heritage's ambition is to collect it all.

For current, open source code, we need an automated process to harvest all software projects, with all the available development history, from the many places where development and distribution take place, like forges and package repositories. Yes, we really mean harvesting everything available, with no a priori filtering. Because the value of an active software project will only be known in the future, and because storing all present and future source code can be done at a reasonable cost.

The technical challenge is to build crawlers for each code-hosting platform, as there is no common protocol available, and to develop adapters for all version-control systems and package formats. It is a significant undertaking, but once a standard platform is available each of these crawlers and adapters can be developed in parallel.

For legacy, open source code, we need a crowdsourcing platform to empower the volunteers that are willing to help recover their preferred software artifacts. Guidelines must be offered to help properly reconstruct from the raw material the interesting history that lies behind it, like in the beautiful work that has been done for the history of Unix.<sup>5</sup>

Closed software contains precious knowledge that is more difficult to recover. For example, the Computer History Museum<sup>b</sup> and Living Computers<sup>c</sup> have shown, in the case of the mythical Alto system,<sup>d</sup> that once the busi-

b See <http://www.computerhistory.org/>

c See <http://www.livingcomputers.org/>

d See <http://xeroxalto.computerhistory.org> and <http://www.livingcomputers.org/Discover/News/ContrAlto-A-Xerox-Alto-Emulator.aspx>

## We are at a unique turning point in the history of computer science and technology.

ness need to keep software closed fades away, a focused search (that requires a costly and dedicated effort) can succeed in recovering and liberating its source code, growing our software commons.

Finally, by providing a means to safely keep closed source software under embargo, much like what happens already with software escrow, we may succeed in collecting current and future closed source, and be ready to liberate it when time comes, dispensing altogether with costly technical recovery efforts.

### Preservation

In the extensive literature on digital preservation, it is now well established that long-term preservation requires full access to the source code of the tools used for the task. Software Heritage uses and develops exclusively free and open source software tools for building its archive.

Also, replication and diversification are best practices to mitigate the threats—from technical failures to legal and economic decisions—that endanger any long-term preservation initiative. Hence, we want to foster a geographically distributed network of mirrors, implemented using a variety of storage technologies, in different administrative domains, controlled by a plurality of institutions, and located in different jurisdictions.

Finally, preserving software source code also requires preserving the development history of source code, which carries precious insights into the structure of programs and also tracks inter-project relationships. Software Heritage's unique approach is to store all available source code and its revisions into a single Merkle DAG (Directed Acyclic Graph), shared among all software projects. This data structure facilitates distribution

and enables full deduplication (massively reducing storage costs), integrity checking, and tracking of reuse across all software projects at the file level. But it also poses novel challenges when it comes to efficiently indexing and querying its contents.

### Sharing

The raw material that Software Heritage collects must be properly organized to ease its fruition. On top of the information captured by version-control systems, we need metadata describing the software and means to classify the millions of harvested projects, written in one of the thousands of known programming languages.<sup>e</sup> We need to extract and reconcile existing information from many different sources, encoded in one of the many different software ontologies, and complete it using either automatic tools or crowdsourcing.

We must also support the many use cases that it enables. Programmers may want to search for specific project versions or code snippets to reuse, and then browse them online or download history-full source code bundles. Companies may want to access an API to build applications that use the archive. Researchers may want to access the whole corpus to perform big data operations or train machine learning models.

We must carefully assess which functionalities are generic enough to be incorporated in the archive, and which are so specific that they are best implemented externally by third parties. And there are of course legal and ethical issues to be dealt with when redistributing parts—or all—of the contents of the archive.

### Current Status

Software Heritage is an active project that has already assembled the largest existing collection of software source code. At the time of writing the Software Heritage Archive contains more than four billion unique source code files and one billion individual commits, gathered from more than 80 million publicly available source code repositories (including a full and up-to-date mirror of GitHub) and packages (including a full and up-to-date mirror of Debian). Three copies are currently maintained,

e See <http://hopli.info/>



including one on a public cloud.

As a graph, the Merkle DAG underpinning the archive consists of 10 billion nodes and 100 billion edges; in terms of resources, the compressed and fully deduplicated archive requires some 200TB of storage space. These figures grow constantly, as the archive is kept up to date by periodically crawling major code hosting sites and software distributions, adding new software artifacts, but never removing anything. The contents of the archive can already be browsed online, or navigated via a REST API.<sup>f</sup>

### Next Steps

We are at a unique turning point in the history of computer science and technology. Looking backward, we see many important pieces of historical software that are lost, misplaced, or behind barriers. On the other hand, many of our founding fathers are still here. They have the knowledge and the will to share what is necessary to rebuild the full history of our discipline—a unique opportunity that no other field of science or technology has ever offered.

Looking to the future, we see software development skyrocketing. It is urgent to build the missing infrastructure and put in place the good practices necessary to ensure our entire software commons will be properly collected and preserved. Every year that goes by without acting significantly increases the backlog.

By launching Software Heritage, Inria has done the initial effort, creating the archive infrastructure, establishing an agreement with UNESCO, and assembling an initial group of supporters<sup>g</sup> and committed sponsors, including Microsoft, Intel, Société Générale, Huawei, Google, GitHub, Qwant, Nokia Bell Labs, DANS, Fossil, UQAM, and the University of Bologna. Now we need to move forward, and grow Software Heritage into an international common infrastructure.

Four ingredients are key to the success of our mission: raising awareness of the importance of source code as a first-class citizen in our cultural heritage; gathering the resources needed to create the infrastructure; leveraging

the expertise from many fields of our discipline; and building on a community that shares the vision.

As an open initiative, Software Heritage strives to act as a host and a catalyzer for this community, and we are now calling for contributors to join forces and tackle the issues highlighted in this Viewpoint, and the many others that will arise along the way. A few of these issues include:

► For the collection phase, we need help recovering important software from the past and building adaptors for the many hosting platforms and source code distribution formats.

► For the preservation phase, we need resources to host mirrors, as well as contributors willing to try different technologies for storing and mirroring the archive.

► For the sharing phase, help is needed to organize the contents, to build efficient indexing and querying mechanisms, and to develop applications for specific domains.

We—technologists, engineers, scientists, and IT professionals—have a noble mission and a grand challenge: let's work together to deliver on it. **C**

### References

1. Abelson, H., Sussman, J., and Sussman, J. *The Structure and Interpretation of Computer Programs*. Preface by A.J. Perlis, MIT Press, 1985.
2. Di Cosmo, R. and Zacchiroli, S. *Software Heritage: Why and How to Preserve Software Source Code*. iPRES 2017.
3. Free Software Foundation, Inc. The GNU General Public License, Version 3, §1, 2007.
4. Shustek, L.J. What should we collect to preserve the history of software. *IEEE Annals of the History of Computing*, 2006.
5. Spinellis, D. A repository of Unix history and evolution. *Empirical Software Engineering*, 2017.
6. Squire, M. *The Lives and Deaths of Open Source Code Forges*. OpenSym, 2017.

**Jean-François Abramatic** (Jean-Francois.Abramatic@inria.fr) is research director emeritus at Inria, the French Institute for Research in Computer Science and Automation.

**Roberto Di Cosmo** (roberto@dicosmo.org) is director of Software Heritage at Inria, and professor of computer science at IRIF, University Paris Diderot.

**Stefano Zacchiroli** (zack@upsilon.cc) is associate professor of computer science at IRIF, University Paris Diderot, and CTO of Software Heritage at Inria.

Copyright held by authors.



Watch the authors discuss their work in this exclusive *Communications* video.  
<https://cacm.acm.org/videos/building-the-universal-archive-of-source-code>

# Calendar of Events

## October 14–17

**UIST '18: The 31<sup>th</sup> Annual ACM Symposium on User Interface Software and Technology**, Berlin, Germany,  
 Co-Sponsored: ACM/SIG,  
 Contact: Patrick Baudisch,  
 Email: patrickbaudisch@gmx.net

## October 15–19

**CCS '18: 2018 ACM SIGSAC Conference on Computer and Communications Security**, Toronto, ON, Canada  
 Sponsored: ACM/SIG,  
 Contact: David J.F. Lie,  
 Email: lie@eceg.toronto.edu

## October 16–20

**ICMI '18: International Conference on Multimodal Interaction**, Boulder, CO, USA  
 Sponsored: ACM/SIG,  
 Contact: Sidney D'Mello,  
 Email: sidney.dmello@gmail.com

## October 22–26

**CIKM 2018: The 27<sup>th</sup> ACM International Conference on Information and Knowledge Management**, Torino, Italy,  
 Co-Sponsored: ACM/SIG,  
 Contact: Alfredo Cuzzocrea,  
 Email: cuzzocrea@si.dimes.unical.it

## October 22–26

**MM '18: ACM Multimedia Conference**, Seoul, Republic of Korea,  
 Sponsored: ACM/SIG,  
 Contact: Kyoung Mu Lee,  
 Email: kyoungmu@snu.ac.kr

## October 28–31

**CHI PLAY '18: The Annual Symposium on Computer-Human Interaction in Play**, Melbourne, VIC, Australia  
 Sponsored: ACM/SIG,  
 Contact: Florian Mueller,  
 Email: floyd@floydmueller.com

## October 28–November 2

**MSWIM '18: 21<sup>th</sup> ACM Int'l Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems**, Montreal, QC, Canada  
 Sponsored: ACM/SIG,  
 Contact: Azzedine Boukerche,  
 Email: boukerch@site.uottawa.ca

<sup>f</sup> See <https://archive.softwareheritage.org/>

<sup>g</sup> See <https://www.softwareheritage.org/support/testimonials/>

## Viewpoint

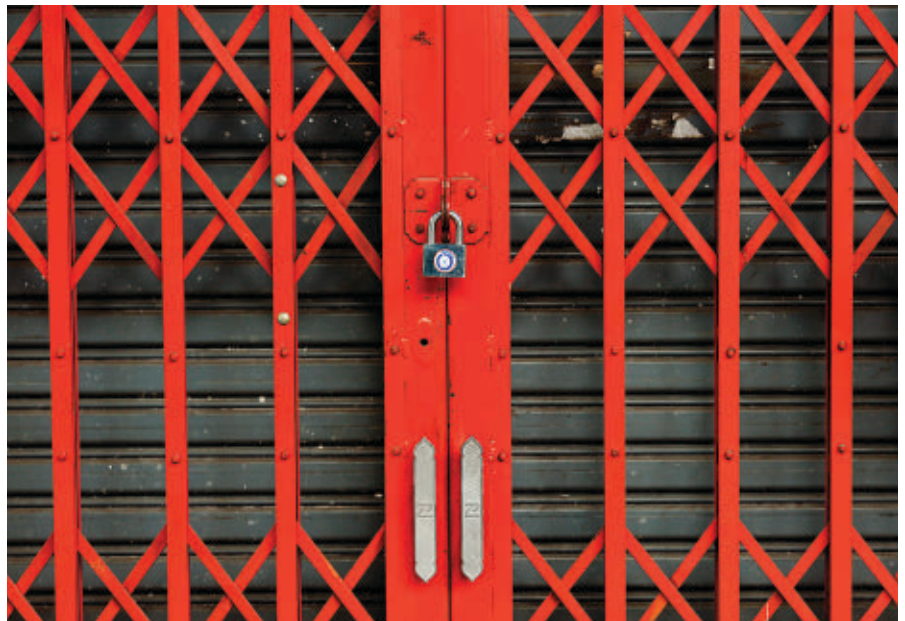
# Are CS Conferences (Too) Closed Communities?

*Assessing whether newcomers have a more difficult time achieving paper acceptance at established conferences.*

**P**UBLICATION IN TOP conferences is a key factor, albeit controversial,<sup>3,4</sup> in the dissemination of ideas and career promotion in many areas of computer science. Therefore, it is a major goal for every CS researcher. However, many researchers believe publishing in a top conference is something reserved for the established members of the conference community. For newcomers, this is a tough nut to crack. Indeed, when talking with fellow researchers the assumed unspoken truth is always the same: If you are not one of “them,” you have no chance to get “in” on your own.

If this were true, it would imply that senior researchers wishing to change fields during their research career may have a difficult time doing so. And the impact would be even more dramatic for junior researchers: they could only access top venues by going together with their supervisor, limiting their options to make a name for themselves—exactly the opposite of what evaluation committees typically require from candidates. Indeed, candidates are supposed to show their ability to propose and develop valid research lines independently of their supervisor, even better if it is in a slightly different research field and hence in a different community.

But is it true that conferences are closed communities? Or is it just a myth spread by those that tried and failed? And if so, how do we change this situation (and do we really need to



change it)? Our goal in this Viewpoint is to shed some light on these issues.

### Looking at the Data

To assess whether it is actually true that newcomers have a difficult time getting their papers accepted, we have evaluated the number of newcomer papers (research papers where all authors are new to the conference, that is, none of the authors has ever published a paper of any kind in that same conference) in 65 conferences. The list of selected conferences corresponds to the list of international CS conferences in the CORE ranking,<sup>a</sup> 2015 edition,

<sup>a</sup> <https://bit.ly/2MnAncz>

*Computer Software* category, for which we were able to find available data in the DBLP dataset, the well-known online reference for computer science bibliographic information. The choice of CORE as ranking system is based on its widespread use.

We have analyzed the conferences using a seven-year window (that is, an author is considered new to a conference if he or she has not published in that conference in the last seven years). We only count full papers in the main research track (since getting short papers, posters, demos, and so forth is typically easier but it barely counts toward promotion).

Results show that newcomers' papers are indeed scarce. Most confer-

ences (88%) show a percentage of newcomer papers under 40%. This value is significantly lower in top conferences, with a median value of 14%. As specific examples, well-regarded conferences show the following values: ICSE (5%), OOPSLA (13%), ICFP (11%), RE (6%). We may be tempted to quickly dismiss these numbers by attributing the low percentage of newcomers papers to a lack of newcomer submissions. While it is true that CS communities are shrinking (at least based on ACM tables for SIG memberships), which could imply that the “newcomers pool” is smaller, our analysis suggests that newcomer paper submissions represent at least one-third of the total number of submissions.<sup>b</sup>

Additionally, for each conference, we have also calculated the number of semi-newcomer papers. A semi-newcomer is a researcher that has never published in the main track but that has published before in other tracks (for example, a demo or a poster). Data indicates publishing a paper as a semi-newcomer is also difficult but slightly easier than doing so as a complete newcomer. If you want to be part of a given community, it seems to pay off to first participate in that community via lesser competitive tracks or collocated satellite events. And the good news is that, unsurprisingly, newcomers have reasonable chances of success to get papers accepted in those satellite events. Our data indicates the percentage of newcomer papers in satellite events is over 30% in most conferences and it frequently goes up to 50% and over. Clearly, satellite events play a positive role in the growth of the community. The full data is available, including all conferences values and the corresponding boxplot distributions based on the conference rankings.<sup>c</sup>

### Opening Up Conferences

We believe the data confirms CS conferences<sup>d</sup> behave as closed communi-

<sup>b</sup> This calculation requires access to the set of papers submitted and rejected. Since this data is not publicly available, this analysis was only done on the four conferences for which one of the authors acted as PC-Chair.

<sup>c</sup> <https://bit.ly/2nCoWzU>

<sup>d</sup> At least in the subarea we have evaluated (computer software category) but we believe results can be generalized to other areas.

## Satellite events play a positive role in the community.

ties. Most likely, some readers believe this is exactly how things should be and that newcomers must first learn the community’s particular “culture” (in the widest sense of the word, including its topics of interest, preferred research methods, social behavior, vocabulary, and even writing style) either by simply attending the conference or warming-up publishing in satellite events, before being able to get their papers accepted in the main research track.

We dare to disagree and argue that the situation is getting to a point in which is worth discussing how to change course. The overall presence of newcomers decreases over time.<sup>2</sup> Besides, increasing travel and economical restrictions make it difficult to follow the (so far) “easier” path to enter the community, for example, many outsider researchers will not get funded to attend a satellite event, preventing them from learning the ropes of that particular community.

While closed communities have indeed some positive aspects (for example, a particular focus, a heritage to build upon, sense of security, and so forth) we believe they are now becoming too closed. In our opinion, a healthier number for conferences would be having at least 25% of newcomer papers in each edition. This would ensure a continuous influx of fresh ideas and new members in the community among other benefits of open communities such as better diversity and inclusiveness. While junior researchers co-authoring a paper with their supervisor for the first time (in fact, the most common path to enter a top conference) could be considered new members as well, we argue that conferences must also make the effort to open up to complete outsiders (including junior researchers trying to start independent

research lines in a new field, senior researchers moving to a new research interest, industrial researchers trying to disseminate their results ...) able to bring a completely fresh perspective to the community.

The main challenge in opening up conferences comes from the fact that we do not really know the reasons why these numbers are so low. Do some potential newcomers refrain from submitting in the first place? Do they get rejected more often than established authors? If the latter, are they being fairly rejected because their papers do not follow the right structure, process, or evaluation standards? Or is there a positive (unconscious) bias toward known community members during the review phase?

Narrowing down a root cause—or causes—requires much more conference data to be publicly disclosed for analysis. We hope this is a direction we will follow as a community. In the meantime, we would like to suggest a few ideas we think are worth pursuing and that, most likely, should be combined in order to tackle this multifaceted challenge:

- **Open the review process.** More and more conferences are adopting a double-blind review model to avoid bias. Its usefulness to avoid author identification seems to be confirmed<sup>6</sup> but it is probably still fairly easy to spot whether the authors are at least members of the community so bias is not completely out of the question. We could go even further and aim for triple-blind reviews or, alternatively, open reviews (where reviewers sign the reviews and/or reviews are later released publicly).

- **Identify and promote research topics with a lower entry barrier for newcomers** either because they are new topics, and therefore not many people in the community work on them, or because they require less advanced skills/infrastructure.

- **Increasing acceptance rates to have more slots available.** This has been proposed as a solution to the randomness of the peer-review system.<sup>8</sup> We could even decide to reserve a few slots for newcomer papers. Obviously, this goes against the traditional conference publication model and could trigger cascade effects on the role of





Association for  
Computing Machinery

## ACM Conference Proceedings Now Available via Print-on-Demand!

*Did you know that you can now order many popular ACM conference proceedings via print-on-demand?*

Institutions, libraries and individuals can choose from more than 100 titles on a continually updated list through Amazon, Barnes & Noble, Baker & Taylor, Ingram and NACSCORP: CHI, KDD, Multimedia, SIGIR, SIGCOMM, SIGCSE, SIGMOD/PODS, and many more.

For available titles and ordering info, visit: [librarians.acm.org/pod](http://librarians.acm.org/pod)



conferences but there is already a part of the community that challenges the idea that very low acceptance rates are indeed good for us. ICSE'17 conference went to the extreme of limiting the number of papers to be submitted by a single author (restriction dropped in 2018 since the community felt it strongly discouraged collaboration). Given that newcomers typically submit far fewer papers, this could help prevent established researchers filling so many slots. An interesting experience nevertheless worth being reevaluated in the future (even if with different “parameters”).


► **Adopt more journal-like review systems.** Introducing revision cycles in a conference could help newcomers to fix obvious but easy-to-correct mistakes that would otherwise force a paper rejection. Even better, a rolling deadline, allowing submissions all year-round (VLDB-style) would avoid paper acceptance to be decided on the basis of the paper itself and not related to the others in order to avoid over the limit acceptance rates.

► **Start mentoring programs where young researchers can pre-submit their work** and get some advice (typically from former PC members) before the actual submission. While mentoring may have a limited success in getting the newcomers’ papers in immediately, it could have a positive long-lasting effect in speeding up the newcomer learning.

► **Draw ideas from other domains where they may face similar problems.** For instance, in the open source community, many projects struggle to attract new contributors and have come up with proposals to attract more people.<sup>7</sup> Examples (adapted to our field) would be to have a dedicated portal for newcomers clearly explaining how papers in the conference are evaluated, showing examples of good papers (in terms of style and structure), listing typical mistakes first submitters do based on the experience of PC members, and so forth. And, importantly, encouraging them to keep trying if they are not initially successful—they may not be aware senior researchers also get many papers rejected.

Despite the number of works analyzing co-authorship graphs, newcomers metrics have been mostly ignored

in previous research works. M. Biryukov et al.<sup>1</sup> study individual newcomer authors, B. Vasilescu et al.<sup>9</sup> and J.L. Cánovas et al.<sup>2</sup> calculate just a coarse-grained newcomers value as part of a larger set of general metrics. We hope to trigger additional research and, especially, general discussions around the trade-offs of closing/opening up more of our research communities<sup>5</sup> with this Viewpoint.

We are aware this is a challenging process due to the leadership role many conferences play in our research system. And we acknowledge opening up a conference is, in fact, an act of generosity. Unless we avoid the zero-sum game of the current publication model (with a somehow fixed number of slots to keep acceptance rates low) any explicit action to increase newcomer participation implies decreasing our own chances to get published. Still, we believe the newcomers’ problem cannot be swept under the carpet any longer if we want to ensure we keep a vibrant and growing community in our research area. 

### References

1. Biryukov, M. and Dong, C. Analysis of computer science communities based on DBLP. *Lecture Notes in Computer Science* (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), 6273 LNCS (2010), 228–235.
2. Cánovas Izquierdo, J.L., Cosentino, V., and Cabot, J. Analysis of co-authorship graphs of CORE-ranked software conferences. *Scientometrics* 109, 3 (Dec. 2016), 1665–1693.
3. Franceschet, M. The role of conference publications in CS. *Commun. ACM* 53, 12 (Dec. 2010), 129.
4. Freyne, J. et al. Relative status of journal and conference publications in computer science. *Commun. ACM* 53, 11 (Nov. 2010), 124.
5. Gebert, D. and Boerner, S. The open and the closed corporation as conflicting forms of organization. *J. Appl. Behav. Sci.* 35, 3 (Sept. 1999), 341–359.
6. Le Goues, C. et al. Effectiveness of anonymization in double-blind review. *Commun. ACM* 61, 6 (June 2018), 30–33.
7. Steinmacher, I. et al. A systematic literature review on the barriers faced by newcomers to open source software projects. *Inf. Softw. Technol.* 59 (2015), 67–85.
8. Vardi, M.Y. Divination by program committee. *Commun. ACM* 60, 9 (Aug. 2017), 7.
9. Vasilescu, B. et al. How healthy are software engineering conferences? *Sci. Comput. Program.* 89, PART C (2014), 251–272.

**Jordi Cabot** ([jordi.cabot@icrea.cat](mailto:jordi.cabot@icrea.cat)) is an ICREA Research Professor at the Universitat Oberta de Catalunya (UOC), an Internet-centered open university based in Barcelona, Spain.

**Javier Luis Cánovas Izquierdo** ([jcanovasi@uoc.edu](mailto:jcanovasi@uoc.edu)) is a Postdoctoral Research Fellow at the Universitat Oberta de Catalunya.

**Valerio Cosentino** ([vcosentino@uoc.edu](mailto:vcosentino@uoc.edu)) was a Postdoctoral Research Fellow at the Universitat Oberta de Catalunya. Since September 2017, he is a software developer at Bitergia, an open source company devoted to offer software development analytics, part of the CHAOSS project of the Linux Foundation.

Copyright held by authors





# AWARD NOMINATIONS SOLICITED

**As part of its mission, ACM brings broad recognition to outstanding technical and professional achievements in computing and information technology.**

ACM welcomes nominations for those who deserve recognition for their accomplishments. Please refer to the ACM Awards website at <https://awards.acm.org> for guidelines on how to nominate, lists of the members of the 2018 Award Committees, and listings of past award recipients and their citations.

Nominations are due **January 15, 2019** with the exceptions of the Doctoral Dissertation Award (due **October 31, 2018**) and the ACM – IEEE CS George Michael Memorial HPC Fellowship (due **May 1, 2019**).

**A.M. Turing Award:** ACM's most prestigious award recognizes contributions of a technical nature which are of lasting and major technical importance to the computing community. The award is accompanied by a prize of \$1,000,000 with financial support provided by Google.

**ACM Prize in Computing (previously known as the ACM-Infosys Foundation Award in the Computing Sciences):** recognizes an early- to mid-career fundamental, innovative contribution in computing that, through its depth, impact and broad implications, exemplifies the greatest achievements in the discipline. The award carries a prize of \$250,000. Financial support is provided by Infosys Ltd.

**Distinguished Service Award:** recognizes outstanding service contributions to the computing community as a whole.

**Doctoral Dissertation Award:** presented annually to the author(s) of the best doctoral dissertation(s) in computer science and engineering, and is accompanied by a prize of \$20,000. The Honorable Mention Award is accompanied by a prize totaling \$10,000. Winning dissertations are published in the ACM Digital Library and the ACM Books Series.

**ACM – IEEE CS George Michael Memorial HPC Fellowships:** honors exceptional PhD students throughout the world whose research focus is on high-performance computing applications, networking, storage, or large-scale data analysis using the most powerful computers that are currently available. The Fellowships includes a \$5,000 honorarium.

**Grace Murray Hopper Award:** presented to the outstanding young computer professional of the year, selected on the basis of a single recent major technical or service contribution. The candidate must have been 35 years of age or less at the time the qualifying contribution was made. A prize of \$35,000 accompanies the award. Financial support is provided by Microsoft.

**Paris Kanellakis Theory and Practice Award:** honors specific theoretical accomplishments that have had a significant and demonstrable effect on the practice of computing. This award is accompanied by a prize of \$10,000 and is endowed by contributions from the Kanellakis family, and financial support by ACM's SIGACT, SIGDA, SIGMOD, SIGPLAN, and the ACM SIG Project Fund, and individual contributions.

**Karl V. Karlstrom Outstanding Educator Award:** presented to an outstanding educator who is appointed to a recognized educational baccalaureate institution, recognized for advancing new teaching methodologies, effecting new curriculum development or expansion in computer science and engineering, or making a significant contribution to ACM's educational mission. The Karlstrom Award is accompanied by a prize of \$10,000. Financial support is provided by Pearson Education.

**Eugene L. Lawler Award for Humanitarian Contributions within Computer Science and Informatics:** recognizes an individual or a group who have made a significant contribution through the use of computing technology; the award is intentionally defined broadly. This biennial, endowed award is accompanied by a prize of \$5,000, and alternates with the ACM Policy Award.

**ACM – AAAI Allen Newell Award:** presented to individuals selected for career contributions that have breadth within computer science, or that bridge computer science and other disciplines. The \$10,000 prize is provided by ACM and AAAI, and by individual contributions.

**Outstanding Contribution to ACM Award:** recognizes outstanding service contributions to the Association. Candidates are selected based on the value and degree of service overall.

**ACM Policy Award:** recognizes an individual or small group that had a significant positive impact on the formation or execution of public policy affecting computing or the computing community. The biennial award is accompanied by a \$10,000 prize. The next award will be the 2019 award.

**Software System Award:** presented to an institution or individuals recognized for developing a software system that has had a lasting influence, reflected in contributions to concepts, in commercial acceptance, or both. A prize of \$35,000 accompanies the award with financial support provided by IBM.

**ACM Athena Lecturer Award:** celebrates women researchers who have made fundamental contributions to computer science. The award includes a \$25,000 honorarium.

For SIG-specific Awards, please visit <https://awards.acm.org/sig-awards>.

**Vinton G. Cerf**, ACM Awards Committee Co-Chair

**Insup Lee**, SIG Governing Board Awards Committee Liaison

**John R. White**, ACM Awards Committee Co-Chair

**Rosemary McGuinness**, ACM Awards Committee Liaison

Article development led by [acmqueue](https://queue.acm.org)  
queue.acm.org

**In machine learning, the concept of interpretability is both important and slippery.**

BY ZACHARY C. LIPTON

# The Mythos of Model Interpretability

SUPERVISED MACHINE-LEARNING models boast remarkable predictive capabilities. But can you trust your model? Will it work in deployment? What else can it tell you about the world? Models should be not only good, but also interpretable, yet the task of interpretation appears underspecified. The academic literature has provided diverse and sometimes non-overlapping motivations for interpretability and has offered myriad techniques for rendering interpretable models. Despite this ambiguity, many authors proclaim their models to be interpretable axiomatically, absent further argument. Problematically, it is not clear what common properties unite these techniques.

This article seeks to refine the discourse on interpretability. First it examines the objectives of previous papers addressing interpretability, finding them to be diverse and occasionally discordant. Then, it explores model properties and techniques thought to confer interpretability, identifying

transparency to humans and post hoc explanations as competing concepts. Throughout, the feasibility and desirability of different notions of interpretability are discussed. The article questions the oft-made assertions that linear models are interpretable and that deep neural networks are not.

Until recently, humans had a monopoly on agency in society. If you applied for a job, loan, or bail, a human decided your fate. If you went to the hospital, a human would attempt to categorize your malady and recommend treatment. For consequential decisions such as these, you might demand an explanation from the decision-making agent.

If your loan application is denied, for example, you might want to understand the agent's reasoning in a bid to strengthen your next application. If the decision was based on a flawed premise, you might contest this premise in the hope of overturning the decision. In the hospital, a doctor's explanation might educate you about your condition.

In societal contexts, the *reasons* for a decision often matter. For example, intentionally causing death (murder) vs. unintentionally (manslaughter) are distinct crimes. Similarly, a hiring decision being based (directly or indirectly) on a protected characteristic such as race has a bearing on its legality. However, today's predictive models are not capable of reasoning at all.

Over the past 20 years, rapid progress in machine learning (ML) has led to the deployment of automatic decision processes. Most ML-based decision making in practical use works in the following way: the ML algorithm is trained to take some input and predict the corresponding output. For example, given a set of attributes characterizing a financial transaction, an ML algorithm can predict the long-term return on investment. Given images from a CT scan, the algorithm can assign a probability that the scan depicts a cancerous tumor. The ML algorithm takes in a large corpus of (in-



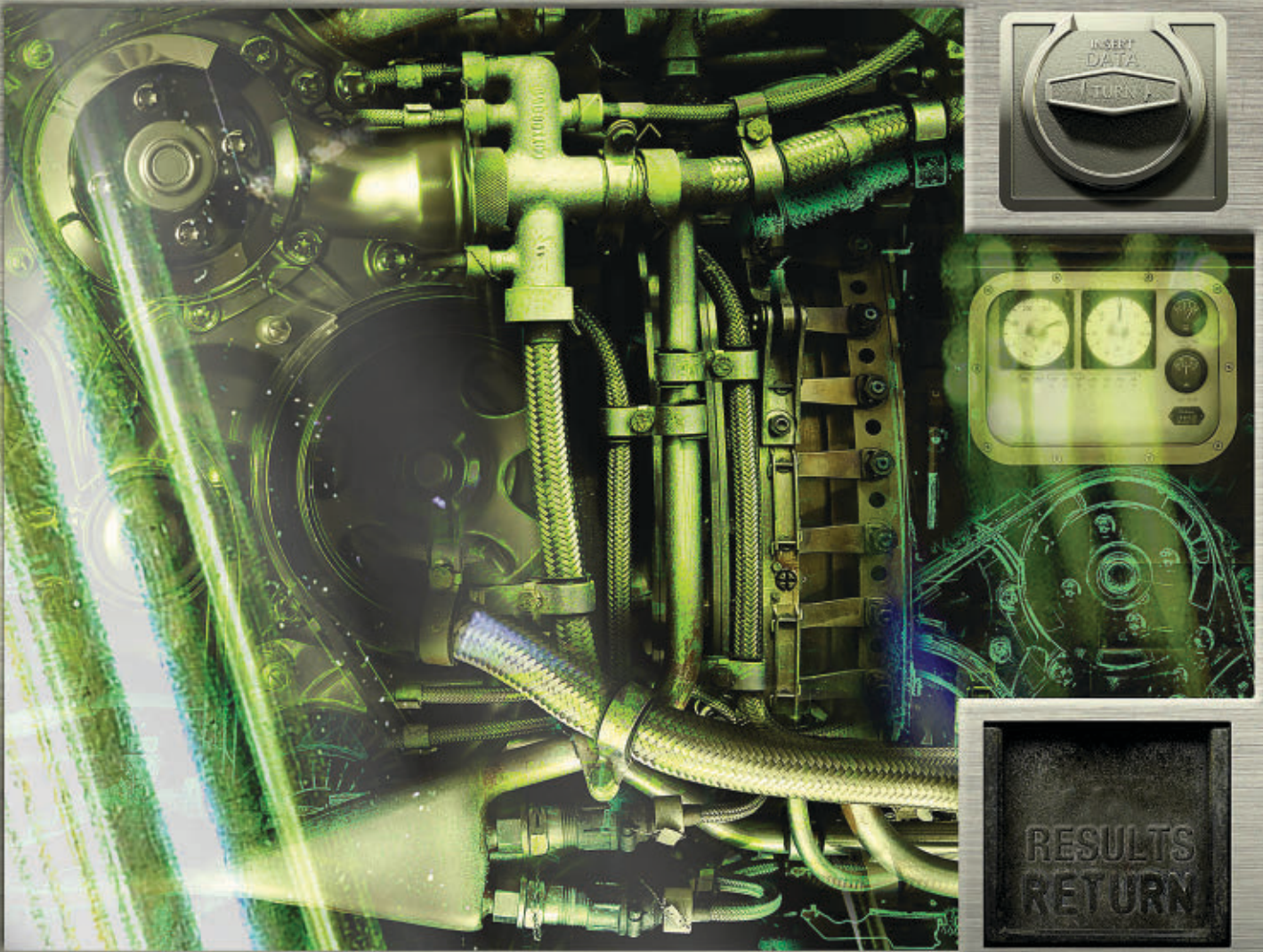


IMAGE BY ALICIA KUBISTA/ANDRIJ BORYS ASSOCIATES

put, output) pairs, and outputs a *model* that can predict the output corresponding to a previously unseen input. Formally, researchers call this problem setting *supervised learning*. Then, to automate decisions fully, one feeds the model's output into some decision rule. For example, spam filters programmatically discard email messages predicted to be spam with a level of confidence exceeding some threshold.

Thus, ML-based systems do not know why a given input should receive some label, only that certain inputs are correlated with that label. For example, shown a dataset in which the only orange objects are basketballs, an image classifier might learn to classify all orange objects as basketballs. This model would achieve high accuracy even on held out images, despite failing to grasp the difference that actually makes a difference.

As ML penetrates critical areas such as medicine, the criminal justice system, and financial markets, the inability of humans to understand these models seems problematic. Some suggest *model interpretability* as a remedy, but in the academic literature, few authors articulate precisely what interpretability means or precisely how their proposed solution is useful.

Despite the lack of a definition, a growing body of literature proposes purportedly interpretable algorithms. From this, you might conclude that either: the definition of *interpretability* is universally agreed upon, but no one has bothered to set it in writing; or the term *interpretability* is ill-defined, and, thus, claims regarding interpretability of various models exhibit a quasi-scientific character. An investigation of the literature suggests the latter. Both the objectives and methods put forth in the literature investigating interpretability are

diverse, suggesting that interpretability is not a monolithic concept but several distinct ideas that must be disentangled before any progress can be made.

This article focuses on supervised learning rather than other ML paradigms such as reinforcement learning and interactive learning. This scope derives from the current primacy of supervised learning in real-world applications and an interest in the common claim that linear models are interpretable while deep neural networks are not.<sup>15</sup> To gain conceptual clarity, consider these refining questions: What is interpretability? Why is it important?

Let's address the second question first. Many authors have proposed interpretability as a means to engender trust.<sup>9,24</sup> This leads to a similarly vexing epistemological question: What is trust? Does it refer to faith that a model will perform well? Does trust require a low-level mechanistic understanding




of models? Or perhaps trust is a subjective concept?

Other authors suggest that an interpretable model is desirable because it might help uncover causal structure in observational data.<sup>1</sup> The legal notion of a *right to explanation* offers yet another lens on interpretability. Finally, sometimes the goal of interpretability might simply be to get more useful information from the model.


While the discussed desiderata, or objectives of interpretability, are diverse, they typically speak to situations where standard ML problem formulations, for example, maximizing accuracy on a set of hold-out data for which the training data is perfectly representative, are imperfectly matched to the complex real-life tasks they are meant to solve. Consider medical research with longitudinal data. The real goal may be to discover potentially causal associations that can guide interventions, as with smoking and cancer.<sup>29</sup> The optimization objective for most supervised learning models, however, is simply to minimize error, a feat that might be achieved in a purely correlative fashion.

Another example of such a mismatch is that available training data imperfectly represents the likely deployment environment. Real environments often have changing dynamics. Imagine training a product recommender for an online store, where new products are periodically introduced, and customer preferences can change over time. In more extreme cases, actions from an ML-based system may alter the environment, invalidating future predictions.

After addressing the desiderata of interpretability, this article considers which properties of models might render them interpretable. Some papers equate interpretability with understandability or intelligibility,<sup>16</sup> (that is, you can grasp how the models work). In these papers, understandable models are sometimes called *transparent*, while incomprehensible models are called *black boxes*. But what constitutes transparency? You might look to the algorithm itself: Will it converge? Does it produce a unique solution? Or you might look to its parameters: Do you understand what each represents? Alternatively,



## What is trust? Is it simply confidence that a model will perform well?



you could consider the model's complexity: Is it simple enough to be examined all at once by a human?

Other work has investigated so-called post hoc interpretations. These interpretations might explain predictions without elucidating the mechanisms by which models work. Examples include the verbal explanations produced by people or the saliency maps used to analyze deep neural networks. Thus, human decisions might admit post hoc interpretability despite the black-box nature of human brains, revealing a contradiction between two popular notions of interpretability.

### Desiderata of Interpretability Research

This section spells out the various desiderata of interpretability research. The demand for interpretability arises when a mismatch occurs between the formal objectives of supervised learning (test-set predictive performance) and the real-world costs in a deployment setting.

Typically, evaluation metrics require only predictions and ground-truth labels. When stakeholders additionally demand interpretability, you might infer the existence of objectives that cannot be captured in this fashion. In other words, because most common evaluation metrics for supervised learning require only predictions, together with ground truth, to produce a score, the very desire for an interpretation suggests that sometimes predictions alone and metrics calculated on them do not suffice to characterize the model. You should then ask, what are these other objectives and under what circumstances are they sought?

Often, real-world objectives are difficult to encode as simple mathematical functions. Otherwise, they might just be incorporated into the objective function and the problem would be considered solved. For example, an algorithm for making hiring decisions should simultaneously optimize productivity, ethics, and legality. But how would you go about writing a function that measures ethics or legality? The problem can also arise when you desire robustness to changes in the dynamics between the training and deployment environments.

*Trust.* Some authors suggest interpretability is a prerequisite for trust.<sup>9,23</sup> Again, what is trust? Is it simply confidence that a model will perform well? If so, a sufficiently accurate model should be demonstrably trustworthy, and interpretability would serve no purpose. Trust might also be defined subjectively. For example, a person might feel more at ease with a well-understood model, even if this understanding serves no obvious purpose. Alternatively, when the training and deployment objectives diverge, trust might denote confidence that the model will perform well with respect to the real objectives and scenarios.

For example, consider the growing use of ML models to forecast crime rates for purposes of allocating police officers. The model may be trusted to make accurate predictions but not to account for racial biases in the training data or for the model's own effect in perpetuating a cycle of incarceration by over-policing some neighborhoods.

Another sense in which an end user might be said to trust an ML model might be if they are comfortable with relinquishing control to it. Through this lens, you might care not only about *how often* a model is right, but also *for which examples* it is right. If the model tends to make mistakes on only those kinds of inputs where humans also make mistakes, and thus is typically accurate whenever humans are accurate, then you might trust the model owing to the absence of any expected cost of relinquishing control. If a model tends to make mistakes for inputs that humans classify accurately, however, then there may always be an advantage to maintaining human supervision of the algorithms.

*Causality.* Although supervised learning models are only optimized directly to make associations, researchers often use them in the hope of inferring properties of the natural world. For example, a simple regression model might reveal a strong association between thalidomide use and birth defects, or between smoking and lung cancer.<sup>29</sup>

The associations learned by supervised learning algorithms are not guaranteed to reflect causal relationships. There could always be unobserved causes responsible for both associated

variables. You might hope, however, that by interpreting supervised learning models, you could generate hypotheses that scientists could then test. For example, Liu et al.<sup>14</sup> emphasize regression trees and Bayesian neural networks, suggesting these models are interpretable and thus better able to provide clues about the causal relationships between physiologic signals and affective states. The task of inferring causal relationships from observational data has been extensively studied.<sup>22</sup> Causal inference methods, however, tend to rely on strong assumptions and are not widely used by practitioners, especially on large, complex datasets.

*Transferability.* Typically, training and test data are chosen by randomly partitioning examples from the same distribution. A model's generalization error is then judged by the gap between its performance on training and test data. Humans exhibit a far richer capacity to generalize, however, transferring learned skills to unfamiliar situations. ML algorithms are already used in these situations, such as when the environment is nonstationary. Models are also deployed in settings where their use might alter the environment, invalidating their future predictions. Along these lines, Caruana et al.<sup>3</sup> describe a model trained to predict probability of death from pneumonia that assigned less risk to patients if they also had asthma. Presumably, asthma was predictive of a lower risk of death because of the more aggressive treatment these patients received. If the model were deployed to aid in triage, these patients might then receive less aggressive treatment, invalidating the model.

Even worse, there are situations, such as machine learning for security, where the environment might be actively adversarial. Consider the recently discovered susceptibility of convolutional neural networks (CNNs). The CNNs were made to misclassify images that were imperceptibly (to a human) perturbed.<sup>26</sup> Of course, this is not overfitting in the classical sense. The models both achieve strong results on training data and generalize well when used to classify held out test data. The crucial distinction is that these images have been altered in ways that, while subtle to human observers, the models

never encountered during training. However, these are mistakes a human would not make, and it would be preferable that models not make these mistakes, either. Already, supervised learning models are regularly subject to such adversarial manipulation. Consider the models used to generate credit ratings; higher scores should signify a higher probability that an individual repays a loan. According to its own technical report, FICO trains credit models using logistic regression,<sup>6</sup> specifically citing interpretability as a motivation for the choice of model. Features include dummy variables representing binned values for average age of accounts, debt ratio, the number of late payments, and the number of accounts in good standing.

Several of these factors can be manipulated at will by credit-seekers. For example, one's debt ratio can be improved simply by requesting periodic increases to credit lines while keeping spending patterns constant.

Similarly, simply applying for new accounts when the probability of acceptance is reasonably high can increase the total number of accounts. Indeed, FICO and Experian both acknowledge that credit ratings can be manipulated, even suggesting guides for improving one's credit rating. These rating-improvement strategies may fundamentally change one's underlying ability to pay a debt. The fact that individuals actively and successfully game the rating system may invalidate its predictive power.

*Informativeness.* Sometimes, decision theory is applied to the outputs of supervised models to take actions in the real world. In another common use paradigm, however, the supervised model is used instead to provide information to human decision-makers, a setting considered by Kim et al.<sup>11</sup> and Huysmans et al.<sup>8</sup> While the machine-learning objective might be to reduce error, the real-world purpose is to provide useful information. The most obvious way that a model conveys information is via its outputs. However, we might hope that by probing the patterns that the model has extracted, we can convey additional information to a human decision maker.

An interpretation may prove informative even without shedding light on

a model's inner workings. For example, a diagnosis model might provide intuition to a human decision maker by pointing to similar cases in support of a diagnostic decision. In some cases, a supervised learning model is trained when the real task more closely resembles unsupervised learning. The real goal might be to explore the underlying structure of the data, and the labeling objective serves only as weak supervision.

*Fair and ethical decision making.* At present, politicians, journalists, and researchers have expressed concern that interpretations must be produced for assessing whether decisions produced automatically by algorithms conform to ethical standards.<sup>7</sup> Recidivism predictions are already used to determine whom to release and whom to detain, raising ethical concerns. How can you be sure predictions do not discriminate on the basis of race? Conventional evaluation metrics such as accuracy or AUC (area under the curve) offer little assurance that ML-based decisions will behave acceptably. Thus, demands for fairness often lead to demands for interpretable models.

### The Transparency Notion of Interpretability

Let's now consider the techniques and model properties that are proposed to confer interpretability. These fall broadly into two categories. The first relates to transparency (that is, how does the model work?). The second consists of post hoc explanations (that is, what else can the model tell me?)

Informally, transparency is the opposite of opacity or "black-boxness." It connotes some sense of understanding the mechanism by which the model works. Transparency is considered here at the level of the entire model (*simulatability*), at the level of individual components such as parameters (*decomposability*), and at the level of the training algorithm (*algorithmic transparency*).

*Simulatability.* In the strictest sense, a model might be called transparent if a person can contemplate the entire model at once. This definition suggests an interpretable model is a simple model. For example, for a model to be fully understood, a human should be able to take the input data together with the parameters of the model and

in reasonable time step through every calculation required to produce a prediction. This accords with the common claim that sparse linear models, as produced by lasso regression,<sup>27</sup> are more interpretable than dense linear models learned on the same inputs. Ribeiro et al.<sup>23</sup> also adopt this notion of interpretability, suggesting that an interpretable model is one that "can be readily presented to the user with visual or textual artifacts."

The trade-offs between model size and computation to apply a single prediction varies across models. For example, in some models, such as decision trees, the size of the model (total number of nodes) may grow quite large compared to the time required to perform inference (length of pass from root to leaf). This suggests simulatability may admit two subtypes: one based on the size of the model and another based on the computation required to perform inference.

Fixing a notion of simulatability, the quantity denoted by *reasonable* is subjective. Clearly, however, given the limited capacity of human cognition, this ambiguity might span only several orders of magnitude. In this light, neither linear models, rule-based systems, nor decision trees are intrinsically interpretable. Sufficiently high-dimensional models, unwieldy rule lists, and deep decision trees could all be considered less transparent than comparatively compact neural networks.

*Decomposability.* A second notion of transparency might be that each part of the model—input, parameter, and calculation—admits an intuitive explanation. This accords with the property of intelligibility as described by Lou et al.<sup>15</sup> For example, each node in a decision tree might correspond to a plain text description (for example, all patients with diastolic blood pressure over 150). Similarly, the parameters of a linear model could be described as representing strengths of association between each feature and the label.

Note this notion of interpretability requires that inputs themselves be individually interpretable, disqualifying some models with highly engineered or anonymous features. While this notion is popular, it should not be accepted blindly. The weights of a linear model might seem intuitive, but they can be

fragile with respect to feature selection and preprocessing. For example, the coefficient corresponding to the association between flu risk and vaccination might be positive or negative, depending on whether the feature set includes indicators of old age, infancy, or immunodeficiency.

*Algorithmic transparency.* A final notion of transparency might apply at the level of the learning algorithm itself. In the case of linear models, you may understand the shape of the error surface. You can prove that training will converge to a unique solution, even for previously unseen datasets. This might provide some confidence that the model will behave in an online setting requiring programmatic retraining on previously unseen data. On the other hand, modern deep learning methods lack this sort of algorithmic transparency. While the heuristic optimization procedures for neural networks are demonstrably powerful, we do not understand how they work, and at present cannot guarantee a priori they will work on new problems. Note, however, that humans exhibit none of these forms of transparency.

*Post hoc interpretability* represents a distinct approach to extracting information from learned models. While post hoc interpretations often do not elucidate precisely how a model works, they may nonetheless confer useful information for practitioners and end users of machine learning. Some common approaches to post hoc interpretations include natural language explanations, visualizations of learned representations or models, and explanations by example (for example, a particular tumor is classified as malignant because to the model it looks a lot like certain other tumors).

To the extent that we might consider humans to be interpretable, this is the sort of interpretability that applies. For all we know, the processes by which humans make decisions and those by which they explain them may be distinct. One advantage of this concept of interpretability is that opaque models can be interpreted after the fact, without sacrificing predictive performance.


*Text explanations.* Humans often justify decisions verbally. Similarly, one model might be trained to generate predictions, and a separate model,




such as a recurrent neural network language model, to generate an explanation. Such an approach is taken in a line of work by Krening et al.<sup>12</sup> They propose a system in which one model (a reinforcement learner) chooses actions to optimize cumulative discounted return. They train another model to map a model's state representation onto verbal explanations of strategy. These explanations are trained to maximize the likelihood of previously observed ground-truth explanations from human players and may not faithfully describe the agent's decisions, however plausible they appear. A connection exists between this approach and recent work on neural image captioning in which the representations learned by a discriminative CNN (trained for image classification) are co-opted by a second model to generate captions. These captions might be regarded as interpretations that accompany classifications.

In work on recommender systems, McAuley and Leskovec<sup>18</sup> use text to explain the decisions of a latent factor model. Their method consists of simultaneously training a latent factor model for rating prediction and a topic model for product reviews. During training they alternate between decreasing the squared error on rating prediction and increasing the likelihood of review text. The models are connected because they use normalized latent factors as topic distributions. In other words, latent factors are regularized such that they are also good at explaining the topic distributions in review text. The authors then explain user-item compatibility by examining the top words in the topics corresponding to matching components of their latent factors. Note that the practice of interpreting topic models by presenting the top words is itself a post hoc interpretation technique that has invited scrutiny.<sup>4</sup> Moreover note we have only spoken to the form factor of an explanation (that it consists of natural language), but not what precisely constitutes correctness. So far, the literature has dodged the issue of correctness, sometimes punting the issue by embracing a subjective view of the problem and asking people what they prefer.

*Visualization.* Another common approach to generating post hoc



**While post hoc interpretations often do not elucidate precisely how a model works, they may confer useful information for practitioners and end users of machine learning.**



interpretations is to render visualizations in the hope of determining qualitatively what a model has learned. One popular method is to visualize high-dimensional distributed representations with t-distributed stochastic neighbor embedding (t-SNE),<sup>28</sup> a technique that renders 2D visualizations in which nearby data points are likely to appear close together.

Mordvintsev et al.<sup>20</sup> attempt to explain what an image classification network has learned by altering the input through gradient descent to enhance the activations of certain nodes selected from the hidden layers. An inspection of the perturbed inputs can give clues to what the model has learned. Likely because the model was trained on a large corpus of animal images, they observed that enhancing some nodes caused certain dog faces to appear throughout the input image.

In the computer vision community, similar approaches have been explored to investigate what information is retained at various layers of a neural network. Mahendran and Vedaldi<sup>17</sup> pass an image through a discriminative CNN to generate a representation. They then demonstrate the original image can be recovered with high fidelity even from reasonably high-level representations (level 6 of an AlexNet) by performing gradient descent on randomly initialized pixels. As before with text, discussions of visualization focus on form factor and appeal, but we still lack a rigorous standard of correctness.

*Local explanations.* While it may be difficult to describe succinctly the full mapping learned by a neural network, some of the literature focuses instead on explaining what a neural network depends on locally. One popular approach for deep neural nets is to compute a saliency map. Typically, they take the gradient of the output corresponding to the correct class with respect to a given input vector. For images, this gradient can be applied as a mask, highlighting regions of the input that, if changed, would most influence the output.<sup>25,30</sup>


Note that these explanations of what a model is focusing on may be misleading. The saliency map is a local explanation only. Once you move a single pixel,

you may get a very different saliency map. This contrasts with linear models, which model global relationships between inputs and outputs.


Another attempt at local explanations is made by Ribeiro et al.<sup>23</sup> In this work, the authors explain the decisions of any model in a local region near a particular point by learning a separate sparse linear model to explain the decisions of the first. Strangely, although the method's appeal over saliency maps owes to its ability to provide explanations for non-differentiable models, it is more often used when the model subject to interpretation is in fact differentiable. In this case, what is provided, besides a noisy estimate of the gradient, remains unclear. In this paper, the explanation is offered in terms of a set of superpixels. Whether or not this is more informative than a plain gradient may depend strongly on how one chooses the superpixels. Moreover, absent a rigorously defined objective, who is to say which hyperparameters are correct?

*Explanation by example.* One post hoc mechanism for explaining the decisions of a model might be to report (in addition to predictions) which other examples are most similar with respect to the model, a method suggested by Caruana et al.<sup>2</sup> Training a deep neural network or latent variable model for a discriminative task provides access to not only predictions but also the learned representations. Then, for any example, in addition to generating a prediction, you can use the activations of the hidden layers to identify the  $k$ -nearest neighbors based on the proximity in the space learned by the model. This sort of explanation by example has precedent in how humans sometimes justify actions by analogy. For example, doctors often refer to case studies to support a planned treatment protocol.

In the neural network literature, Mikolov et al.<sup>19</sup> use such an approach to examine the learned representations of words after training the word2vec model. Their model is trained for discriminative skip-gram prediction, to examine which relationships the model has learned they enumerate nearest neighbors of words based on distances calculated in the latent space. Kim et al.<sup>10</sup> and Doshi-Velez et al.<sup>5</sup> have done



## An inspection of the perturbed inputs can give clues to what the model has learned.



related work in Bayesian methods, investigating case-based reasoning approaches for interpreting generative models.

### Discussion

The concept of interpretability appears simultaneously important and slippery. Earlier, this article analyzed both the motivations for interpretability and some attempts by the research community to confer it. Now let's consider the implications of this analysis and offer several takeaways.

► *Linear models are not strictly more interpretable than deep neural networks.* Despite this claim's enduring popularity, its truth value depends on which notion of interpretability is employed. With respect to algorithmic transparency, this claim seems uncontroversial, but given high-dimensional or heavily engineered features, linear models lose simulatability or decomposability, respectively.

When choosing between linear and deep models, you must often make a tradeoff between algorithmic transparency and decomposability. This is because deep neural networks tend to operate on raw or lightly processed features. So, if nothing else, the features are intuitively meaningful, and post hoc reasoning is sensible. To get comparable performance, however, linear models often must operate on heavily hand-engineered features. Lipton et al.<sup>13</sup> demonstrate such a case where linear models can approach the performance of recurrent neural networks (RNNs) only at the cost of decomposability.

For some kinds of post hoc interpretation, deep neural networks exhibit a clear advantage. They learn rich representations that can be visualized, verbalized, or used for clustering. Considering the desiderata for interpretability, linear models appear to have a better track record for studying the natural world, but there seems to be no theoretical reason why this must be so. Conceivably, post hoc interpretations could prove useful in similar scenarios.

► *Claims about interpretability must be qualified.* As demonstrated here, the term interpretability does not reference a monolithic concept. To be meaningful, any assertion regarding interpretability should fix a specific definition. If the model satisfies a form

of transparency, this can be shown directly. For post hoc interpretability, work in this field should fix a clear objective and demonstrate evidence that the offered form of interpretation achieves it.

► *In some cases, transparency may be at odds with the broader objectives of AI (artificial intelligence).* Some arguments against black-box algorithms appear to preclude any model that could match or surpass human abilities on complex tasks. As a concrete example, the short-term goal of building trust with doctors by developing transparent models might clash with the longer-term goal of improving health care. Be careful when giving up predictive power that the desire for transparency is justified and not simply a concession to institutional biases against new methods.

► *Post hoc interpretations can potentially mislead.* Beware of blindly embracing post hoc notions of interpretability, especially when optimized to placate subjective demands. In such cases, one might—deliberately or not—optimize an algorithm to present misleading but plausible explanations. As humans, we are known to engage in this behavior, as evidenced in hiring practices and college admissions. Several journalists and social scientists have demonstrated that acceptance decisions attributed to virtues such as leadership or originality often disguise racial or gender discrimination.<sup>21</sup> In the rush to gain acceptance for machine learning and to emulate human intelligence, we should all be careful not to reproduce pathological behavior at scale.

## Future Work

There are several promising directions for future work. First, for some problems, the discrepancy between real-life and machine-learning objectives could be mitigated by developing richer loss functions and performance metrics. Exemplars of this direction include research on sparsity-inducing regularizers and cost-sensitive learning. Second, this analysis can be expanded to other ML paradigms such as reinforcement learning. Reinforcement learners can address some (but not all) of the objectives of interpretability research by directly modeling interaction between

models and environments. This capability, however, may come at the cost of allowing models to experiment in the world, incurring real consequences.

Notably, reinforcement learners are able to learn causal relationships between their actions and real-world impacts. Like supervised learning, however, reinforcement learning relies on a well-defined scalar objective. For problems such as fairness, where we struggle to verbalize precise definitions of success, a shift of the ML paradigm is unlikely to eliminate the problems we face. **C**

## Related articles on queue.acm.org

### Accountability in Algorithmic Decision Making

Nicholas Diakopoulos

<https://queue.acm.org/detail.cfm?id=2886105>

### Black Box Debugging

James A. Whittaker and Herbert H. Thompson

<https://queue.acm.org/detail.cfm?id=966807>

### Hazy: Making It Easier to Build and Maintain Big-Data Analytics

Arun Kumar, Feng Niu, and Christopher Ré

<https://queue.acm.org/detail.cfm?id=2431055>


## References

1. Athey, S. and Imbens, G.W. Machine-learning methods 2015; <https://arxiv.org/abs/1504.01132v1>.
2. Caruana, R., Kangaroo, H., Dionisio, J. D., Sinha, U. and Johnson, D. Case-based explanation of non-case-based learning methods. In *Proceedings of the Amer. Med. Info. Assoc. Symp.*, 1999, 12–215.
3. Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M. and Elhadad, N. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21st SIGKDD Intern. Conf. Knowledge Discovery and Data Mining*, 2017, 1721–1730.
4. Chang, J., Gerrish, S., Wang, C., Boyd-Graber, J.L., Blei, D.M. 2009. Reading tea leaves: how humans interpret topic models. In *Proceedings of the 22nd Intern. Conf. Neural Information Processing Systems*, 2009, 288–296.
5. Doshi-Velez, F., Wallace, B. and Adams, R. Graph-sparse LDA: A topic model with structured sparsity. In *Proceedings of the 29th Assoc. Advance. Artificial Intelligence Conf.*, 2015, 2575–2581.
6. Fair Isaac Corporation (FICO). Introduction to model builder scorecard, 2011; <http://www.fico.com/en/latest-thinking/white-papers/introduction-to-model-builder-scorecard>.
7. Goodman, B. and Flaxman, S. European Union regulations on algorithmic decision-making and a 'right to explanation'; 2016; <https://arxiv.org/abs/1606.08813v3>.
8. Huysmans, J., Dejaeger, K., Mues, C., Vanthienen, J. and Baesens, B. An empirical evaluation of the comprehensibility of decision table, tree- and rule-based predictive models. *J. Decision Support Systems* 57, 1 (2011), 141–154.
9. Kim, B. Interactive and interpretable machine-learning models for human-machine collaboration. Ph.D. thesis. Massachusetts Institute of Technology, Cambridge, MA, 2015.
10. Kim, B., Rudin, C. and Shah, J.A. The Bayesian case model: A generative approach for case-based reasoning and prototype classification. In *Proceedings of the 27th Intern. Conf. Neural Information Processing Systems*, Vol. 2, 1952–1960, 2014.
11. Kim, B., Glassman, E., Johnson, B. and Shah, J. iBCM: Interactive Bayesian case model empowering humans via intuitive interaction. Massachusetts Institute of Technology, Cambridge, MA, 2015.
12. Krenging, S., Harrison, B., Feigh, K., Isbell, C., Riedl, M. and Thomaz, A. Learning from explanations using sentiment and advice in RL. *IEEE Trans. Cognitive and Developmental Systems* 9, 1 (2017), 41–55.
13. Lipton, Z.C., Kale, D.C. and Wetzel, R. Modeling missing data in clinical time series with RNNs. In *Proceedings of Machine Learning for Healthcare*, 2016.
14. Liu, C., Rani, P. and Sarkar, N. 2006. An empirical study of machine-learning techniques for affect recognition in human-robot interaction. *Pattern Analysis and Applications* 9, 1 (2006), 58–69.
15. Lou, Y., Caruana, R. and Gehrke, J. Intelligible models for classification and regression. In *Proceedings of the 18th ACM SIGKDD Intern. Conf. Knowledge Discovery and Data Mining*, 2012, 150–158.
16. Lou, Y., Caruana, R., Gehrke, J. and Hooker, G. Accurate intelligible models with pairwise interactions. In *Proceedings of the 19th ACM SIGKDD Intern. Conf. Knowledge Discovery and Data Mining*, 2013, 623–631.
17. Mahendran, A. and Vedaldi, A. Understanding deep image representations by inverting them. In *Proceedings of the IEEE Conf. Computer Vision and Pattern Recognition*, 2015, 1–9.
18. McAuley, J. and Leskovec, J. Hidden factors and hidden topics: Understanding rating dimensions with review text. In *Proceedings of the 7th ACM Conf. Recommender Systems*, 2013, 165–172.
19. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S. and Dean, J. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th Intern. Conf. Neural Information Processing Systems* 2, 2013, 3111–3119.
20. Mordvintsev, A., Olah, C. and Tyka, M. Inceptionism: Going deeper into neural networks. Google AI Blog; <https://ai.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html>.
21. Mounk, Y. Is Harvard unfair to Asian-Americans? *New York Times* (Nov. 24, 2014); <http://www.nytimes.com/2014/11/25/opinion/is-harvard-unfair-to-asian-americans.html>.
22. Pearl, J. *Causality*. Cambridge University Press, Cambridge, MA, 2009.
23. Ribeiro, M.T., Singh, S. and Guestrin, C. 'Why should I trust you?' Explaining the predictions of any classifier. In *Proceedings of the 22nd SIGKDD Intern. Conf. Knowledge Discovery and Data Mining*, 2016, 1135–1144.
24. Ridgeway, G., Madigan, D., Richardson, T. and O'Kane, J. Interpretable boosted naïve Bayes classification. In *Proceedings of the 4th Intern. Conf. Knowledge Discovery and Data Mining*, 1998, 101–104.
25. Simonyan, K., Vedaldi, A., Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps, 2013; <https://arxiv.org/abs/1312.6034>.
26. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I. and Fergus, R. Intriguing properties of neural networks, 2013; <https://arxiv.org/abs/1312.6199>.
27. Tibshirani, R. 1996. Regression shrinkage and selection via the lasso. *J. Royal Statistical Society: Series B: Statistical Methodology* 58, 1 (1996), 267–288.
28. Van der Maaten, L. and Hinton, G. Visualizing data using t-SNE. *J. Machine Learning Research* 9 (2008), 2579–2605.
29. Wang, H.-X., Fratiglioni, L., Frisoni, G. B., Viitanen, M. and Winblad, B. Smoking and the occurrence of Alzheimer's disease: Cross-sectional and longitudinal data in a population-based study. *Amer. J. Epidemiology* 149, 7 (1999), 640–644.
30. Wang, Z., Freitas, N. and Lanctot, M. Dueling network architectures for deep reinforcement learning. In *Proceedings of the 33rd Intern. Conf. Machine Learning* 48, 2016, 1995–2003.

**Zachary C. Lipton** (Twitter @zacharylipton or GitHub @zackchase) is an assistant professor at Carnegie Mellon University in Pittsburgh, PA, USA. His work addresses diverse application areas, including medical diagnosis, dialogue systems, and product recommendation. He is the founding editor of the *Approximately Correct* blog and the lead author of *Deep Learning—The Straight Dope*, an open source interactive book teaching deep learning through Jupyter notebooks.

Copyright held by owner/author.  
Publication rights licensed to ACM. \$15.00.



 Article development led by [acmqueue](https://queue.acm.org)  
queue.acm.org

**The best careers are not defined  
by titles or résumé bullet points.**

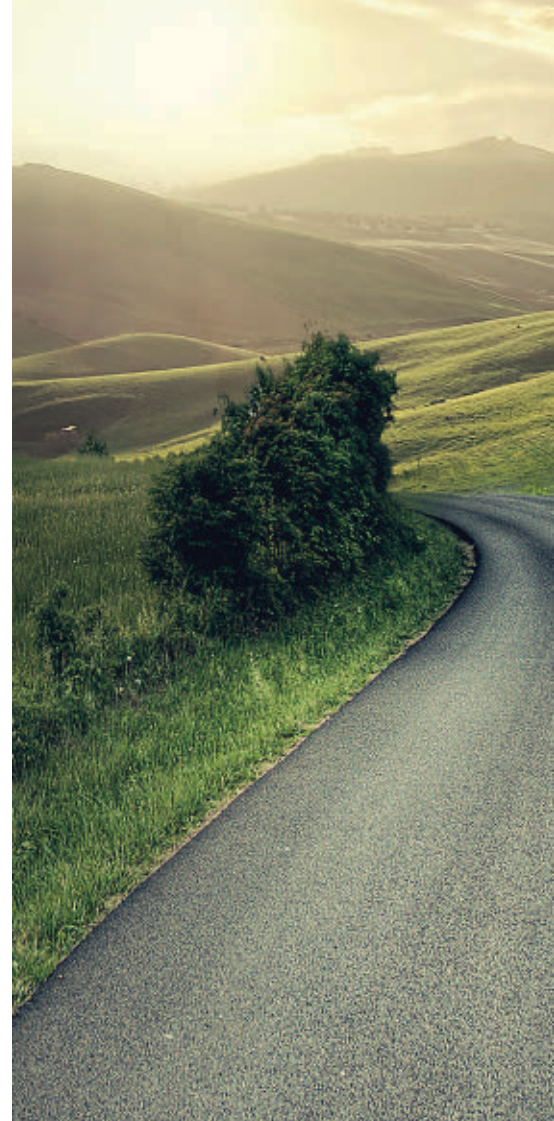
BY KATE MATSUDAIRA

# The Secret Formula for Choosing the Right Next Role

CHANGING JOBS—ESPECIALLY the higher up you get in your career—is a complex process. There are so many factors to consider, and often the factors that stand out most are the ones that matter the least: fancy titles, exciting projects, tempting promises of future success ...

But those factors that seem so valuable in the moment are just that—they are momentary. Your career isn't just about this one next step you are taking. Your career is a journey that will last a long time.

It is smarter to invest in your long-term success. Focus on factors that will increase your career capital and make you a more valuable hire in your next role,



and the one after that, and the one after that.

When you are looking at the options for your next role, there are smarter choices that you can make. Here are the most important factors to consider when picking your next opportunity.

## **Pick a Goal, Not a Title**

A title looks good on a résumé, and might pump up your ego a little bit, but making your job title a serious factor in your job search is a big mistake.

Your title is so much less important than the work you do and the skills you develop while in a role. Those hiring you for your next role will know that. They might see that you were a VP in your last job, but if you don't have any results or skills to show for it, you won't stand out among the many other candidates who were also VPs in their last jobs.

If you want to be truly successful, then your career path should be about acquiring skills and accomplishments, not just upgrading to shinier and fancier titles.



First of all, different titles mean different things in different companies. I have been everything from VP and CTO at successful startups to CEO of my own company, but after years of having executive-level titles, I took a role without one.

If I had rejected that job opportunity because the title was lower than any that I had had in the previous 10 years, I would have missed out on one of the biggest, most life-changing growth opportunities that I have ever had.

Moreover, in that role, instead of being a software engineer I was in the job category of technical program manager (TPM). I had never been a TPM before, and to be honest, it was not a role I identified with. No one would describe me as organized, and I didn't have the background skills; I write code and lead engineering teams.

Even though the title was a demotion, and it was a job family that didn't fit, I still took the position because of what I could gain from it.

Yes, I had to learn some TPM skills, but what was truly valuable about that job was the access it gave me. Because of what my team was focused on, I got to be in meetings with top executives who were running 1,000-plus-person teams. I was presenting to VPs who had decision-making power for a huge organization.

I had a huge scope. Instead of being siloed in one department where I was the boss, I was able to get on the radar of key leaders throughout the organization. I was able to gain influence and visibility; I saw the priorities for the whole company (not just my department), which allowed me to align myself with the most important work being done.

I got to learn, and I gained visibility. I built my network and got to know many people in the organization as a whole. Over time, I earned even bigger influence and control. And my title had nothing to do with it.

When you are looking at different job opportunities, think about the big-

ger picture for your career. Where do you want to be in 10 years? What is your ultimate career goal?

This is different for everyone. Think about where you want to end up, and work backward from there. What skills do you need in order to get there? What steps will you need to take along the way?

Focusing on the short-term win of getting a fancy title or bigger paycheck is a mistake. If a job is not actively putting you into the situations you need in order to grow or make the right contacts, then it is not really the right choice. It will delay you getting where you need to go.

When you are looking at an opportunity, consider whether this role will help you level up your career. Ask yourself the following questions:

- ▶ What skills do I still need to build in order to make progress toward my goals?
- ▶ What benefits will the job afford me that maybe are not visible in the job description?
- ▶ Who will I meet?



- ▶ What is this job setting me up for?
- ▶ What will I have gained from this role in two years, and are those gains valuable to me?

### Pick People, Not Projects

Another easy trap to fall into when picking your next job is to focus too much on the projects you think you will get to work on.

Of course, we all want to work on things that are interesting and exciting or that could make us rich and famous. The truth is projects get canceled all the time. They change and become less exciting. The roles within them change, and you could end up doing legwork that is not actually very interesting or exciting to you.

In college, I got a job working in a lab. I was so happy because I was envisioning myself working on exciting experiments and getting my work published in major journals. While those exciting projects did happen in this lab, I never got to do them. I ended up running the same experiment day after day, collecting the same data over and over again. This is often what research is—you need to make sure any results are statistically significant, so you do the same thing repeatedly.

The projects the lab was working on were exciting, but my life in the lab was not.

It is so important to consider what your day-to-day life will be like in a role. What will you actually spend your time doing? Will it add value to your career? What will you get the chance to learn?

Remember, when you are new to a team, you have no career capital built up with this organization. Career capital is your currency at work; when you provide a lot of concrete, visible value to the team or the organization, you have more leverage to do the things you want, such as work on the most exciting projects or get more flexibility in your schedule.

When you are new, you have not earned this leverage. That means if you are assigned to a boring role on an exciting project, you pretty much just have to do it. Sometimes that can be OK (maybe you actually wanted to learn this boring skill because it will help you get a job you want in

the future), but if it's not, then you are just stuck.

For example, I have a friend who really wanted to work on machine learning, so he joined a team doing that type of work. For the 18 months he didn't get to do anything related to machine learning, and instead was stuck writing deployment scripts and updates to data loaders—work that was much less interesting to him than the project he was on previously.

Projects are never guaranteed, so ensure you understand the specifics and exactly what work you will get the chance to do. Also, instead of thinking just about the work, I recommend thinking also about whom you will be working with.

Basing your decision on the people you will be working with is one of the best ways to pick a job. If you must choose between an exciting project or a great team, always go for the great team.

Some 99% of my happiness in a job has to do with who my manager and coworkers are. I bet it is the same for you. You spend so much time at work; if you work full time, you probably spend as much (or more) time with your coworkers than you do with your friends or family.

In some organizations, it is common to interview with the boss and at least one other member of the team, though this does not always happen. You should always ask for the opportunity to meet more of the people you will be working with.

This has a few benefits:

- ▶ You can meet with the people you will work with every day. Not only will you get a feel for what it will be like working with them, you can also ask them for insight into other aspects of the role. Do they like working there? How much turnover is there on the team? How does collaboration work? Does leadership listen to input on decisions? What are the things they would want to change about the team/company/culture? Why do they work there vs. anywhere else?

- ▶ Your coworkers will feel invested in your success if they are part of the process of hiring you. Think about it—if you met with a candidate you liked and fought for him or her to be hired, wouldn't you be

extra invested in that new hire doing well once he or she joined the team? Even a minimal investment will have a psychological impact on your potential coworkers. If they meet you or interview you, they will have already invested some amount of time in you and will be more inclined to want to see that investment rewarded.

- ▶ You will not be “brand new” on your first day. As humans, we are naturally resistant to change and to new people whom we know nothing about. If you show up on your first day having met no one yet, you are a stranger; your coworkers are more likely to see you as an “outsider” taking up space. Even a short meeting in advance will prime them to see you as familiar the next time you see them. Plus, you will have some baseline knowledge about the team that can help you fit in more quickly, as opposed to starting to learn about the team culture after you have joined.

### Be Smart When You Choose Your Next Role

When you are searching for the next step in your career, don't just think about the surface-level benefits. Drill down on your biggest goals and do a little thinking about whether or not each job will help you get closer to those goals.

The best careers are not defined by titles or résumé bullet points. The smarter you are about what you choose next, the closer you will get to the things you truly want from your life and your work. □

#### Related articles on [queue.acm.org](https://queue.acm.org)

##### 10 Ways to Be a Better Interviewer

Kate Matsudaira

<https://queue.acm.org/detail.cfm?id=3125635>

##### Avoiding Obsolescence

Kode Vicious

<https://queue.acm.org/detail.cfm?id=1781175>

##### A Generation Lost in the Bazaar

Poul-Henning Kamp

<https://queue.acm.org/detail.cfm?id=2349257>

**Kate Matsudaira** ([katemats.com](https://katemats.com)) is an experienced technology leader. She has worked at Microsoft and Amazon and successful startups before starting her own company, Popforms, which was acquired by Safari Books.

Copyright © 2018 held by owner/author. Publication rights licensed to ACM. \$15.00



---

**The interactions between storage and applications can be complex and subtle.**

---

**BY PAT HELLAND**

---

# Mind Your State for Your State of Mind

APPLICATIONS HAVE HAD an interesting evolution as they have moved into the distributed and scalable world. Similarly, storage and its cousin databases have changed side by side with applications. Many times, the semantics, performance, and failure models of storage and applications do a subtle dance as they change in

support of changing business requirements and environmental challenges. Adding scale to the mix has really stirred things up. This article looks at some of these issues and their impact on systems.

Before database transactions, there were complexities in updating data, especially if failures happened. This held true even though the systems were centralized and avoided the complexities presented by distribution. Database transactions dramatically simplified the life of application developers. It was great while it lasted ...

As solutions scaled beyond a single database, life got ever more challenging. First, we tried to make multiple databases look like one database. Then, we were hooking multiple applications together using service-oriented architecture (SOA). In SOA, each service had

its own discrete database with its own transactions but used messaging to coordinate across boundaries. Soon, we were using microservices, each of which likely did not have its own data but reached directly to a distributed store shared across many separate services. This scaled better—if you got the implementation right.

Different types of distributed stores offer various average speeds, variation in responsiveness, capacity, availability, and durability. Diverse application patterns use the stored data for distinct purposes. They provide various guarantees to their users based largely on their use of storage. These different guarantees from the app sometimes show variations in what the users see in semantics, response time, durability, and more. While these can be surprising, it may be OK. What matters is the

fulfillment of the business needs and clarity of expectations.

This article provides a partial taxonomy of diverse storage solutions available over a distributed cluster. Part of this is an exploration of the interactions among different features of a store. The article then considers how distinct application patterns have grown over time to leverage these stores and the business requirements they meet. This may have surprising implications.

### The Evolution of State, Storage, And Computing ... At Least So Far

This section starts by examining some of the profound changes that have occurred in both storage and computation. The focus then turns to a discussion of both durable state and session state and how they have evolved over time. Finally, there is a brief reminder of how data is treated differently inside a classic database and outside as it moves across trust and transactional boundaries.

**Trends in storage and computing.** Changes in storage and computing have put demands on how storage is accessed and the expected behavior in doing so. This is especially interesting as work is smeared over pools of small computation known as microservices.

*Storage has evolved.* It used to be that storage was only directly attached to your computer. Then came shared appliances such as storage area networks (SANs). These are big, expensive devices with a lot of sophisticated software and hardware to provide highly available storage to a bunch of servers attached to them. This led to storage clusters of commodity servers contained in a network.

*Computing has evolved.* A few decades ago, it was only a single process on a single server. Years went by before people started worrying about communicating across multiple processes on a single server. Then the world moved on with great excitement to RPCs (remote procedure calls) across a tiny cluster of servers. At the time, we didn't think about trust since everyone was in the same trust zone. We were all in the family!

In the 2000s, the concept of services or SOA began to emerge, sometimes under different names.<sup>6</sup> The basic aspect of a service is *trust isolation*. This natu-

rally leads to applications and app code encapsulating the data so the distrusted outsider cannot just modify the data with abandon.

As the industry started running stuff at huge scale, it learned that busting a service into smaller microservices has a couple of big advantages:

- ▶ Better engineering. Breaking your services (that is, trust boundaries) into smaller pieces allows better engineering flexibility as small teams make quicker changes.

- ▶ Better operability. Making these smaller pieces stateless and restartable allows for more resilient operations as failures, rolling upgrades of versions, and adjustments for varying demand are dynamically handled.

Microservices became an essential part of the software engineering and operations landscape.

### Careful Replacement Variations

- ▶ A write may trash the previous value ... write somewhere else first.

- ▶ A client crash may interrupt a sequence of writes ... plan carefully.

*Computing's use of storage has evolved.* It has been quite a wild ride of application changes as their use of storage has evolved:

- ▶ Direct file I/O used *careful replacement* for recoverability. Careful replacement is a technique that is at least as old as the 1960s. It involves thoughtful ordering of changes to durable storage such that failures can be tolerated.

- ▶ Transactional changes were supported for application developers, providing a huge improvement. It meant the app developer did not need to be so careful when dealing with storage. It also allowed a grouping of changes to records so a bunch of records were atomically updated. This was a *lot* easier. SANs implemented the required careful replacement for the hardware storage, allowing bigger and better databases. Databases evolved to support two-tier and *N*-tier applications using transactional updates.

- ▶ Key-value stores offered more scale but less declarative functionality for processing the application's data. Multirecord transactions were lost as scale was gained.

There have been and continue to be significant changes to the style of computation, to storage, and to how these application patterns are used to access storage.

This is only a partial list of storage and compute models. It is not meant to be complete.

**Challenges in modern microservice-based applications.** These days, microservices power many scalable apps. Microservices are pools of identical or equivalent services running over a collection of servers. Incoming requests are load balanced across the pool.

---

When a request waits for a microservice, any one from the same pool will do the job. Sometimes, systems implement *affinitization*, where a subsequent request is likely to go to the same specific microservice. Still, the outcome must be correct if you land on any of the microservices.

---

Microservices help scalable systems in two broad ways:

- ▶ *Improved software engineering.* Building systems consisting of small and independent microservices results in agility. Teams owning the microservices must be accountable and have independence and ownership. When something needs changing, change it. When something is broken, the owning team is responsible.

- ▶ *Improved operations.* Health-mediated deployment allows for slow rollout of new versions into the running system. By watching the system's health, new versions can be rolled back. These rolling upgrades to the microservices can be sensitive to fault zones so an independent failure during a flaky upgrade is not too damaging. Simply having a lot of separate and equivalent microservices means a failure of one or more of them is automatically repaired.

Durable state is not usually kept in microservices. Instead, it is kept in back-end databases, key-value stores, caches, or other things. The remainder of this article looks at some of these.

Microservices cannot easily update the state across all of the microservices in the pool. This is especially true when they are coming and going willy-nilly. It is common to keep the latest

state out of reach of the microservices and provide older versions of the state that are accessible in a scalable cache. Sometimes, this leads to read-through requests by the scalable cache to durable state that is not directly addressable to the calling microservice.

This is now becoming a tried and true pattern. Figure 1 is taken from a 2007 paper by DeCandia et al. on Amazon's Dynamo.<sup>2</sup> While the nomenclature is slightly different, it shows three tiers of microservices accessing a back-end tier of different stores.

**Durable state and session state.** *Durable state* is stuff that gets remembered across requests and persists across failures. This may be captured as database data, file-system files, key values, and more. Durable state is updated in a number of different ways, largely dependent on the kind of store holding it. It may be changed by single updates to a key value or file, or it may be changed by a transaction or distributed transaction implemented by a database or other store.

*Session state* is the stuff that gets remembered across requests in a session but not across failures. Session state exists within the endpoints associated with the session. Multioperation transactions use a form of session state.<sup>7</sup>

Session state is hard to do when the session is smeared across service instances. If different microservices in the pool process subsequent messages in the transaction, session state is challenging to implement. It's difficult to retain session state at the instance when the next message to the pool may land at a different service instance.

**Data on the outside versus data on the inside.** The 2005 paper "Data on the Outside Versus Data on the Inside"<sup>5</sup> speaks about the fundamental differences between data kept in a locked transactional store (for example, a relational database) and data kept in other representations.

Data on the inside refers to locked transactionally updated data. It lives in one place (for example, a database) and at one time, the transactional point in time.

Data on the outside is unlocked and immutable, although it may be versioned with a sequence of versions that are in their own right immutable.

Outside data always has some form of a unique identifier such as a URI (uniform resource identifier) or a key. The identifier may be implicit within a session or an environment. Outside data typically is manifest as a message, file, or key-value pair.

### The Evolution of Durable State Semantics

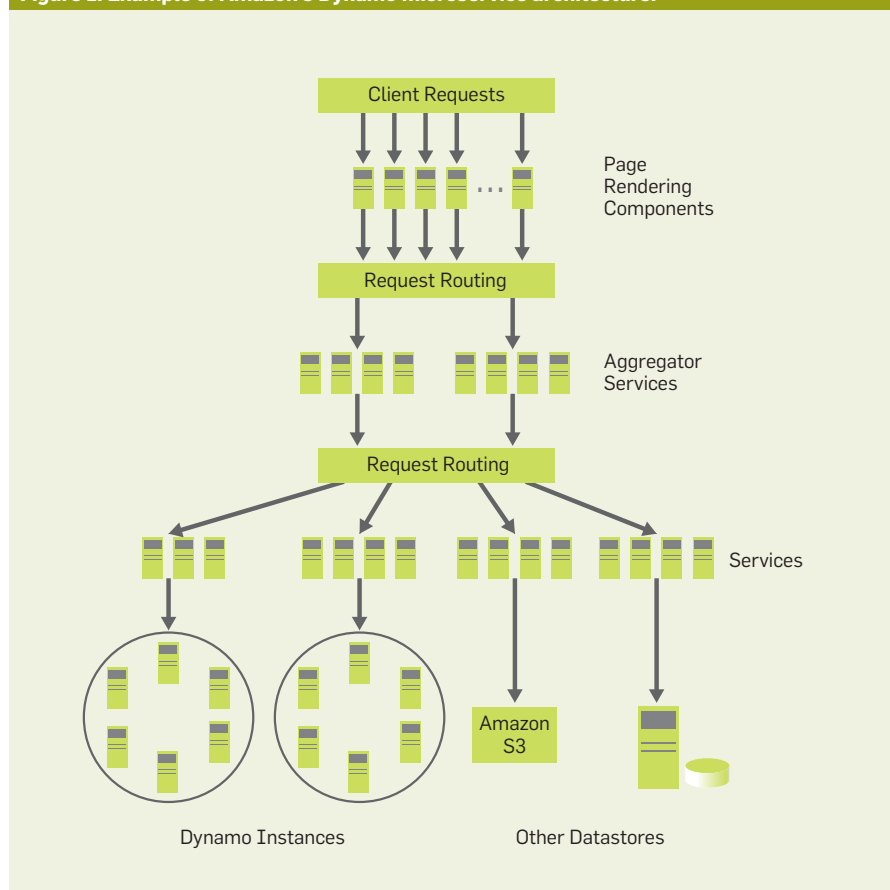
Storage systems and databases have evolved through the decades and so have the semantics of updating their state. This section begins in the bad old days when I first started building systems. Back in the 1970s and 1980s, disk storage had to be carefully updated to avoid trashing disk blocks. From there, we move forward to atomic record updates and the challenges that arose before transactions. When transactions came along a lot of things got a lot easier—if you were making a change at one place and one time. Adding cross-database and cross-time behavior led to the same challenges you had with more primitive storage systems. This was helped by using messaging subsystems to glue stuff together.

Then, an interesting development in storage occurred. Some stores are fast but sometimes return stale values. Others always return the latest value but occasionally stall when one of the servers is slow. This section shows how predictable answers result in unpredictable latencies.<sup>10</sup> Finally, it examines the role immutable data can play in supporting very large systems with predictable answers and response times for some business functions.

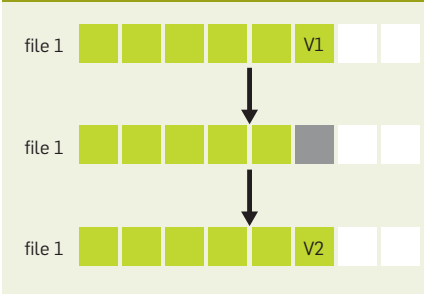
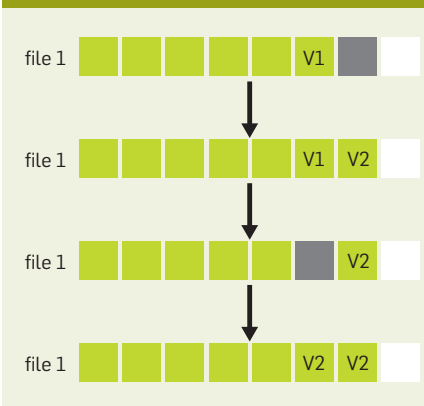
**Careful replacement of disk blocks.** It used to be, back in the 1970s and 1980s, that a disk write might leave data unreadable. The write went through a number of state changes from the old V1 version, to unreadable garbage, to the new V2 version. When the disk head was writing a block, the magnetic representation of the bits in the block would be turned to mush on the way to being updated to the new version. A power failure would cause you to lose the old value (see Figure 2).

When implementing a reliable application, it's essential that you do not lose the old value of the data. For example, if you're implementing the trans-

Figure 1. Example of Amazon's Dynamo microservice architecture.





**Figure 2. V1 is trashed before V2 is written.****Figure 3. “Ping-Pong” technique delays overwrite of V1.**

action system for a database, it's really bad to lose the most recently committed transactions because the partially full last block of your transaction log is being rewritten. One trick to avoid this is to take turns writing to mirrored logs on different disks. Only after knowing for sure that mirror A has the new block do you write it to mirror B. After a crash, you rewrite the last block of the log onto both mirrors to ensure a consistent answer.

Another well-known technique, especially for the tail of the log, is called *ping-pong*.<sup>4</sup> In this approach, the last (and incomplete) block of the log is left where it lies at the end of the log. The next version of that block, containing the previous contents and more, is written to a later block. Only after the extended contents are durable on the later block will the new version overwrite the earlier version. In this fashion, there are no windows in which a power failure will lose the contents of the log (see Figure 3).

**Careful replacement for record writes.** Updates to records in pre-database days didn't have transactions. Assuming each record write was atomic, you still couldn't update two records and get any guarantees they would both

be updated. Typically, you would write to record X, wait to know it's permanent, and then write to record Y.

So, could you untangle the mess if a crash happened?

Frequently, there was an application-dependent pattern that provided insight into the order you needed to write. After a crash and restart:

- ▶ If record A was updated but record B was not written, the application can clean up the mess.

- ▶ If record B was updated but record A was not written, the application could not cope and could not recover.

An example of careful replacement for records is message queuing. If the application writes and confirms the presence of a message in a queue (call it record A), and the work to process that message is idempotent, then the application can cope with crashes based on careful replacement for records. Idempotent means it is correct if restarted.<sup>4,7</sup>

**Transactions and careful replacement.** Transactions bundle and solve *careful record replacement*. Multiple application records may be updated in a single transaction, and they are all-or-nothing. The database system ensures the record updates are atomic.

- ▶ Databases automatically handle any challenges with *careful storage replacement*. Users are not aware of the funky failure behaviors that may occur when systems crash or power fails. If present, databases also support distributed transactions over a small number of intimate database servers.

- ▶ Work across time (that is, workflow) needs *careful transactional replacement*. While the set of records in a transaction is atomically updated with the help of the database, long-running workflows<sup>3,4</sup> are essential to accomplish correct work over time. Failures, restarts, and new work can advance the state of the application transaction by transaction. Work across time leverages message processing.

- ▶ Work across space (that is, across boundaries) also needs *careful transactional replacement*. Different systems, applications, departments, and/or companies have separate trust boundaries and typically do not do transactions across them. Work across space necessitates work across time, trans-

action by transaction. This leads to messaging semantics.

**Messaging semantics.** In transactional messaging a transaction makes a bunch of changes to its data and then expresses a desire to send a message. This desire is atomically recorded with the transaction. A transaction may atomically consume an incoming message. That means the work of the transaction, including changes to the application data, occurs if, and only if, the incoming message is consumed.

It is possible to support the semantics of exactly-once delivery. The desire to send is atomically committed with the sending transaction. A committed desire to send a message causes one or more transmissions. The system retries until the destination acknowledges it has received the message in its queue. The message must be processed at the receiver at most once. This means it must be idempotently processed (see Figure 4).

There are challenges with at-most-once processing at the destination. To accomplish this, you need to remember the messages you have processed so you don't process them twice. But how do you remember the messages? You have to detect duplicates. How long do you remember? Does the destination split? Does the destination move? If you mess this up, will the application process the message more than once? What if the message is being delivered to a microservice-based application? Where is the knowledge of the set of processed messages kept?

**Read your writes? Yes? No?** It used to be, back in the day, if you wrote something, you could read it. Now, it's not always that simple. Consider the following:

Linearizable stores offer read-your-write behavior. In a linearizable store each update creates a new version of the value, and the store never returns an old value or a different value. It always returns the latest in a linear series of values.

---

Linearizable stores will sometimes delay for a *loooooong* time.

To ensure they always give the correct value, they will always update every replica.

If a server is slow or dead and contains one of the replicas, it may take tens of seconds to decide what to do ... Meanwhile, the user waits.

*Nonlinearizable stores* do not offer to read your writes. A nonlinearizable store means there's no guarantee that a write will update all the replicas. Sometimes, a read may find an old value. Reading and writing a nonlinearizable store has a very consistent response time with much higher probability. A read or write can skip over a sick or dead server. Occasionally, this results in an older value coming back from the skipped server. But, hey, it's fast—and predictably so.

Imagine a key/value store where key-K has value V1 and the store keeps it on servers S1, S2, and S3. You decide to update the value to V2. The store tries to change the values on its three servers, but S2 does not answer because it is down. Therefore, the store decides to write V2 onto S1, S3, and S4 so that the new value is always written to three servers. Later, when S2 comes up, a read might find the old value V1. This has the following trade-offs:

- ▶ The write of three stores always happens quickly.
- ▶ The store is not linearizable and sometimes returns an old value.

This very useful technique underlies a number of scalable storage systems such as Dynamo<sup>2</sup> and Cassandra.<sup>11</sup>

Cached data offers scalable read throughput with great response time. Key-value pairs live in many servers and are updated by propagating new versions. Each read hits one of the servers and returns one of the versions (see Figure 5).

**Different Stores for Different Uses**

- OK to stall on reads?
- OK to stall on writes?
- OK to return stale versions?
- You can't have everything!

**Immutability: A solid rock to stand on.** When you store immutable data, each lookup always returns the same result.<sup>8</sup> Immutable stores do not ever exhibit update anomalies because you never update them. All you can do is

store a brand-new value for an identifier and, later on, delete it. Many application patterns are based on immutable items.

Imagine a system where you are simply recording stuff you have seen. Everything you know is based on observations. The past is never changed—sort of like an accountant's ledger where nothing is updated. You can put a unique ID on each artifact and look at it later but never change it. This is an extremely common pattern.

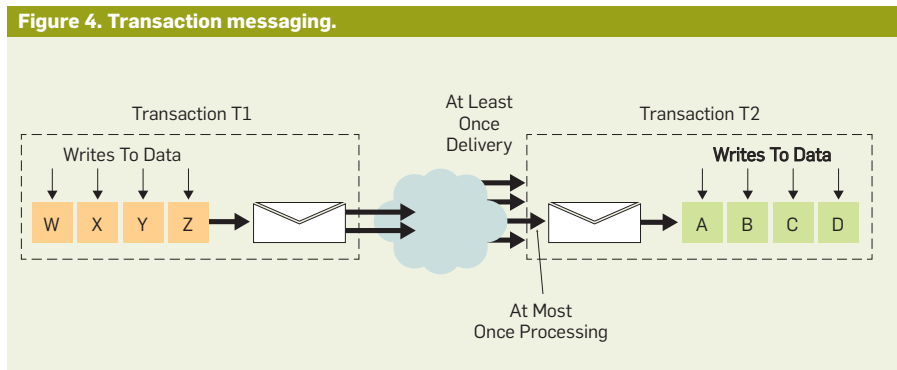
When keeping immutable objects or values in a key/value store, you never get a stale answer. There's only one immutable value for the unique key. That means a nonlinearizable store offers the one and only correct answer. All the store types give the correct answer, just with different characteristics for read and write latencies (see Figure 6). Storing immutable data means you never get a stale version because there is not one.

**Slip-Slidin' Away ...**

This section looks at a number of guarantees that are slipping away. Everyone wishes they had a computational model such as a von Neumann machine,<sup>12</sup> which provides computation, storage, and predictable linear behavior. Once distribution kicks in, however, that's indeed only a wish.

Single-process computation as John von Neumann conceived has evolved to multiprocess- and multiserver-using sessions and session state. These stateful sessions supported composable transactions that spanned multiple records and multiple servers working together. As the work started decomposing into microservices, however, it became hard to use transactions the way they had been used.

To cope with scalable environments, data had to be busted up into key values. Scalable stores worked well for updating a single key at a time but not for atomic transactions across keys. Most



**Figure 5. Different types of storage offer different guarantees.**

	Fast Predictable Reads?	Fast Predictable Writes?	Read Your Writes?
Linearizable Store	No	No	Yes
Non-Linearizable Store	Yes	Yes	No
Scalable Cache	Yes w/Scale	No	No

**Figure 6. Immutable data allows “read-your-write-behavior.”**

	Fast Predictable Reads?	Fast Predictable Writes?	Read Your Writes?
Linearizable Store	No	No	Immutable
Non-Linearizable Store	Yes	Yes	Immutable
Scalable Cache	Yes w/Scale	No	Immutable

of these scalable key-value stores ensured linearizable, strongly consistent updates to their single keys. Unfortunately, these linearizable stores would occasionally cause delays seen by users. This led to the construction of nonlinearizable stores with the big advantage that they have excellent response times for reads and writes. In exchange, they sometimes give a reader an old value.

Finally, this section points out that some uses of data find the correct answer important enough to use careful replacement of the stored values. These uses are not the best for nonlinearizable stores.

Honestly, it ain't like it used to be.

**Same process evolves to different process.** Applications and the database used to run in the same process. A library call to the database code allowed access to the data. Sometimes, multiple applications were loaded together.

Later, the database and applications were split into different processes connected by a session. The session described the session state and had information about the user, transaction in flight, the application being run, and the cursor state and return values.

Later still, the application and database moved to different servers. The session state made that possible.

**Stateful sessions and transactions.** Stateful sessions were a natural outcome of shared processes. You knew who you were talking to and you could remember stuff about the other guy.

Stateful sessions worked well for classic SOA. When talking to a service, you expected a long session with state on each side. Stateful sessions meant the application could do multiple interactions within a transac-

tion. In many circumstances, rich and complex transactions could occur over  $N$ -tier environments, even across multiple back-end databases using distributed transactions.

**Transactions, sessions, and microservices.** Microservices leave much to be desired when it comes to session state. Requests are load balanced through a router, and one of many microservice instances is selected. Usually, later traffic is sent to the same instance but not always. You cannot count on getting back to where you were.

Without session state, you cannot easily create transactions crossing requests. Typically, microservice environments support a transaction within a single request but not across multiple requests.

Furthermore, if a microservice accesses a scalable key-value store as it processes a single request, the scalable key-value store will usually support only atomic updates to a single key. While it won't break the data by failing in the middle of updating a key as older file systems did, programmers are on their own when changing values tied to multiple keys.

**Keys, versions, and nonlinear history.** Each key is represented by some number, string, key, or URI. That key can reference something that's immutable. For example, "*The New York Times*, June 1, 2018, San Francisco Bay Area edition" is immutable across space and time. A key may also reference something that changes over time—for example, "today's *New York Times*."

When a key references something that changes, it can be understood as referencing a sequence of versions, each of which is immutable. By first binding the changing value of the key to a unique version of the key (for example, [Key, Ver-

sion-1]), you can view the version as immutable data. Each version becomes an immutable thing to be kept. Using the extended [Key, Version], you can reference immutable data in the store.

Version history may be linear, meaning one version supersedes the previous one. This is achieved by using a *linearizable store*. Version history may be a directed acyclic graph (DAG). This happens when writing to a *nonlinearizable store*.

Imagine you have a notepad on which to scribble stuff. But you really have multiple notepads. You scribble stuff on whichever notepad is closest to you at the time. When you want to read the information, you look at the closest notepad even if it's not the one you wrote on most recently. Sometimes, you get two notepads next to each other, look at both, and write something in both to consolidate the scribbles. This is the kind of behavior that comes from a nonlinearizable store. Updates do not march forward in linear order.

**Careful replacement and read your writes.** In careful replacement you need to be careful about the ordering of what you update. This is essential to handle some failures, as discussed earlier. Predictable behavior across trust boundaries is needed when working with other companies. It's also essential when doing long-running workflows.

Careful replacement is predicated on read-your-writes behavior, which depends on a linearizable store. Linearizable stores almost always have the property of occasionally stalling when waiting for a bum server.

## Some Example Application Patterns

Let's look at some application patterns and how they impact the management of durable state (see Figure 7).

**Workflow over key-value with careful replacement.** This pattern demonstrates how applications perform workflow when the durable state is too large to fit in a single database.

An object is uniquely identified by its key. Work arrives from the outside via human interaction or messaging. Workflow can be captured in the values. New values replace old ones. The messages are contained as data within the object.<sup>9</sup>

Scalable workflow applications can be built over key-value stores. You must have single-item linearizability (read your writes, see Figure 8.) With a linear

Figure 7. Applications patterns.

workflow over key-value	A traditional workflow application over a scalable collection of key-value data.
transactional blobs-by-ref	A centralized and transactional system managing very large collections of immutable blobs.
e-commerce—shopping cart	The familiar but still surprising world of e-commerce shopping carts.
e-commerce—product catalog	Consider a very large ecommerce product catalog with enormous numbers of product descriptions and huge traffic reading the catalog.
search	Track a ginormous number of document (for example, the entire Web) and organize searchable indices to locate documents by words and phrases. Must scale to ever increasing read workload.



version history, one new version always supersedes the earlier one. A nonlinear history has a DAG version history. In this case, the linearizable behavior of the store also implies that a stall within one of the store servers will stall the write to the store. This is the “must be right” even if it’s not “right now” case.

The workflow implemented by careful replacement will be a mess if you can’t read the last value written. Hence, this usage pattern will *stall* and not be *stale*.

**Transactional blobs-by-ref.** This is a pretty common application pattern. The application runs using transactions and a relational database. It also stores big blobs such as documents, photos, PDFs, videos, music, and more. The blobs can be large and numerous. Hence, these are a challenge to implement directly in the relational database.

Each of these blobs is an immutable set of bits. To modify a blob (for example, editing a photo), you always create a new blob to replace the old one. The immutable blobs typically have a universally unique identifier (UUID) as their key in a scalable key-value store.

Storing immutable blobs in a non-linearizable database does not have any problems with returning a stale version. Since there’s only one immutable version, there are no stale versions.

Storing immutable data in a non-linearizable store enjoys the best of both worlds: it’s both *right* and *right now*.

**E-commerce shopping cart.** In e-commerce, each shopping cart is for a separate customer. There’s no need or desire for cross-cart consistency. Each shopping cart has a unique identity or key.

Customers are very unhappy if their access to a shopping cart stalls. Large e-commerce sites can measure the percentage of abandoned carts and customer sessions when they get slow. Slow carts correspond to a large drop-off in business. Product catalogs, reviews, and more must be fast and responsive or customers leave.

Shopping carts should be *right now* even if they are not *right*. It is measurably better for business and the customer experience to return a stale or otherwise incorrect answer if it can be done quickly. Users are asked to verify the contents of the shopping cart before confirming the sale.

In a non-linearizable store, sometimes multiple old versions of the cart

exist in the version history DAG. Relatively simple shopping-cart semantics facilitate combining different versions of a single user’s shopping cart.<sup>2</sup>

**E-commerce—Product catalog.** Product catalogs for large e-commerce sites are processed offline and stuffed into large scalable caches. Feeds from partners and crawls of the Web are crunched to produce a sanitized and hopefully consistent collection of product-catalog entries.

Each product in the catalog has a unique identifier. Typically, the identifier takes you to a partition of the catalog. The partition has a bunch of replicas, each containing many product descriptions (see Figure 9). One typical implementation of a scalable product cache has partitions with replicas. In this depiction, the columns are partitions and the rows depict replicas. The back-end processing produces new product descriptions that are distributed with pub-sub. Incoming requests are sent to

the partition for the product identifier and then load-balanced across replicas.

Back-end processing of the feeds and crawls, as well as the pub-sub distribution of updates to the caches, are throughput sensitive, not latency-sensitive. Different replicas may be updated

Figure 8. Linear vs. nonlinear histories.

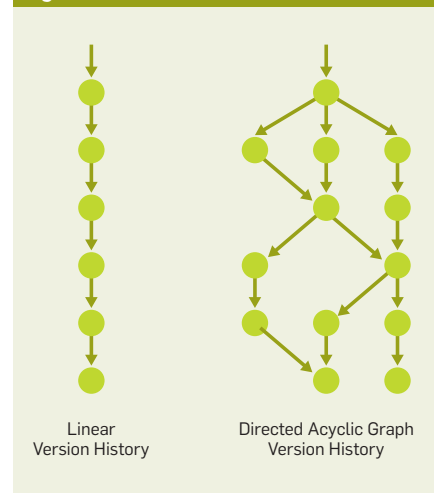


Figure 9. Partitions with replicas.

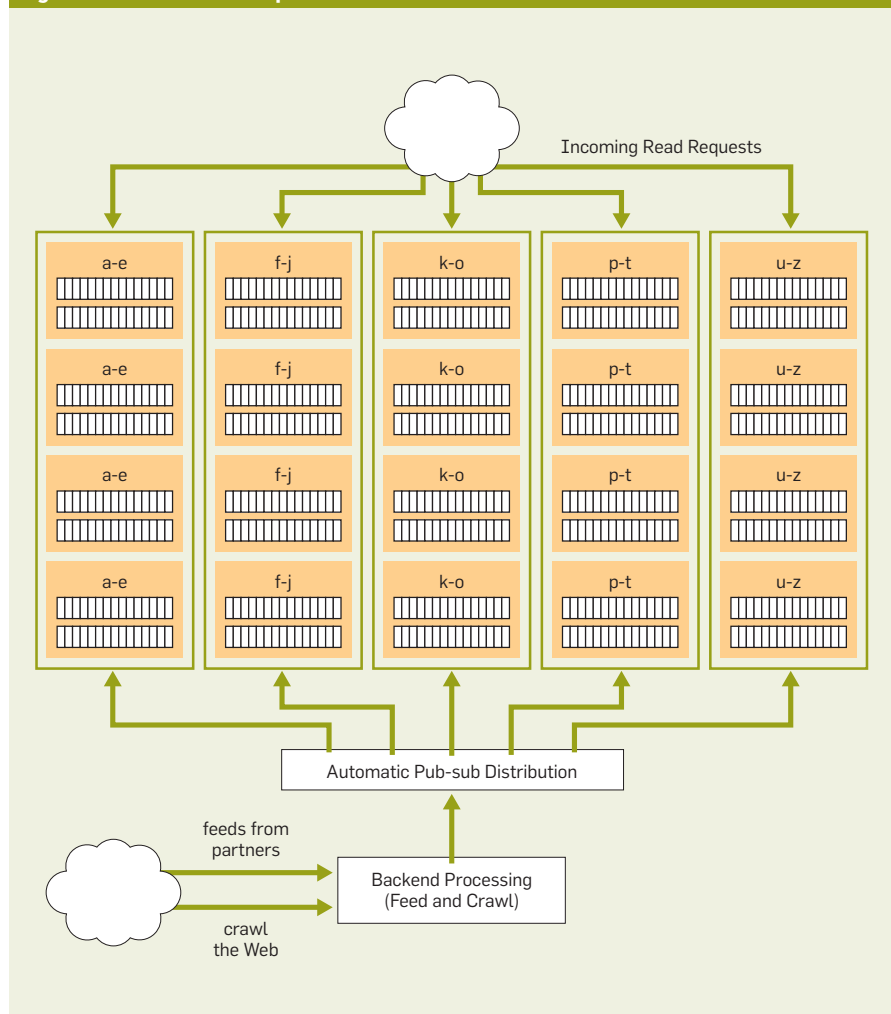


Figure 10. Application pattern trade-offs.

Application pattern	Predictable read latency?	Predictable write latency?	Reads your writes?	Trade-offs
Careful replacement with k/v	No	No	Yes	Works across multiple key/values
Tx'l blobs for ref	Yes	Yes	Immutable	Non-linearizable plus immutable
E-commerce—shopping cart	Yes	Yes	No	Sometimes give stale results
E-commerce—product catalog	Yes	No	No	Scalable cache means that stale is ok
Search	Yes	No	No	Scalable cache plus search

asynchronously, meaning it is not surprising to read a new version of the description, retry, and then get an old version from a cache replica that's not yet updated.

User lookups are very sensitive to latency. Just as shopping cart response times must be fast, product-catalog lookups must be fast. It is common for a client working to display the description of a product to wait for an answer, time out, and retry to a different replica if necessary to ensure the latency for the response is fast.

Note the management of the short latency depends on the fact that any version of the product-catalog description is OK. This is another example of the business needing an answer *right now* more than it needs the answer to be *right*.

**Search.** Say you are building a search system for the contents of the Web. Web crawlers feed search indexers. Each document is given a unique ID. Search terms are identified for each document. The index terms are assigned to a shard.

Updates to the index are not super latency-sensitive. Mostly, changes observed by crawling the Web are not latency-sensitive. Other than time-sensitive news feeds, the changes need not be immediately visible. When a random document is produced at some remote location in the world, it might take a while to be seen.

Search results are, however, sensitive to latency. In general, a search request from a user is fed into servers that ask all of the shards for matching results. This looks a lot like the product catalog depicted in Figure 9, but the user requests hit all the shards, not just one of them.

It's very important that searches get quick results, or users will get frus-

trated. This is aggravated by the need to hear back from all the servers. If any server is a laggard, the response is delayed. The mechanism for coping with this at Google is beautifully described in the 2013 article "The Tail at Scale."<sup>1</sup>

In search, it is OK to get stale answers, but the latency for the response must be short. There's no notion of linearizable reads nor of read-your-writes. Search clearly needs to return answers *right now* even if they are not *right*.

**It's about the application pattern.** Each application pattern shows different characteristics and trade-offs, shown in Figure 10.

### Conclusion

State means different things. Session state captures stuff across requests but not across failures. Durable state remembers stuff across failures.

Increasingly, most scalable computing consists of microservices with *stateless interfaces*. Microservices need partitioning, failures, and rolling upgrades, and this implies that stateful sessions are problematic. Microservices may call other microservices to read data or get stuff done.

Transactions across stateless calls are usually not supported in microservice solutions. Microservices and their load-balanced service pools make server-side session state difficult, which, in turn, makes it difficult to have transactions across calls and objects. Without transactions, coordinated changes across objects in durable state need to use the careful replacement technique in which updates are ordered, confirmed, and idempotent. This is challenging to program but is a natural consequence of microservices, which have emerged as the

leading technique to support scalable applications.

Finally, different applications demand different behaviors from durable state. Do you want it *right* or do you want it *right now*? Human beings usually want an answer *right now* rather than *right*. Many application solutions based on object identity may be tolerant of stale versions. Immutable objects can provide the best of both worlds by being both *right* and *right now*.

Consider your application's requirements carefully. If you are not careful, you will have problems with your state that you will definitely mind. □

### Related articles on queue.acm.org

#### Non-volatile Storage

Mihir Nanavati et al.

<https://queue.acm.org/detail.cfm?id=2874238>

#### Network Applications Are Interactive

Antony Alappatt

<https://queue.acm.org/detail.cfm?id=3145628>

#### Storage Systems: Not Just a Bunch of Disks Anymore

Erik Riedel

<https://queue.acm.org/detail.cfm?id=864059>

### References

- Dean, J. and Barosso, L.A. The tail at scale. *Commun. ACM* 56, 2 (Feb. 2013), 74–80.
- DeCandia, G. et al. Dynamo: Amazon's highly available key-value store. In *Proceedings of the 21st ACM SIGOPS Symp. Operating System Principals*, 2007, 205–220.
- Garcia-Molina, H. and Salem, K. Sagas. In *Proceedings of the ACM SIGMOD Conf. Management of Data*, 1987, 249–259; <https://www.cs.cornell.edu/andru/cs711/2002fa/reading/sagas.pdf>
- Gray, J. and Reuter, A. *Transaction Processing: Concepts and Techniques*. Morgan Kaufmann, Burlington, MA, 1992, 508–509.
- Helland, P. Data on the outside versus data on the inside. In *Proceedings of the Conf. Innovative Database Research*, 2005.
- Helland, P. Fiefdoms and emissaries, 2002. download. [microsoft.com/documents/uk/msdn/architecture/.../fiefdoms\\_emissaries.ppt](https://www.microsoft.com/documents/uk/msdn/architecture/.../fiefdoms_emissaries.ppt).
- Helland, P. Idempotence is not a medical condition. *acmqueue* 10, 4 (2012), 56–65.
- Helland, P. Immutability changes everything. *acmqueue* 13, 9 (2016); <https://queue.acm.org/detail.cfm?id=2884038>.
- Helland, P. Life beyond distributed transactions. *acmqueue* 14, 5 (2016); <https://queue.acm.org/detail.cfm?id=3025012>.
- Helland, P. Standing on distributed shoulders of giants. *acmqueue* 14, 2 (2016); <https://queue.acm.org/detail.cfm?id=2953944>.
- Lakshman, A. and Malik, P. Cassandra: A decentralized structured storage system. *ACM SIGOPS Operating Systems Review* 44, 2 (2010), 35–40.
- von Neumann, J. First draft of a report on the EDVAC. *IEEE Annals of the History of Computing* 15, 4 (1993), 27–75.

**Pat Helland** has been implementing transaction systems, databases, application platforms, distributed systems, fault-tolerant systems, and messaging systems since 1978. He currently works at Salesforce.

Copyright held by owner/author.  
Publication rights licensed to ACM. \$15.00.

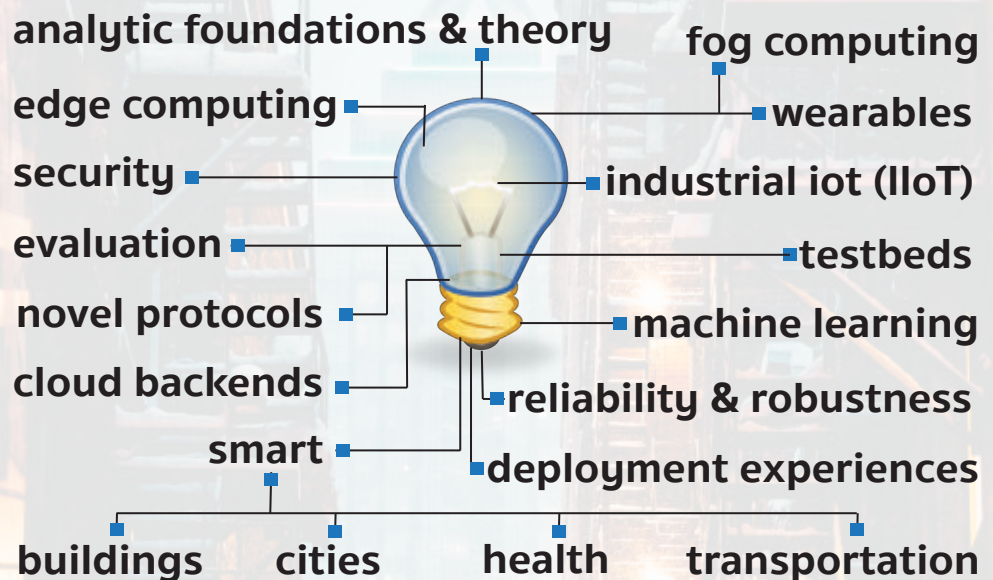
Montreal, QC  
Canada

April 15-18, 2019



ACM/IEEE IoTDI is the premier venue for all topics related to the Internet of Things. This conference is an interdisciplinary forum to discuss challenges, technologies and emerging directions in system design and implementation that pertain to IoT.

Selected papers will be invited to the  
ACM Transactions on the Internet of Things (TIOT)



### General Co-Chairs

Klara Nahrstedt, University of Illinois at Urbana-Champaign, USA  
Olaf Landsiedel, Chalmers University, Sweden

### Program Co-Chairs

Gian Pietro Picco, University of Trento, Italy  
Prashant Shenoy, University of Massachusetts, Amherst, USA

### Important Dates

Abstract Registration:	Oct. 10, 2018
Paper Submission:	Oct. 17, 2018
Author Rebuttal:	Dec. 10, 2018
Notification Date:	Jan. 15, 2019

<http://conferences.computer.org/iotdi/2019/>

iotdi 2019

4th ACM/IEEE Conference on Internet of Things Design and Implementation

CPS-IoT Week



DOI:10.1145/3271625

## What just happened in artificial intelligence and how it is being misunderstood.

BY ADNAN DARWICHE

# Human-Level Intelligence or Animal-Like Abilities?

“The vision systems of the eagle and the snake outperform everything that we can make in the laboratory, but snakes and eagles cannot build an eyeglass or a telescope or a microscope.”  
— Judea Pearl<sup>a</sup>

THE RECENT SUCCESSES of neural networks in applications like speech recognition, vision, and autonomous navigation has led to great excitement by members of the artificial intelligence (AI) community, as well as by the general public. Over a relatively short time, by the science clock, we managed to automate some tasks that have defied us for decades, using one of the more classical techniques due to AI research.

<sup>a</sup> Lecture by Judea Pearl, *The Mathematics of Causal Inference, with Reflections on Machine Learning and the Logic of Science*; <https://www.youtube.com/watch?v=zHjdd-W6o4>

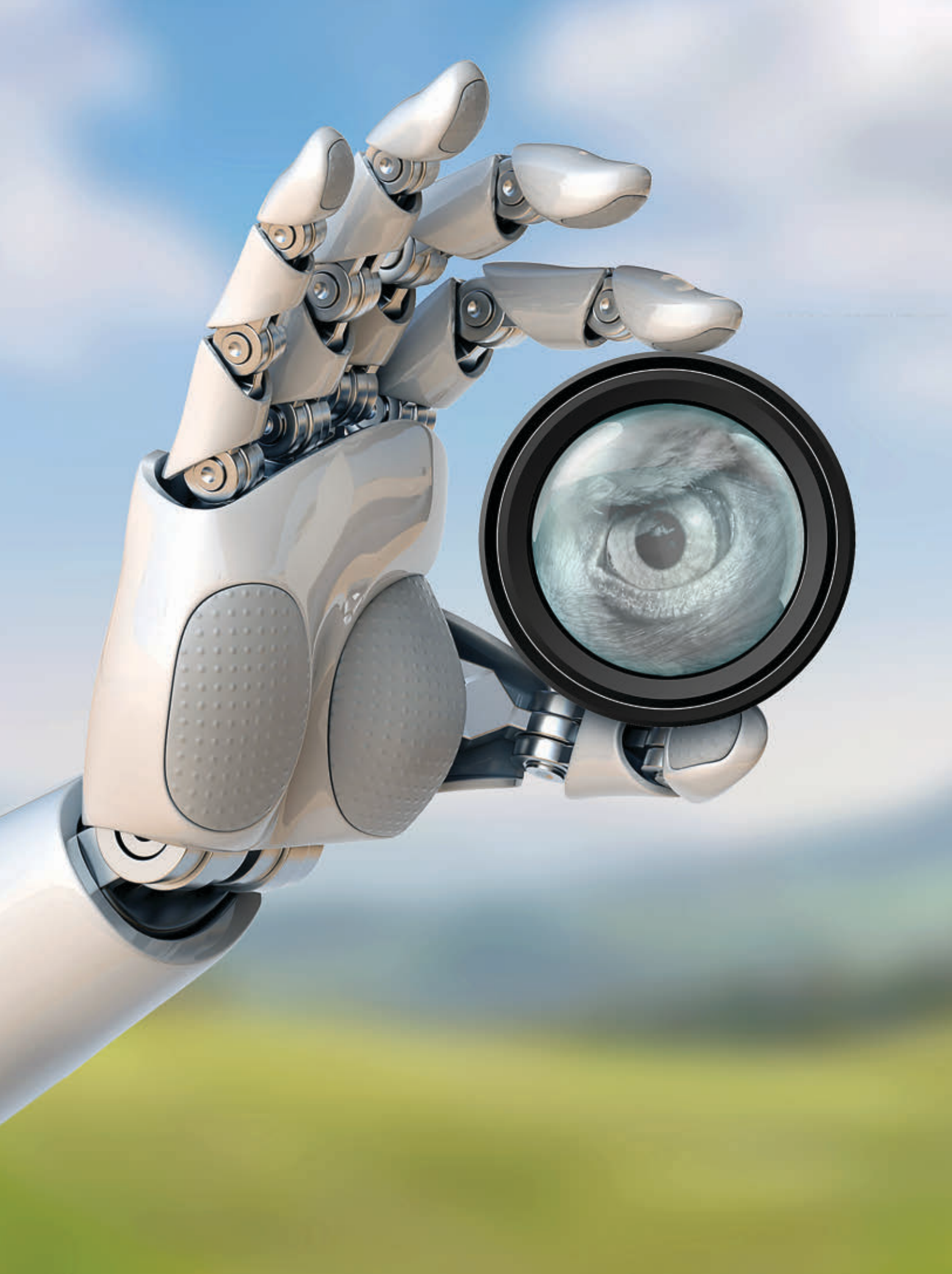
The triumph of these achievements has led some to describe the automation of these tasks as having reached human-level intelligence. This perception, originally hinted at in academic circles, has gained momentum more broadly and is leading to some implications. For example, some coverage of AI in public arenas, particularly comments made by several notable figures, has led to mixing this excitement with fear of what AI might bring us all in the future (doomsday scenarios).<sup>b</sup> Moreover, a trend is emerging in which machine learning research is being streamlined into neural network research, under its newly acquired label “deep learning.” This perception has also caused some to question the wisdom of continuing to invest in other machine learning approaches or even other mainstream areas of AI (such as knowledge representation, symbolic reasoning, and planning).

This turn of events in the history of AI has created a dilemma for researchers in the broader AI community. On the one hand, one cannot but be impressed with, and enjoy, what we have been able to accomplish with neural networks. On the other hand, mainstream scientific intuition stands in the way of accepting that a method

<sup>b</sup> Stephen Hawking said: “The development of full artificial intelligence could spell the end of the human race;” and Elon Musk said AI is: “... potentially more dangerous than nukes.”

### » key insights

- The recent successes of deep learning have revealed something very interesting about the structure of our world, yet this seems to be the least pursued and talked about topic today.
- In AI, the key question today is not whether we should use model-based or function-based approaches but how to integrate and fuse them so we can realize their collective benefits.
- We need a new generation of AI researchers who are well versed in and appreciate classical AI, machine learning, and computer science more broadly while also being informed about AI history.




that does not require explicit modeling or sophisticated reasoning is sufficient for reproducing human-level intelligence. This dilemma is further amplified by the observation that recent developments did not culminate in a clearly characterized and profound scientific discovery (such as a new theory of the mind) that would normally mandate massive updates to the AI curricula. Scholars from outside AI and computer science often sense this dilemma, as they complain they are not receiving an intellectually satisfying answer to the question: “What just happened in AI?”

The answer lies in a careful assessment of what we managed to achieve with deep learning and in identifying and appreciating the key scientific outcomes of recent developments in this area of research. This has unfortunately been lacking to a great extent. My aim here is to trigger such a discussion, encouraged by the positive and curious feedback I have been receiving on the thoughts expressed in this article.


### Background

To lay the ground for the discussion, I first mark two distinct approaches for tackling problems that have been of interest to AI. I call the first one “model-based” and the second “function-based.” Consider the object-recognition and -localization task in Figure 1. To solve it, the model-based approach requires one to represent knowledge about dogs and hats, among other things, and involves reasoning with such knowledge. The main tools of the approach today are logic and probability (mathematical modeling more generally) and can be thought of as the “represent-and-reason”<sup>c</sup> approach originally envisioned by the founders of AI. It is also the approach normally expected, at some level, by informed members of the scientific community. The function-based approach, on the other hand, formulates this task as a function-fitting problem, with function inputs coming directly from the image pixels and outputs corresponding to the high-level recognitions we seek. The function must have a form that can be evaluated efficiently so no

c This term might be likened to what has been called “good old-fashioned AI.”



**In my own quest to fully appreciate the progress enabled by deep learning, I came to the conclusion that recent developments tell us more about the problems tackled and the structure of our world than about neural networks per se.**



reasoning is required to compute the function outputs from its inputs. The main tool of this approach is the neural network. Many college students have exercised a version of it in a physics or chemistry lab, where they fit simple functions to data collected from various experiments, as in Figure 2. The main difference here is we are now employing functions with multiple inputs and outputs; the structure of these functions can be quite complex; and the problems being tackled are ones we tend to associate with perception or cognition, as opposed to, say, estimating the relationship between volume and pressure in a sealed container.<sup>d</sup>

The main observation in AI recently is that the function-based approach can be quite effective at certain AI tasks, more so than the model-based approach or at least earlier attempts at using this approach. This has surprised not only mainstream AI researchers, who mainly practice the model-based approach, but also machine learning researchers who practice various approaches, of which the function-based approach is but one.<sup>e</sup> This has had many implications, some positive and some giving grounds for concern.

On the positive side is the increasing number of tasks and applications now within reach, using a tool that can be very familiar to someone with only a broad engineering background, particularly one accustomed to estimating functions and using them to make predictions. What is of concern, however, is the current imbalance between exploiting, enjoying, and cheering this tool on the one hand and *thinking* about it on the other. This thinking is not only important for realizing the full potential of the tool but also for scientifically characterizing its potential

d This is also called the “curve-fitting” approach. While the term “curve” highlights the efficient evaluation of a function and captures the spirit of the function-based approach, it underplays the complex and rich structure of functions encoded by today’s (deep) neural networks, which can have millions if not billions of parameters.

e Machine learning includes the function-based approach but has a wide enough span that it overlaps with the model-based approach; for example, one can learn the parameters and structure of a model but may still need non-trivial reasoning to obtain answers from the learned model.



reach. The lack of such characterization is a culprit of current misconceptions about AI progress and where it may lead us in the future.

**What Just Happened in AI?**

In my own quest to fully appreciate the progress enabled by deep learning, I came to the conclusion that recent developments tell us more about the problems tackled and the structure of our world than about neural networks per se. These networks are parameterized functions that are expressive enough to capture any relationship between inputs and outputs and have a form that can be evaluated efficiently. This has been known for decades and described at length in textbooks. What caused the current turn of events?

To shed some light on this question, let me state again what we have discovered recently. That is, some seemingly complex abilities that are typically associated with perception or cognition can be captured and reproduced to a reasonable extent by simply fitting functions to data, without having to explicitly model the environment or symbolically reason about it. While this is a remarkable finding, it highlights problems and thresholds more than it highlights technology, a point I explain next.

Every behavior, intelligent or not, can be captured by a function that maps inputs (environmental sensing) to outputs (thoughts or actions). However, the size of this function can be quite large for certain tasks, assuming the function can be evaluated efficiently. In fact, the function may have an unbounded size in general, as it may have to map from life histories. The two key questions then are the following: For tasks of interest, are the corresponding functions simple enough to admit a compact representation that allows mapping inputs to outputs efficiently, as in neural networks (without the need for reasoning)? And, if the answer is yes, are we currently able to estimate these functions from input-output pairs (labeled data)?

What has happened in AI recently are three developments that bear directly on these questions: The first is our improved ability to fit functions to data, which has been enabled by the availability of massive amounts

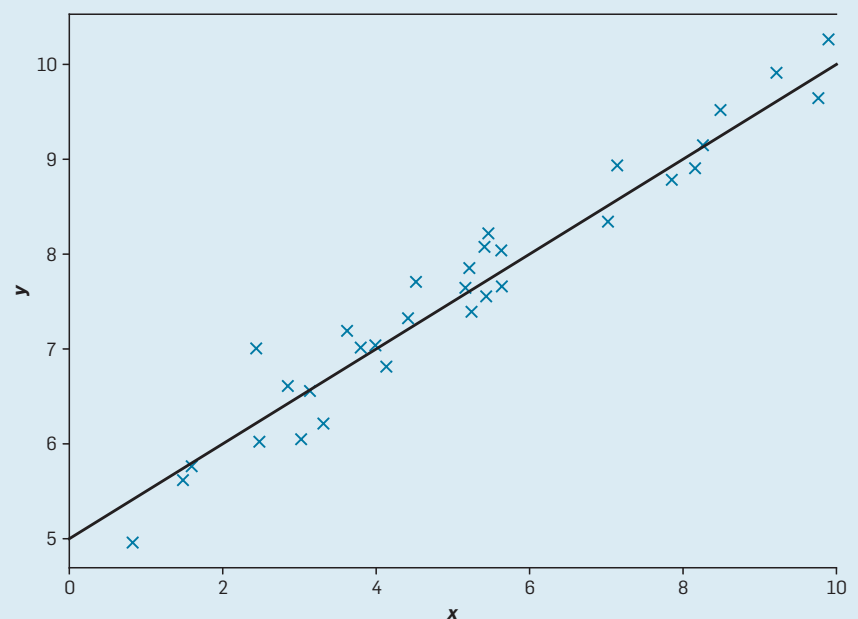
of labeled data; the increased computational power we now have at our hands; and the increasingly sophisticated statistical and optimization techniques for fitting functions (including new activation functions and new/deeper network structures). The second is that we have identified

a class of practical applications that correspond to functions that, we now know, are simple enough to allow compact representations that can be evaluated efficiently (again, without the need for reasoning), and whose estimation is within reach of current thresholds for gathering data, com-

**Figure 1. Object recognition and localization in an image (ImageNet).**



**Figure 2. Fitting a simple function to data.**



putational speed, and estimation techniques. This includes recognizing and localizing objects in some classes of images and certain tasks that pertain to natural language and speech. The third development, which goes largely unnoticed, is that we gradually changed our objectives and measures for success in ways that reduced the technical challenges considerably, at least as entertained by early AI researchers, while maintaining our ability to capitalize on the obtained results commercially, a point I discuss further later in the section on objectives and success.

Interestingly, none of these developments amounts to a major technical breakthrough in AI per se (such as the establishment of probability as a foundation of commonsense reasoning in the late 1980s and the introduction of neural networks more than 50 years ago).<sup>f</sup> Yet the combination of these factors created a milestone in AI history, as it had a profound impact on real-world applications and the successful deployment of various AI techniques that have been in the works for a very long time, particularly neural networks.<sup>g</sup>

### 'I Beg to Differ'

I shared these remarks in various contexts during the course of preparing this article. The audiences ranged from AI and computer science to law and public-policy researchers with an interest in AI. What I found striking is the great interest in this discussion and the com-

fort, if not general agreement, with the remarks I made. I did get a few "I beg to differ" responses though, all centering on recent advancements relating to optimizing functions, which are key to the successful training of neural networks (such as results on stochastic gradient descent, dropouts, and new activation functions). The objections stemmed from not having named them as breakthroughs (in AI). My answer: They all fall under the enabler I outlined earlier: "increasingly sophisticated statistical and optimization techniques for fitting functions." Follow up question: Does it matter that they are statistical and optimization techniques, as opposed to classical AI techniques? Answer: It does not matter as far as acknowledging and appreciating scientific inquiry and progress, but it does matter as far as explaining what just happened and, more important, forecasting what may happen next.

Consider an educated individual sitting next to you, the AI researcher, on a plane; I get that a lot. They figure out you do AI research and ask: What are the developments that enabled the current progress in AI? You recount the function-based story and lay out the three enablers. They will likely be impressed and also intellectually satisfied. However, if the answer is, "We just discovered a new theory of the mind," you will likely not be surprised if they also end up worrying about a Skynet coming soon to mess up our lives. Public perceptions about AI progress and its future are very important. The current misperceptions and associated fears are being nurtured by the absence of scientific, precise, and bold perspectives on what just happened, leaving much to the imagination.

This is not to suggest that only a new theory of the mind or an advance of such scale would justify some of the legitimate concerns surrounding AI. In fact, even limited AI technologies can lead to autonomous systems that may pose all kinds of risks. However, these concerns are not new to our industrialized society; recall safety concerns when the autopilot was introduced into the aerospace industry and job-loss concerns when ATMs were introduced into the banking industry. The headline here should therefore be "automation" more than "AI," as the latter is just a technology that happened to improve and

speed up automation.<sup>h</sup> To address these concerns, the focus should be shifted toward policy and regulatory considerations for dealing with the new level of automation our society is embarking on, instead of fearing AI.

### On Objectives and Success

Let me now address the third reason for the current turn of events, which relates to the change in objectives and how we measure success as a broad AI community. This reason is quite substantial yet goes largely unnoticed, especially by younger researchers. I am referring here to the gradual but sustained shift over AI history from trying to develop technologies that were meant to be intelligent and part of integrated AI systems to developing technologies that perform well and are integrated with consumer products; this distinction can be likened to what has been called "Strong AI" vs. "Weak AI."

This shift was paralleled by a sharpening of performance metrics and by progress against these metrics, particularly by deep learning, leading to an increased deployment of AI systems. However, these metrics and corresponding progress did not necessarily align with improving intelligence, or furthering our understanding of intelligence as sought by early AI researchers.<sup>i</sup> One must thus be careful not to draw certain conclusions based on current progress, which would be justified only if one were to make progress against earlier objectives. This caution particularly refers to current perceptions that we may have made considerable progress toward achieving "full AI."

Consider machine translation, which received significant attention in the early days of AI. The represent-and-reason approach aimed to comprehend text before translating it and is considered to have failed on this task, with function-based approaches being the state of the art today. In the early days of AI, success was measured by how far a system's accuracy was

<sup>f</sup> Research on neural networks has gone through many turns since their early traces in the 1940s. Nils Nilsson of Stanford University told me he does not think the pessimistic predictions of the 1969 book *Perceptrons: An Introduction to Computational Geometry* by Marvin Minsky and Seymour Papert was the real reason for the decline in neural network research back then, as is widely believed. Instead, it was the inability to train multiple layers of weights that Nilsson also wrestled with at SRI during that time "but couldn't get anywhere," as he explained to me.

<sup>g</sup> A perspective relayed to me by an anonymous reviewer is that science advances because instruments improve and that recent developments in neural networks could be viewed as improvements to our machine learning instruments. The analogy given here was to genomics and the development of high-throughput sequencing, which was not the result of a scientific breakthrough but rather of intense engineering efforts, yet such efforts have indeed revealed a vast amount about the human genome.

<sup>h</sup> See also the first report of the *One Hundred Year Study on Artificial Intelligence (AI100)* for a complementary perspective; <https://ai100.stanford.edu/>


<sup>i</sup> An anonymous reviewer said that throughout AI there are metrics for evaluating task performance but not for evaluating the fit among an agent, its goals, and its environment. Such global metrics may be needed to assess and improve the intelligence of AI systems.

from 100% compared to humans, and successful translation was predicated on the ability to comprehend text. Government intelligence was a main driving application; a failure to translate correctly can potentially lead to a political crisis. Today, the main application of machine translation is to webpages and social-media content, leading to a new mode of operation and a different measure of success. In the new context, there is no explicit need for a translation system to comprehend text, only to perform well based on the adopted metrics. From a consumer's viewpoint, success is effectively measured in terms of how far a system's accuracy is from 0%. If I am looking at a page written in French, a language I do not speak, I am happy with any translation that gives me a sense of what the page is saying. In fact, the machine-translation community rightfully calls this "gist translation." It can work impressively well on prototypical sentences that appear often in the data (such as in social media) but can fail badly on novel text (such as poetry). It is still very valuable yet corresponds to a task that is significantly different from what was tackled by early AI researchers. We did indeed make significant progress recently with function-based translation, thanks to deep learning. But this progress has not been directed toward the classical challenge of comprehending text, which aimed to acquire knowledge from text to enable reasoning about its content,<sup>j</sup> instead of just translating it.<sup>k</sup>


Similar observations can be made about speech-recognition systems.

<sup>j</sup> There are other views as to what "comprehension" might mean, as in, say, what might be revealed about language from the internal encodings of learned translation functions.

<sup>k</sup> With regard to the observation that the represent-and-reason approach is considered to have failed on machine translation, Stuart Russell of the University of California, Berkeley, pointed out to me that this is probably a correct description of an incorrect diagnosis, as not enough effort was directed toward pursuing an adequate represent-and-reason approach, particularly one that is trainable, since language has too many quirks to be captured by hand. This observation is part of a broader perspective I subscribe to calling for revisiting represent-and-reason approaches while augmenting them with advances in machine learning. This task would, however, require a new generation of researchers well versed in both approaches; see the section in this article on the power of success for hints as to what might stand in the way of having this breed of researchers.



**Some seemingly complex abilities that are typically associated with perception or cognition can be captured and reproduced to a reasonable extent by simply fitting functions to data.**



Perhaps one of the broadest applications of these systems today is in user interfaces (such as automated technical support and the commanding of software systems, as in phone and navigation systems in vehicles). These systems fail often; try to say something that is not very prototypical or not to hide your accent if you have one. But when these systems fail, they send the user back to a human operator or force the user to command the software through classical means; some users even adjust their speech to get the systems to work. Again, while the performance of these systems has improved, according to the adopted metrics, they are today embedded in new contexts and governed by new modes of operation that can tolerate lack of robustness or intelligence. Moreover, as in text, improving their performance against current metrics is not necessarily directed toward, nor requires addressing, the challenge of comprehending speech.<sup>l</sup>

Moving to vision applications, it has been noted that some object-recognition systems, based on neural networks, surpass human performance in recognizing certain objects in images. But reports also indicate how making simple changes to images may sometimes hinder the ability of neural networks to recognize objects correctly. Some transformations or deformations to objects in images, which preserve the human ability to recognize them, can also hinder the ability of networks to recognize them. While this does not measure up to the expectations of early AI researchers or even contemporary vision researchers, as far as robustness and intelligence is concerned, we still manage to benefit from these technologies in a number of applications. This includes recognizing faces during autofocus in smart cameras (people do not normally deform their faces but if they do, bad luck, an unfocused image); looking up images that contain cats in online search (it is ok if you end up getting a dog instead); and localizing surrounding vehicles in an image taken by

<sup>l</sup> An anonymous reviewer suggested that transcription is perhaps the main application of speech systems today, with substantial progress made toward the preferred metric of "word error rate." The same observation applies to this class of applications.



the camera of a self-driving car (the vulnerability of these systems to mistakes remains controversial in both its scope and how to deal with it at the policy and regulatory levels).

The significance of these observations stems from their bearing on our ability to forecast the future and decisions as to what research to invest in. In particular, does the success in addressing these selected tasks, which are driven by circumscribed commercial applications, justify the worry about doomsday scenarios? Does it justify claims that AI-based systems can now comprehend language or speech or do vision at the levels that humans do? Does it justify this current imbalance of attitudes toward various machine learning and AI approaches? If you work for a company that has an interest in such an application, then the answer is perhaps, and justifiably, yes. But, if you are concerned with scientific inquiry and understanding intelligence more broadly, then the answer is hopefully no.

In summary, what has just happened in AI is nothing close to a breakthrough that justifies worrying about doomsday scenarios. What just happened is the successful employment of AI technology in some widespread applications, aided greatly by developments in related fields, and by new modes of operation that can tolerate lack of robustness or intelligence. Put another way—and in response to headlines I see today, like “AI Has Arrived” and “I Didn’t See AI Coming”—AI has not yet arrived according to the early objective of capturing intelligent behavior. What really has arrived are numerous applications that can benefit from improved AI techniques that still fall short of AI ambitions but are good enough to be capitalized on commercially. This by itself is positive, until we confuse it with something else.

Let me close this section by stressing two points: The first is to reemphasize an earlier observation that while current AI technology is still quite limited, the impact it may have on automation, and hence society, may be substantial (such as in jobs and safety). This in turn calls for profound treatments at the technological, policy,



**We succeeded in these applications by having circumvented certain technical challenges instead of having solved them directly.**



and regulatory levels.<sup>m</sup> The second is that while function-based systems have been an enabling and positive development, we do need to be acutely aware of the reasons behind their success to better understand the implications. A key finding here is that some tasks in perception and cognition can be emulated to a reasonable extent without having to understand or formalize these tasks as originally believed and sought, as in some text, speech, and vision applications. That is, we succeeded in these applications by having circumvented certain technical challenges instead of having solved them directly.<sup>n</sup> This observation is not meant to discount current success but to highlight its nature and lay the grounds for this question: How far can we go with this direction? I revisit this issue later in the article.

#### **Human-Level or Animal-Level?**

Let me now get to the thoughts that triggered the title of this article in the first place. I believe human-level intelligence is not required for the tasks currently conquered by neural networks, as such tasks barely rise to the level of abilities possessed by many animals. Judea Pearl cited eagles and snakes as having vision systems that surpass what we can build today. Cats have navigation abilities that are far superior to any of those in existing automaton-navigation systems, including self-driving cars. Dogs can recognize and react to hu-

<sup>m</sup> Eric Horvitz of Microsoft Research brought up the idea of subjecting certain AI systems to trials as is done to approve drugs. The proper labeling of certain AI systems should also be considered, also as is done with drugs. For example, it has been suggested that the term “self-driving car” is perhaps responsible for the misuse of this AI-based technology by some drivers who expect more from the technology than is currently warranted.

<sup>n</sup> For example, one can now use learned functions to recognize cats in images without having to describe or model what a cat is, as originally thought and sought, by simply fitting a function based on labeled data of the form: (image, cat), (image, not cat). While this approach works better than modeling a cat (for now), it does not entail success in “learning” what a cat is, to the point where one can recognize, say, deformed images of cats or infer aspects of cats that are not relayed in the training dataset.

man speech, and African grey parrots can generate sounds that mimic human speech to impressive levels. Yet none of these animals has the cognitive abilities and intelligence typically attributed to humans.

One of the reactions I received to such remarks was: “I don’t know of any animal that can play Go!” This was in reference to the AlphaGo system, which set a milestone in 2016 by beating the world champion in the game. Indeed, we do not know of animals that can play a game as complex as Go. But first recall the difference between performance and intelligence: A calculator outperforms humans at arithmetic without possessing human or even animal cognitive abilities. Moreover, contrary to what seems to be widely believed, AlphaGo is not a neural network since its architecture is based on a collection of AI techniques that have been in the works for at least 50 years.<sup>o</sup> This includes the minimax technique for two-player games, stochastic search, learning from self-play, use of evaluation functions to cut off minimax search trees, and reinforcement learning, in addition to two neural networks. While a Go player can be viewed as a function that maps a board configuration (input) to an action (output), the AlphaGo player was not built by learning a single function from input-output pairs; only some of its components were built that way.<sup>p</sup> The issue here is not only about assigning credit but about whether a competitive Go function can be small enough to be represented and estimated under current data-gathering, storage, and computational thresholds. It would be quite interesting if this was the case, but we do not yet know the answer. I should also note that AlphaGo is a great example of what one can achieve today by integrating model-based and function-based approaches.

### Pushing Thresholds

One cannot of course preclude the possibility of constructing a competitive Go function or similarly complex

functions, even though we may not be there today, given current thresholds. But it begs the question: If it is a matter of thresholds, and given current successes, why not focus all our attention on moving thresholds further? While there is merit to this proposal, which seems to have been adopted by key industries, it does face challenges that stem from both academic and policy considerations. I address academic considerations next while leaving policy considerations to a later section.

From an academic viewpoint, the history of AI tells us to be quite cautious, as we have seen similar phenomena before. Those of us who have been around long enough can recall the era of expert systems in the 1980s. At that time, we discovered ways to build functions using rules that were devised through “knowledge engineering” sessions, as they were then called. The functions created through this process, called “expert systems” and “knowledge-based systems,” were claimed to achieve performance that surpassed human experts in some cases, particularly in medical diagnosis.<sup>q</sup> The term “knowledge is power” was used and symbolized a jubilant state of affairs, resembling what “deep learning” has come to symbolize today.<sup>r</sup> The period following this era came to be known as the “AI Winter,” as we could finally delimit the class of applications that yielded to such systems, and that class fell well short of AI ambitions.

While the current derivative for progress on neural networks has been impressive, it has not been sustained long enough to allow sufficient visibil-

ity into this consequential question: How effective will function-based approaches be when applied to new and broader applications than those already targeted, particularly those that mandate more stringent measures of success? The question has two parts: The first concerns the class of cognitive tasks whose corresponding functions are simple enough to allow compact representations that can be evaluated efficiently (as in neural networks) and whose estimation is within reach of current thresholds—or thresholds we expect to attain in, say, 10 to 20 years. The second alludes to the fact that these functions are only approximations of cognitive tasks; that is, they do not always get it right. How suitable or acceptable will such approximations be when targeting cognitive tasks that mandate measures of success that are tighter than those required by the currently targeted applications?

### The Power of Success

Before I comment on policy considerations, let me highlight a relevant phenomenon that recurs in the history of science, with AI no exception. I call it the “bullied-by-success” phenomenon, in reference to the subduing of a research community into mainly pursuing what is currently successful, at the expense of pursuing enough what may be more successful or needed in the future.

Going back to AI history, some of the perspectives promoted during the expert-systems era can be safely characterized today as having been scientifically absurd. Yet, due to the perceived success of expert systems then, these perspectives had a dominating effect on the course of scientific dialogue and direction, leading to a bullied-by-success community.<sup>s</sup> I saw a similar phenomenon during the transition from logic-based approaches to probability-based approaches for commonsense reasoning in the late 1980s. Popular arguments then, like “People don’t reason probabilistically,”

<sup>o</sup> Oren Etzioni of the Allen Institute for Artificial Intelligence laid out this argument during a talk at UCLA in March 2016 called *Myths and Facts about the Future of AI*.

<sup>p</sup> AlphaZero, the successor to AlphaGo, used one neural network instead of two and data generated through self-play, setting another milestone.

<sup>q</sup> One academic outcome of the expert system era was the introduction of a dedicated master’s degree at Stanford University called the “Master’s in AI” that was separate from the master’s in computer science and had significantly looser course requirements. It was a two-year program, with the second year dedicated to building an expert system. I was a member of the very last class that graduated from the program before it was terminated and recall that one of its justifications was that classical computer science techniques can be harmful to the “heuristic” thinking needed to effectively build expert systems.

<sup>r</sup> The phrase “knowledge is power” is apparently due to English philosopher Sir Francis Bacon (1561–1626).

<sup>s</sup> A colleague could not but joke that the broad machine learning community is being bullied today by the success of its deep learning sub-community, just as the broader AI community has been bullied by the success of its machine learning sub-community.

which I believe carries merit, were completely silenced when probabilistic approaches started solving commonsense reasoning problems that had defied logical approaches for more than a decade. The bullied-by-success community then made even more far-reaching choices in this case, as symbolic logic almost disappeared from the AI curricula. Departments that were viewed as world centers for representing and reasoning with symbolic logic barely offered any logic courses as a result. Now we are paying the price. As one example: Not realizing that probabilistic reasoning attributes numbers to Boolean propositions in the first place, and that logic was at the heart of probabilistic reasoning except in its simplest form, we have now come to the conclusion that we need to attribute probabilities to more complex Boolean propositions and even to first-order sentences. The resulting frameworks are referred to as “first-order probabilistic models” or “relational probabilistic models,” and there is a great need for skill in symbolic logic to advance these formalisms. The only problem is that this skill has almost vanished from within the AI community.

The blame for this phenomenon cannot be assigned to any particular party. It is natural for the successful to be overjoyed and sometimes also inflate that success. It is expected that industry will exploit such success in ways that may redefine the employment market and influence the academic interests of graduate students. It is also understandable that the rest of the academic community may play along for the sake of its survival: win a grant, get a paper in, attract a student. While each of these behaviors seems rational locally, their combination can be harmful to scientific inquiry and hence irrational globally. Beyond raising awareness about this recurring phenomenon, decision makers at the governmental and academic levels bear a particular responsibility for mitigating its negative effects. Senior members of the academic community also bear the responsibility of putting current developments in historical perspective, to empower junior researchers in pursuing their

genuine academic interests instead of just yielding to current fashions.<sup>t</sup>

### Policy Considerations

Let me now address some policy concerns with regard to focusing all our attention on functions instead of also on models. A major concern here relates to interpretability and explainability. If a medical-diagnosis system recommends surgery, we would need to know why. If a self-driving car kills someone, we would also need to know why. If a voice command unintentionally shuts down a power-generation system, it would need to be explained as well. Answering “Why?” questions is central to assigning blame and responsibility and lies at the heart of legal systems. It is also now recognized that opacity, or lack of explainability, is “one of the biggest obstacles to widespread adoption of artificial intelligence.”<sup>u</sup>

Models are more interpretable than functions.<sup>v</sup> Moreover, models offer a wider class of explanations than functions, including explanations of novel situations and explanations that can form a basis for “understanding” and “control.” This is due to models having access to in-

formation that goes beyond what can be extracted from data. To elaborate on these points, I first need to explain why a function may not qualify as a model, a question I received during a discussion on the subject.

Consider an engineered system that allows us to blow air into a balloon that then raises a lever that is positioned on top of the balloon. The input to this system is the amount of air we blow ( $X$ ), while the output is the position of the lever ( $Y$ ). We can learn a function that captures the behavior of the system by collecting  $X$ - $Y$  pairs and then estimating the function  $Y = f(X)$ . While this function may be all we need for certain applications, it would not qualify as a model, as it does not capture the system mechanism. Modeling that mechanism is essential for certain explanations (Why is the change in the lever position not a linear function of the amount of air blown?) and for causal reasoning more generally (What if the balloon is pinched?). One may try to address these issues by adding more inputs to the function but may also blow up the function size, among other difficulties; more on this next.

In his *The Book of Why: The New Science of Cause and Effect*, Judea Pearl explained further the differences between a (causal) model and a function, even though he did not use the term “function” explicitly. In Chapter 1, he wrote: “There is only one way a thinking entity (computer or human) can work out what would happen in multiple scenarios, including some that it has never experienced before. It must possess, consult, and manipulate a mental causal model of that reality.” He then gave an example of a navigation system based on either reasoning with a map (model) or consulting a GPS system that gives only a list of left-right turns for arriving at a destination (function). The rest of the discussion focused on what can be done with the model but not the function. Pearl’s argument particularly focused on how a model can handle novel scenarios (such as encountering roadblocks that invalidate the function recommendations) while pointing to the combinatorial impossibility of encoding such contingencies in the function, as it must have a bounded size.

<sup>t</sup> I made these remarks over a dinner table that included a young machine learning researcher, whose reaction was: “I feel much better now.” He was apparently subjected to this phenomenon by support-vector-machine (SVM) researchers during his Ph.D. work when SVMs were at their peak and considered “it” at the time. Another young vision researcher, pressed on whether deep learning is able to address the ambitions of vision research, said, “The reality is that you cannot publish a vision paper today in a top conference if it does not contain a deep learning component, which is kind of depressing.”

<sup>u</sup> See Castellanos, S. and Norton, S. Inside Darpa’s push to make artificial intelligence explain itself. *The Wall Street Journal* (Aug. 10, 2017); <http://on.wsj.com/2vmZKlM>; DARPA’s program on “explainable artificial intelligence”; <https://www.darpa.mil/program/explainable-artificial-intelligence>; and the E.U. general data protection regulation on “explainability”; <https://www.privacy-regulation.eu/en/r71.htm>


<sup>v</sup> I am referring here to learned and large functions of the kind that stand behind some of the current successes (such as neural networks with thousands or millions of parameters). This excludes simple or well-understood learned functions and functions synthesized from models, as they can be interpretable or explainable by design.




There is today growing work on explaining functions, where the vocabulary of explanations is restricted to the function inputs. For example, in medical diagnosis, an explanation may point to important inputs (such as age, weight, and heart attack history) when explaining why the function is recommending surgery. The function may have many more additional inputs, so the role of an explanation is to deem them irrelevant. In vision applications, such explanations may point to a specific part of the image that has led to recognizing an object; again, the role of an explanation is to deem some pixels irrelevant to the recognition. These explanations are practically useful, but due to their limited vocabulary and the limited information they can access, they could face challenges when encountering novel situations. Moreover, they may not be sufficient when one is seeking explanations for the purpose of understanding or control.

Consider a function that predicts the sound of an alarm based on many inputs, including fire. An input-based explanation may point to fire as a culprit of the alarm sound. Such an explanation relies effectively on comparing this scenario to similar scenarios in the data, in which the sound of the alarm was heard soon after fire was detected; these scenarios are summarized by the function parameters. While this may explain why the function reached a certain conclusion, it does not explain why the conclusion (alarm sound) may be true in the physical world.<sup>w</sup> Nor does it explain how fire triggers the alarm; is it, say, through smoke or through heat? The importance of these distinctions surfaces when novel situations arise that have not been seen before. For example, if the alarm is triggered by smoke, then inviting a smoker into our living room might trigger an alarm even in the absence of fire. In this case, pointing to fire as an explanation of the sound would be problematic. Humans arrive at such conclusions without ever seeing a smoker, which can also be achieved through reasoning on an appropriate

<sup>w</sup> The function imitates data instead of reasoning about a model of the physical world.



**Human-level intelligence is not required for the tasks currently conquered by neural networks, as such tasks barely rise to the level of abilities possessed by many animals.**



model. However, to do this based on a learned function, the function would need to be trained in the presence of smokers or other smoke-producing agents while defining smoke as an input to the function and assuring that smoke mediates the relationship between fire and alarm, a task that requires external manipulation.

As Pearl told me, model-based explanations are also important because they give us a sense of “understanding” or “being in control” of a phenomenon. For example, knowing that a certain diet prevents heart disease does not satisfy our desire for understanding unless we know why. Knowing that the diet works by lowering the cholesterol level in the blood partially satisfies this desire because it opens up new possibilities of control. For instance, it drives us to explore cholesterol-lowering drugs, which may be more effective than diet. Such control possibilities are implicit in models but cannot be inferred from a learned, black-box function, as it has no access to the necessary information (such as that cholesterol level mediates the relationship between diet and heart disease).

A number of researchers contacted me about the first draft of this section, which was focused entirely on explanations, to turn my attention to additional policy considerations that seem to require models. Like explanations, they all fell under the label “reasoning about AI systems” but this time to ensure that the developed systems would satisfy certain properties. At the top of these properties were safety and fairness, particularly as they relate to AI systems that are driven only by data. These considerations constitute further examples where models may be needed, not only to explain or compensate for the lack of enough data, but to further ensure we are able to build the right AI systems and reason about them rigorously.

### **A Theory of Cognitive Functions**

One reaction I received concerning my model-based vs. function-based perspective was during a workshop dedicated to deep learning at the Simons Institute for the Theory of Computing in March 2017. The workshop

title was “Representation Learning,” a term used with increasing frequency by deep learning researchers. If you have followed presentations on deep learning, you will notice that a critical component of getting these systems to work amounts to finding the correct architecture of the neural network. Moreover, the architectures vary depending on the task, and some of their components are sometimes portrayed as doing something that can be described at an intuitive level. For example, in language, one uses an encoder-decoder architecture in which the encoder transforms a sentence in the source language into an internal encoding, and the decoder then generates a sentence in the target language.

The reaction here was that deep learning is not learning a function (black box) but a representation since the architecture is not arbitrary but driven by the given task.<sup>x</sup> I see this differently. Architecting the structure of a neural network is “function engineering” not “representation learning,” particularly since the structure is penalized and rewarded by virtue of its conformity with input-output pairs. The outcome of function engineering amounts to restricting the class of functions that can be learned using parameter estimation techniques. This process is akin to restricting the class of distributions that can be learned after one fixes the topology of a probabilistic graphical model. The practice of representation learning is then an exercise in identifying the classes of functions that are suitable for certain tasks.<sup>y</sup>

In this context, I think what is needed most is a theory of cognitive functions. A cognitive function captures a relationship that is typically associated with cognition (such

**If I had my way, I would rename the field of deep learning as “learning approximations of cognitive functions.”**

as mapping audio signals to words and mapping words to some meaning). What is needed is a catalogue of cognitive functions and a study of their representational complexity—the size and nature of architectures needed to represent them—in addition to a study of their learnability and approximability. For Boolean functions, we have a deep theory of this kind. In particular, researchers have cataloged various functions in terms of the space needed to represent them in different forms (such as CNFs, DNFs, and OBDDs). What we need is something similar for real-valued functions that are meant to capture cognitive behaviors. In a sense, we already have some leads into such a theory; for example, researchers seem to know what architectures, or “function classes,” can be more effective for certain object-recognition tasks. This needs to be formalized and put on solid theoretical ground.<sup>z</sup> Such a theory would also include results on the learnability of function classes using estimation techniques employed by the deep learning community, particularly “gradient descent.” Interestingly, such results were presented at the Representation Learning workshop I referenced earlier in a talk called “Failures of Deep Learning” in which very simple functions were presented that defeat current estimation techniques. Even more interestingly, some have dismissed the importance of such results in side discussions on the grounds that the identified functions are not of practical significance; read “these are not cognitive functions” or “we have come a long way by learning approximations to functions.” In fact, if I had my way, I would rename the field of deep learning as “learning approximations of cognitive functions.”

The term “cognitive functions” surprised some colleagues who told me that “perception functions” may be more suitable, given that the current successes of deep learning have been

<sup>x</sup> There are other broader interpretations of the term “representation learning.”

<sup>y</sup> An anonymous reviewer suggested today’s practice of building deep neural networks can be viewed as the application of a new programming paradigm called “differentiable programming.” In this view, networks are carefully structured by a programmer using various differentiable program modules (such as convolutional layers, pooling layers, LSTM layers, residual blocks, and embedding layers). The compiler then differentiates and structures them for GPU execution. The key is to structure the program so the gradients are guided to do the right thing.

<sup>z</sup> The properties of learned functions may carry quite a bit of insight about the structure of our world; for example, linguists are called upon to study this phenomenon and unveil what learned translation functions may be revealing about the structure of language.

mostly in instinct-based perception (such as computer vision and language processing). I agree with this observation, except nothing at this stage prohibits functions from providing reasonable approximations to more high-level cognitive tasks. In fact, Go functions have been constructed using neural networks, even though they are not yet competitive with hybrid systems (such as AlphaGo). Admittedly, it is also possible that we might later realize that functions (of practical size) cannot provide reasonable approximations to a wide enough class of cognitive functions despite progress on pushing computational and data thresholds. The association with perception would then be more established in that case. Time will tell.

## Conclusion

This article was motivated by concerns I and others have had on how current progress in AI is being framed and perceived. Without a scholarly discussion of the causes and effects of recent achievements, and without a proper perspective on the obtained results, one stands to hinder further progress by perhaps misguiding the young generation of researchers or misallocating resources at the academic, industrial, and governmental levels. One also stands to misinform a public that has developed a keen interest in AI and its implications. The current negative discussions by the general public on the AI singularity, also called “super intelligence,” is partly due to the lack of accurate framings and characterizations of recent progress. With almost everyone being either overexcited or overwhelmed by the new developments, substantial scholarly discussions and reflections have gone missing.

I had the privilege of starting my research career in AI around the mid-to-late 1980s during one of the major crises in the field, a period marked by inability instead of ability. I was dismayed then, as I sat in classes at Stanford University, witnessing how AI researchers were being significantly challenged by some of the simpler tasks performed routinely by humans. I now realize how such crises can be enabling for scientific discovery, as they fuel academic thinking, empower researchers, and create grounds for


profound scientific contributions.<sup>aa</sup> On the other hand, I am reminded how times of achievements can potentially slow scientific progress by shifting academic interests, resources, and brain power too significantly toward exploiting what was just discovered, at the expense of understanding the discoveries and preparing for the moment when their practical applications have been delimited or exhausted.

There are many dimensions to such preparation. For the deep learning community, perhaps the most significant is a transition from the “look what else we can do” mode to a “look what else you can do” mode. This is not only an invitation to reach out to and empower the broader AI community; it is also a challenge since such a transition is not only a function of attitude but also an ability to characterize progress in ways that enable people from outside the community to understand and capitalize on it. The broader AI community is also both invited and challenged to identify fundamental ways in which functions can be turned into a boon for building and learning models. Given where we stand today, the question is not whether it is functions or models but how to profoundly integrate and fuse functions with models.<sup>ab</sup> This aim requires genuine cross-fertilization and the training of a new generation of researchers who are well-versed in and appreciative of various AI methods, and who are better informed about the history of AI.

I conclude with this reflection: I wrote the first draft of this article in November 2016. A number of colleagues provided positive feedback then, with one warning about a negative tone. I put the draft on hold for some months as a result while con-

tinuing to share its contents verbally in various contexts and revising accordingly. The decision to eventually release a first draft in July 2017 was triggered by two events: a discussion of these thoughts at a workshop organized by the UCLA School of Law and other discussions with colleagues outside of AI, including architecture, programming languages, networks, and theory. These discussions revealed a substantial interest in the subject and led me to conclude that the most important objective I should be seeking is “starting a discussion.” I may have erred in certain parts, I may have failed to give due credit, and I may have missed parts of the evolving scene. I just hope the thoughts I share here will start that discussion, and the collective wisdom of the community will correct what I may have gotten wrong.

## Acknowledgments

I benefited greatly from the feedback I received from anonymous reviewers and from colleagues who are too many to enumerate but whose input and discussions were critical to shaping the thoughts expressed here. However, I must specifically acknowledge Judea Pearl for inspiring the article and for helping with various arguments; Stuart Russell for providing very thoughtful and constructive feedback; Guy Van den Broeck for keeping me interested in the project every time I almost gave up; and Arthur Choi for being a generous and honest companion to the thinking that went into it. Finally, I wish to thank Nils Nilsson for telling me that he wished he had written the article and for kindly inviting me to share his feedback with others. This is an ultimate reward. 

<sup>aa</sup> Judea Pearl’s seminal work on probabilistic approaches to commonsense reasoning is one example outcome of the crisis.

<sup>ab</sup> An anonymous reviewer brought to my attention works on the analyses of human cognition, particularly Daniel Kahneman’s book *Thinking Fast and Slow*. The reviewer said “fast” naturally maps onto function-based and “slow” onto model-based, and there is a strong argument in the literature on cognitive science that people must at least combine them both. The reviewer further pointed out that there are a variety of cognitive architectures that embody specific hypotheses about such hybrids.

**Adnan Darwiche** (darwiche@cs.ucla.edu) is a professor in and chairman of the Computer Science Department at the University of California, Los Angeles, CA, USA.

Copyright held by author.



Watch the author discuss his work in this exclusive *Communications* video. <https://cacm.acm.org/videos/human-level-intelligence-or-animal-like-abilities>



DOI:10.1145/3230627

## Verified software secures the Unmanned Little Bird autonomous helicopter against mid-flight cyber attacks.

BY GERWIN KLEIN, JUNE ANDRONICK, MATTHEW FERNANDEZ, IHOR KUZ, TOBY MURRAY, AND GERNOT HEISER

# Formally Verified Software in the Real World

IN FEBRUARY 2017, a helicopter took off from a Boeing facility in Mesa, AZ, on a routine mission around nearby hills. It flew its course fully autonomously, and the safety pilot, required by the Federal Aviation Administration, did not touch any controls during the flight. This was not the first autonomous flight of the AH-6, dubbed the Unmanned Little Bird (ULB);<sup>3</sup> it had been doing them for years. This time, however, the aircraft was subjected to mid-flight cyber attacks. The central mission computer was attacked by rogue camera software, as well as by a virus delivered through a compromised USB stick that had been inserted during maintenance. The attack compromised some subsystems but could not affect the safe operation of the aircraft.

One might think surviving such an attack is not a big deal, certainly that military aircraft would be robust against cyber attacks. In reality, a “red team” of professional penetration testers hired by the Defense Advanced Research Projects Agency (DARPA) under its High-Assurance Cyber Military Systems (HACMS) program had in 2013 compromised the baseline version of the ULB, designed for safety rather than security, to the point where it could have crashed it or diverted to any location of its choice. In this light, risking an in-flight attack with a human on board indicates that something had changed dramatically.

This article explains that change and the technology that enabled it. Specifically, it is about technology developed under the HACMS program, aiming to ensure the safe operation of critical real-world systems in a hostile cyber environment—multiple autonomous vehicles in this case. The technology is based on formally verified software, or software with machine-checked mathematical proofs it behaves according to its specification. While this article is not about the formal methods themselves, it explains how the verified artifacts can be used to secure practical systems. The most impressive outcome of HACMS is arguably that the technology could be retrofitted onto existing real-world systems, dramatically improving their cyber resilience, a process called “seismic security retrofit” in analogy to, say, the seismic retrofit of buildings. Moreover, most of the re-engineering

### » key insights

- **Formal proof based on micro-kernel-enforced software architecture can scale to real systems at low cost.**
- **Mixed assurance levels and security levels within one system are possible and desirable; not all code has to be assured to the highest level.**
- **High assurance can be retrofitted to suitable existing systems with only moderate redesign and refactoring.**



**Boeing Little Bird in unmanned flight test.**

was done by Boeing engineers, not by formal verification researchers.

By far, not all the software on the HACMS vehicles was built on the basis of mathematical models and reasoning; the field of formal verification is not yet ready for such scale. However, HACMS demonstrated that significant improvement is feasible by applying formal techniques strategically to the most critical parts of the overall system. The HACMS approach works for systems in which the desired security property can be achieved through purely architecture-level enforcement. Its foundation is our verified microkernel, seL4, discussed later, which guarantees isolation between subsystems except for well-defined communication channels that are subject to the system's security policy. This isolation is leveraged by system-level component architectures that, through archi-

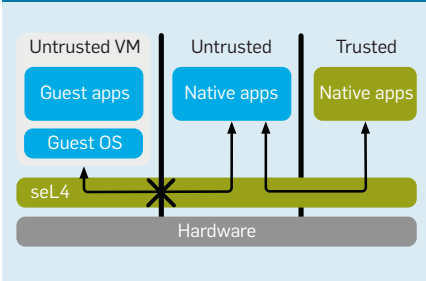
itecture, enforce the desired security property, and our verified component framework, CAMkES. The CAMkES framework integrates with architecture analysis tools from Rockwell Collins and the University of Minnesota, along with trusted high-assurance software components using domain-specific languages from Galois Inc.

The HACMS achievements are based on the software engineer's trusty old friend—modularization. What is new is that formal methods provide proof that interfaces are observed and module internals are encapsulated. This guaranteed enforcement of modularization allows engineers, like those at Boeing, who are not formal-method experts, to construct new or even retrofit existing systems, as discussed later, and achieve high resilience, even though the tools do not yet provide an overall proof of system security.

### **Formal Verification**

Mathematical correctness proofs of programs go back to at least the 1960s,<sup>14</sup> but for a long time, their real-world benefit to software development was limited in scale and depth. However, a number of impressive breakthroughs have been seen in recent years in the formal code-level verification of real-life systems, from the verified C compiler CompCert<sup>28</sup> to the verified seL4 microkernel,<sup>22,23,33</sup> verified conference system CoCon,<sup>21</sup> verified ML compiler CakeML,<sup>25</sup> verified interactive theorem provers Milawa,<sup>9</sup> and Candle,<sup>24</sup> verified crash-resistant file system FSCQ,<sup>5</sup> verified distributed system IronFleet,<sup>19</sup> and verified concurrent kernel framework CertiKOS,<sup>17</sup> as well as significant mathematical theorems, including the Four Colour Theorem,<sup>15</sup> mechanized proof of the Kepler Conjecture,<sup>18</sup> and Odd Order Theorem.<sup>16</sup> None of these

**Figure 1. Isolation and controlled communication with seL4.**



are toy systems. For instance, CompCert is a commercial product, the seL4 microkernel is used in aerospace, autonomous aviation, and as an Internet of Things platform, and the CoCon system has been used in multiple full-scale scientific conferences.

These verification projects required significant effort, and for verification to be practical for widespread use, the effort needs to decrease. Here, we demonstrate how strategically combining formal and informal techniques, partially automating the formal ones, and carefully architecting the software to maximize the benefits of isolated components, allowed us to dramatically increase the assurance of systems whose overall size and complexity is orders-of-magnitude greater than that of the systems mentioned earlier.

Note we primarily use formal verification to provide proofs about correctness of code that a system’s safety or security relies on. But it has other benefits as well. For example, code correctness proofs make assumptions about the context in which the code is run (such as behavior of hardware and configuration of software). Since formal verification makes these assumptions explicit, developer effort can focus on ensuring the assumptions

hold—through other means of verification like testing. Moreover, in many cases systems consist of a combination of verified and non-verified code, and in them, formal verification acts as a lens, focusing review, testing, and debugging on the system’s critical non-verified code.

**seL4**

We begin with the foundation for building provably trustworthy systems—the operating system (OS) kernel, the system’s most critical part and enabler of cost-effective trustworthiness of the entire system.

The seL4 microkernel provides a formally verified minimal set of mechanisms for implementing secure systems. Unlike standard separation kernels<sup>31</sup> they are purposefully general and so can be combined for implementing a range of security policies for a range of system requirements.

One of the main design goals of seL4 (see the sidebar “Proof Effort”) is to enforce strong isolation between mutually distrusting components that may run on top of it. The mechanisms support its use as a hypervisor to, say, host entire Linux operating systems while keeping them isolated from security-critical components that might run alongside, as outlined in Figure 1. In particular, this functionality allows system designers to deploy legacy components that may have latent vulnerabilities alongside highly trustworthy components.

The seL4 kernel is unique among general-purpose microkernels. Not only does it deliver the best performance in its class,<sup>20</sup> its 10,000 lines of C code have been subjected to more formal verification than any

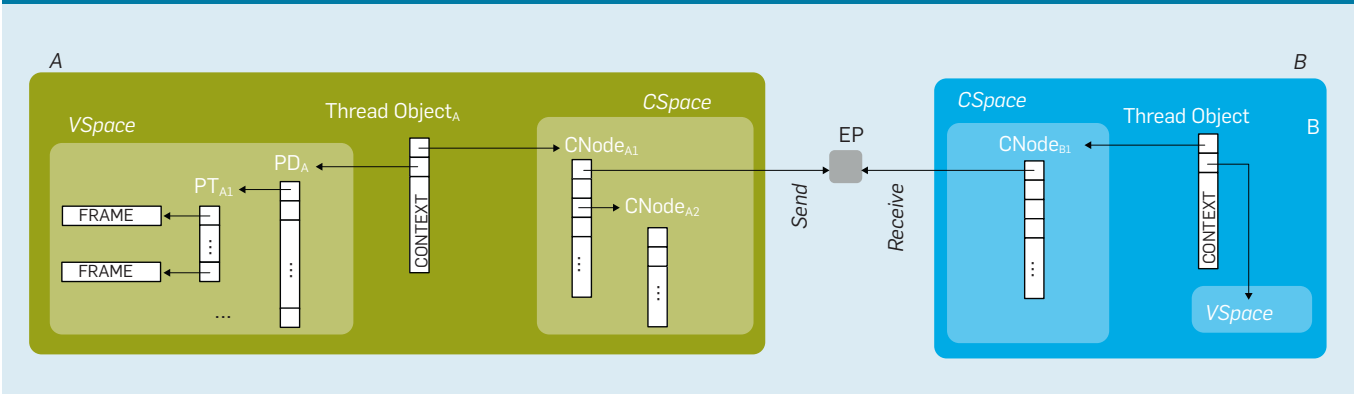
other publicly available software in human history in terms not only of lines of proof but strength of properties proved. At the heart of this verification story sits the proof of “functional correctness” of the kernel’s C implementation,<sup>23</sup> guaranteeing every behavior of the kernel is predicted by its formal abstract specification; see the online appendix (dl.acm.org/citation.cfm?doid=3230627&picked=formats) for an idea of how these proofs look. Following this guarantee, we added further proofs we explain after first introducing the main kernel mechanisms.

**seL4 API.** The seL4 kernel provides a minimal set of mechanisms for implementing secure systems: threads, capability management, virtual address spaces, inter-process communication (IPC), signaling, and interrupt delivery.

The kernel maintains its state in “kernel objects.” For example, for each thread in a system there is a “thread object” that stores information about scheduling, execution, and access control. User-space programs can refer to kernel objects only indirectly through “capabilities”<sup>10</sup> that combine a reference to an object with a set of access rights to this object. For example, a thread cannot start or stop another thread unless it has a capability to the corresponding thread object.

Threads communicate and synchronize by sending messages through IPC “endpoint” objects. One thread with a send capability to an appropriate endpoint can message another thread that has a receive capability to that endpoint. “Notification” objects provide synchronization through sets of binary semaphores. Virtual address translation is managed by kernel objects that represent page directories,

**Figure 2. Kernel objects for an example seL4-based system with two threads communicating via an endpoint.**





page tables, and frame objects, or thin abstractions over the corresponding entities of the processor architecture. Each thread has a designated “VSpace” capability that points to the root of the thread’s address-translation object tree. Capabilities themselves are managed by the kernel and stored in kernel objects called “CNodes” arranged in a graph structure that maps object references to access rights, analogous to page tables mapping virtual to physical addresses. Each thread has a distinguished capability identifying a root CNode. We call the set of capabilities reachable from this root the thread’s “CSpace.” Capabilities can be transmitted over endpoints with the grant operation and can be shared via shared CSpaces. Figure 2 outlines these kernel objects on an example.

**Security proofs.** With its generality, seL4’s kernel API is necessarily low-level and admits highly dynamic system architectures. Direct reasoning about this API can thus be a challenge.

The higher-level concept of access control policies abstracts away from individual kernel objects and capabilities, capturing instead the access-control configuration of a system via a set of abstract “subjects” (think components) and the authorities each has over the others (such as to read data and send a message). In the example in Figure 2, the system would have components *A* and *B* with authority over the endpoint.

Sewell et al.<sup>36</sup> proved for such suitable access control policies that seL4 enforces two main security properties: authority confinement and integrity.

Authority confinement states that the access control policy is a static (unchanging) safe approximation of the concrete capabilities and kernel objects in the system for any future state of execution. This property implies that no matter how the system develops, no component will ever gain more authority than the access control policy predicts. In Figure 2, the policy for component *B* does not contain write access to component *A*, and *B* will thus never be able to gain this access in the future. The property thus implies that reasoning at the policy level is a safe approximation over reasoning about the concrete access-control state of the system.

Integrity states that no matter what a component does, it will never be able

## Proof Effort

seL4 design and code development took two person-years. Adding up all seL4-specific proofs over the years comes to a total of 18 person-years for 8,700 lines of C code. In comparison, L4Ka::Pistachio, another microkernel in the L4 family, comparable in size to seL4, took six person-years to develop and provides no significant level of assurance. This means there is only a factor 3.3 between verified software and traditionally engineered software. According to the estimation method by Colbert and Boehm,<sup>8</sup> a traditional Common Criteria EAL7 certification for 8,700 lines of C code would take more than 45.9 person-years. That means formal binary-level implementation verification is already more than a factor of 2.3 less costly than the highest certification level of Common Criteria yet provides significantly stronger assurance.

In comparison, the HACMS approach described here uses only these existing proofs for each new system, including the proofs generated from tools. The overall proof effort for a system that fits this approach is thus reduced to person-weeks instead of years, and testing can be significantly reduced to only validating proof assumptions.

to modify data in the system (including by any system calls it might perform) the access control policy does not explicitly allow it to modify. For instance, in Figure 2, the only authority component *A* has over another component is the send right to the endpoint from which component *B* receives. This means the maximum state change *A* can effect in the system is in *A* itself and in *B*’s thread state and message buffer. It cannot modify any other parts of the system.

The dual of integrity is confidentiality, which states that a component cannot read another component’s data without permission,<sup>29</sup> proved the stronger property of intransitive non-interference for seL4; that is, given a suitably configured system (with stronger restrictions than for integrity), no component is able to learn information about another component or its execution without explicit permission. The proof expresses this property in terms of an information-flow policy that can be extracted from the access-control policy used in the integrity proof. Information will flow only when explicitly allowed by the policy. The proof covers explicit information flows, as well as potential in-kernel covert storage channels, but timing channels are outside its scope and must be addressed through different means.<sup>6</sup>

Further proofs about seL4 include the extension of functional correctness, and thus the security theorems, to the binary level for the ARMv7 architecture<sup>35</sup> and a sound worst-case execution time profile for the kernel<sup>2,34</sup> necessary for real-time systems. The seL4 kernel is available for multiple ar-

chitectures—ARMv6, ARMv7, ARMv7a, ARMv8, RISC-V, Intel x86, and Intel x64—and its machine-checked proof<sup>33</sup> is current on the ARMv7 architecture for the whole verification stack, as well as on ARMv7a with hypervisor extensions for functional correctness.

### Security by Architecture

The previous section summarized the seL4 kernel software engineers can use as a strong foundation for provably trustworthy systems. The kernel forms the bottom layer of the trusted computing base (TCB) of such systems. The TCB is the part of the software that needs to work correctly for the security property of interest to hold. Real systems have a much larger TCB than just the microkernel they run on, and more of the software stack would need to be formally verified to gain the same level of assurance as for the kernel. However, there are classes of systems for which this is not necessary, for which the kernel-level isolation theorems are already enough to enforce specific system-level security properties. This section includes an example of such a system.

The systems for which this works are those in which component architectures alone already enforce the critical property, potentially together with a few small, trusted components. Our example is the mission-control software of a quadcopter that was the research-demonstration vehicle in the HACMS program mentioned earlier.

Figure 3 outlines the quadcopter’s main hardware components. It is intentionally more complex than needed for a quadcopter, as it is meant to be

representative of the ULB, and is, at this level of abstraction, the same as the ULB architecture.

The figure includes two main computers: a mission computer that communicates with the ground-control station and manages mission-payload software (such as for controlling a camera); and a flight computer with the task of flying the vehicle, reading sensor data, and controlling motors. The computers communicate via an internal network, a controller area network, or CAN bus, on the quadcopter, a dedicated Ethernet on the ULB. On the quadcopter, the mission computer also has an insecure WiFi link, giving us the opportunity to demonstrate further security techniques.

The subsystem under consideration in this example is the mission computer. Four main properties must be enforced: only correctly authenticated commands from the ground station are sent to the flight computer; cryptographic keys are not leaked; no additional messages are sent to the flight computer; and untrusted payload software cannot influence the ve-

hicle's flight behavior. The operating assumption is that the camera is untrusted and potentially compromised, or malicious, that its drivers and the legacy payload software are potentially compromised, and any outside communication is likewise potentially compromised. For the purpose of this example, we assume a correct and strong cryptography implementation, or the key cannot be guessed, and that basic radio jamming and denial-of-service by overwhelming the ground station radio link are out of scope.

Figure 4 outlines how we design the quadcopter architecture to achieve these properties. We use a virtual machine (VM) running Linux as a containment vessel for legacy payload software, camera drivers, and WiFi link. We isolate the cryptography control module in its own component, with connections to the CAN bus component, to the ground station link, and to the Linux VM for sending image-recognition data back to the ground station. The purpose of the crypto component is to forward (only) authorized messages to the flight computer via the CAN interface stack and send back diagnostic data to the ground station. The radio-link component sends and receives raw messages that are encrypted, decrypted, and authenticated, respectively, by the crypto component.

Establishing the desired system properties is now reduced purely to the isolation properties and information-flow behavior of the architecture, and to the behavior of the single trusted crypto component. Assuming correct

behavior of that component, keys cannot be leaked, as no other component has access to them; the link between Linux and the crypto component in Figure 4 is for message passing only and does not give access to memory. Only authenticated messages can reach the CAN bus, as the crypto component is the only connection to the driver. Untrusted payload software and WiFi are, as part of the Linux VM, encapsulated by component isolation and can communicate to the rest of the system only via the trusted crypto component.

It is easy to imagine that this kind of architecture analysis could be automated to a high degree through model checking and higher-level mechanized reasoning tools. As observed in MILS systems,<sup>1</sup> component boundaries in an architecture are not just a convenient decomposition tool for modularity and code management but, with enforced isolation, provide effective boundaries for formal reasoning about the behavior of the system. However, the entire argument hinges on the fact that component boundaries in the architecture are correctly enforced at runtime in the final, binary implementation of the system.

The mechanisms of the seL4 kernel discussed earlier can achieve this enforcement, but the level of abstraction of the mechanisms is in stark contrast to the boxes and arrows of an architecture diagram; even the more abstract access-control policy still contains far more detail than the architecture diagram. A running system of this size contains tens of thousands of kernel objects and capabilities that are created programmatically, and errors in configuration could lead to security violations. We next discuss how we not only automate the configuration and construction of such code but also how we can automatically prove that architecture boundaries are enforced.

**Verified Componentization**

The same way reasoning about security becomes easier with the formal abstractions of security policies, abstraction also helps in building systems. The CAMkES component platform,<sup>27</sup> which runs on seL4 abstracts over the low-level kernel mechanisms, provides communication primitives, as well as support for decomposing a system into

Figure 3. Autonomous-air-vehicle architecture.

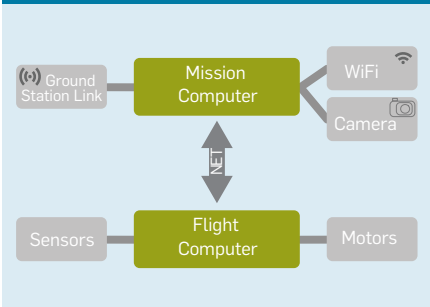
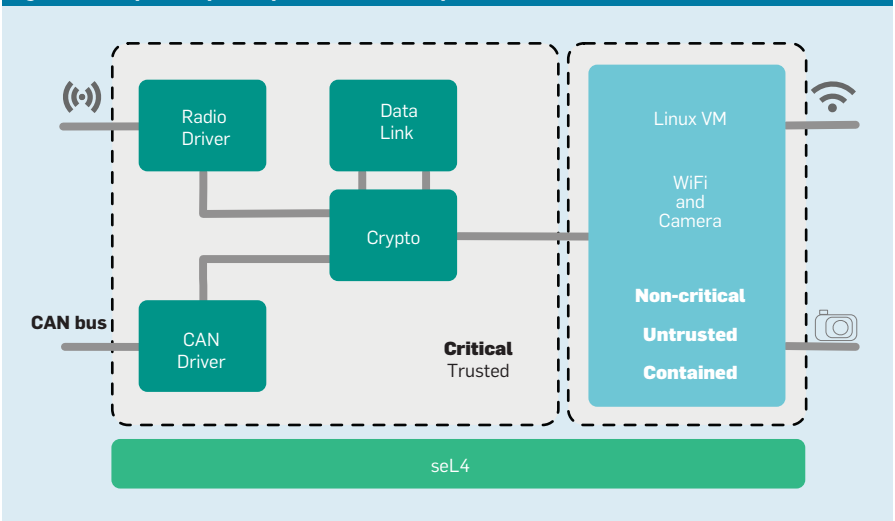


Figure 4. Simplified quadcopter mission-computer architecture.



functional units, as in Figure 5. Using this platform, systems architects can design and build seL4-based systems in terms of high-level components that communicate with each other and with hardware devices through connectors like remote procedure calls (RPCs), dataports, and events.

**Generated code.** Internally, CAMkES implements these abstractions using seL4’s low-level kernel objects. Each component comprises (at least) one thread, a CSpace, and a VSpace. RPC connectors use endpoint objects, and CAMkES generates glue code to marshal and unmarshal messages and send them over IPC endpoints. Likewise, a dataport connector is implemented through shared memory, shared frame objects present in the address spaces of two components, and optionally restricting the direction of communication. Finally, an event connector is implemented using seL4’s notification mechanism.

CAMkES also generates, in the capDL language,<sup>26</sup> a low-level specification of the system’s initial configuration of kernel objects and capabilities. This capDL specification is the input for the generic seL4 initializer that runs as the first task after boot and performs the necessary seL4 operations to instantiate and initialize the system.<sup>4</sup>

In summary, a component platform provides free code. The component architecture describes a set of boxes and arrows, and the implementation task is reduced to simply filling in the boxes; the platform generates the rest while enforcing the architecture.

With a traditional component platform, the enforcement process would mean the generated code increases the trusted computing base of the system, as it has the ability to influence the functionality of the components. However, CAMkES also generates proofs.

**Automated proofs.** While generating glue code, CAMkES produces formal proofs in Isabelle/HOL, following a translation-validation approach,<sup>30</sup> demonstrating that the generated glue code obeys a high-level specification and the generated capDL specification is a correct refinement of the CAMkES description.<sup>12</sup> We have also proved that the generic seL4 initializer correctly sets up the system in the desired initial configuration. In doing so, we au-

Figure 5. CAMkES workflow.

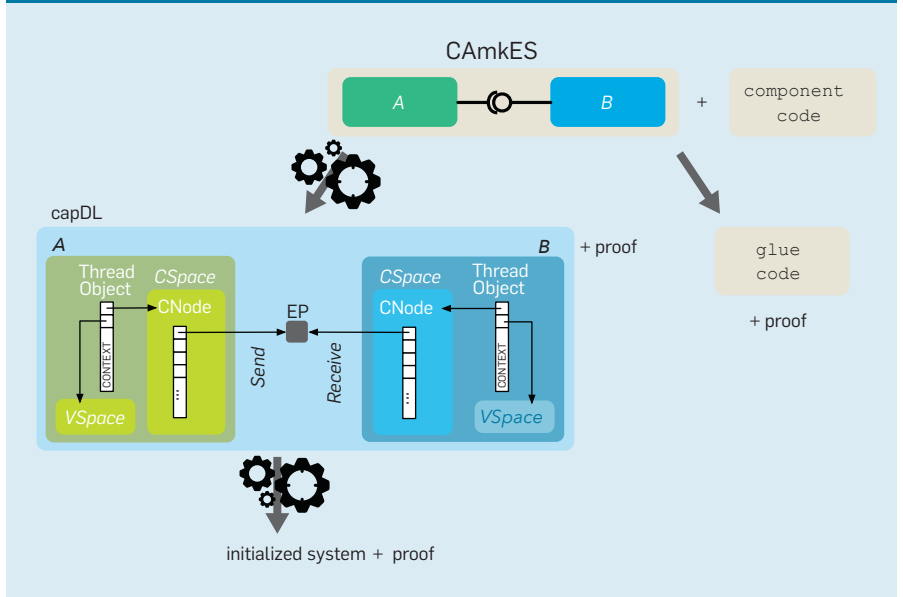
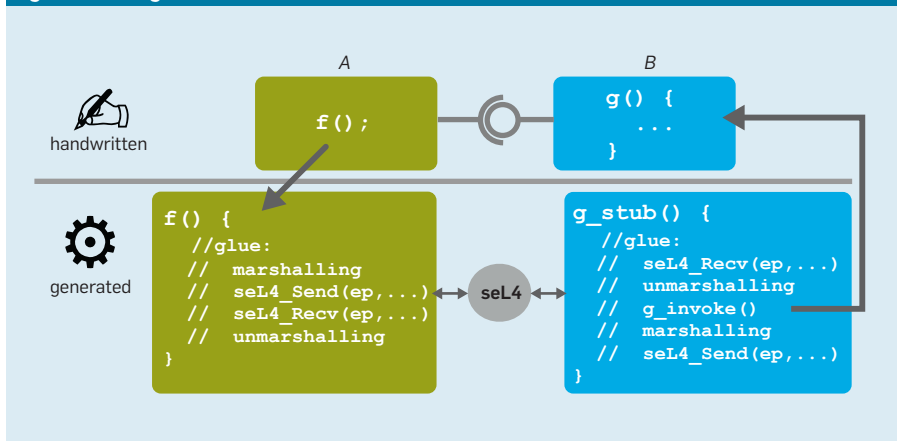


Figure 6. RPC-generated code.



tomate large parts of system construction without expanding the trusted computing base.

Developers rarely look at the output of code generators, focusing instead on the functionality and business logic of their systems. In the same way, we intend the glue code proofs to be artifacts that do not need to be examined, meaning developers can focus on proving the correctness of their handwritten code. Mirroring the way a header generated by CAMkES gives the developer an API for the generated code, the top-level generated lemma statements produce a proof API. The lemmas describe the expected behavior of the connectors. In the example of RPC glue code outlined in Figure 6, the generated function  $f$  provides a way to invoke a remote function  $g$  in another component. To preserve the abstraction, calling  $f$  must be equiva-

lent to calling  $g$ . The lemma the system generates ensures the invocation of the generated RPC glue code  $f$  behaves as a direct invocation of  $g$ , as if it were co-located with the caller.

To be useful, the proofs the system generates must be composable with (almost) arbitrary user-provided proofs, both of the function  $g$  and of the contexts where  $g$  and  $f$  are used. To enable this composable, the specification of the connectors is parameterized through user-provided specifications of remote functions. In this way, proof engineers can reason about their architecture, providing specifications and proofs for their components, and rely on specifications for the generated code.

To date, we have demonstrated this process end-to-end using a specific CAMkES RPC connector.<sup>12,13</sup> Extending the proof generator to support other



connectors, allowing construction of more diverse verified systems, should be simpler to achieve, because other connector patterns (data ports and events) are significantly less complex than RPC.

Next to communication code, CAMkES produces the initial access control configuration that is designed to enforce architecture boundaries. To prove the two system descriptions—capDL and CAMkES—correspond, we consider the CAMkES description as an abstraction of the capDL description. We use the established framework<sup>36</sup> mentioned earlier to infer authority of one object over another object from a capDL description to lift reasoning to a policy level. Additionally, we have defined rules for inferring authority between components in a CAMkES description. The produced proof ensures the capDL objects, when represented as an authority graph with objects grouped per component, have the same intergroup edges as the equivalent graph between CAMkES components.<sup>12</sup> Intuitively, this correspondence between the edges means an architecture analysis of the policy inferred by the CAMkES description will hold for the policy inferred by the generated capDL description, which in turn is proved to satisfy authority confinement, integrity, and confidentiality, as mentioned earlier.

Finally, to prove correct initialization, CAMkES leverages the generic initializer that will run as the first user task following boot time. In seL4, this first (and unique) user task has access to all available memory, using it to create objects and capabilities according to the detailed capDL description it takes as input. We proved that the state following execution of the initial-

izer satisfies the one described in the given specification.<sup>4</sup> This proof holds for a precise model of the initializer but not yet at the implementation level. Compared to the depth of the rest of the proof chain, this limitation may appear weak, but it is already more formal proof than would be required for the highest level (EAL7) of a Common Criteria security evaluation.

**Seismic Security Retrofit**

In practice, there are few opportunities to engineer a system from scratch for security, so the ability to retrofit for security is crucial for engineering secure systems. Our seL4-based framework supports an iterative process we call “seismic security retrofit,” as a regular structural architect might retrofit an existing building for greater resilience against earthquakes. We illustrate the process by walking through an example that incrementally adapts the existing software architecture of an autonomous air vehicle, moving it from a traditional testing approach to a high-assurance system with theorems backed by formal methods. While this example is based on work done for a real vehicle—the ULB—it is simplified for presentation and does not include all details.

The original vehicle architecture is the same as the architecture outlined in Figure 3. Its functionality is split over two separate computers: a flight computer that controls the actual flying and the mission computer that performs high-level tasks (such as ground-station communication and camera-based navigation). The original version of the mission computer was a monolithic software application running on Linux. The rest of the example concentrates on a retrofit of this

mission-computer functionality. The system was built and re-engineered by Boeing engineers, using the methods, tools, and components provided by the HACMS partners.

**Step 1. Virtualization.** The first step was to take the system as is and run it in a VM on top of a secure hypervisor (see Figure 7). In the seismic-retrofit metaphor, doing so corresponds to situating the system on a more flexible foundation. A VM on top of seL4 in this system consists of one CAMkES component that includes a virtual machine monitor (VMM) and the guest operating system, in this case Linux. The kernel provides abstractions of the virtualization hardware, while the VMM manages these abstractions for the VM. The seL4 kernel constrains not only the guest but also the VMM, so the VMM implementation does not need to be trusted to enforce isolation. Failure of the VMM will lead to failure of the guest but not to failure of the complete system.

Depending on system configuration, the VM may have access to hardware devices through para-virtualized drivers, pass-through drivers, or both. In the case of pass-through drivers, developers can make use of a system MMU or IOMMU to prevent hardware devices and drivers in the guest from breaching isolation boundaries. Note that simply running a system in a VM adds no additional security or reliability benefits. Instead, the reason for this first step is to enable step 2.

**Step 2. Multiple VMs.** The second step in a seismic retrofit strengthens existing walls. In software, the developer can improve security and reliability by splitting the original system into multiple subsystem partitions, each consisting of a VM running the

Figure 7. All functionality in a single VM.

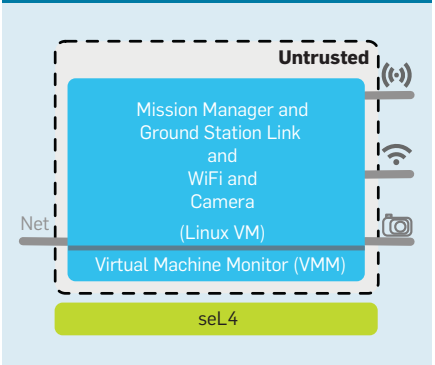


Figure 8. Functionality split into multiple VMs.

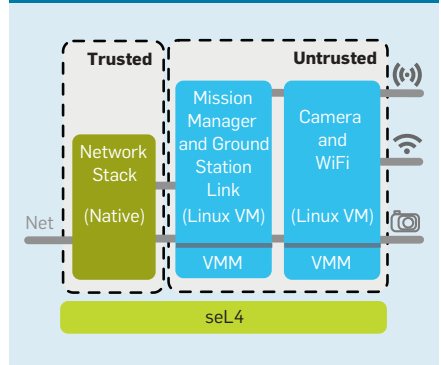
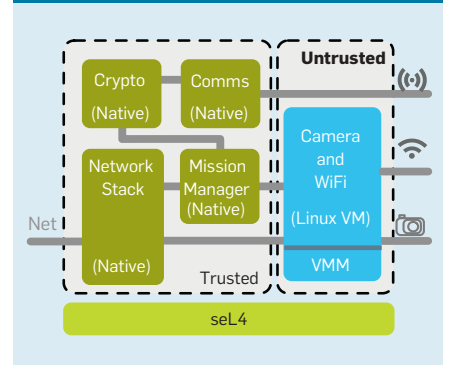


Figure 9. Functionality in native components.



code of only part of the original system. Each VM/VMM combination runs in a separate CAMkES component that introduces isolation between the different subsystems, keeping mutually distrusting ones from affecting each other, and, later, allowing different assurance levels to coexist.

In general, the partitions follow the existing software architecture, although a redesign may be necessary where the software architecture is inadequate for effective isolation.

The partitions will in general need to communicate with each other, so in this step we also add appropriate communication channels between them. For security, it is critically important that these interfaces are narrow, limiting the communication between partitions to only what is absolutely necessary to maximize the benefits of isolation. Moreover, interface protocols should be efficient, keeping the required number of messages or amount of data copying minimal. Critically, seL4's ability to enable controlled and limited sharing of memory between partitions allows a developer to minimize the amount of data copying.

Besides the VMs that represent subsystems of the original system, we also extract and implement components for any shared resources (such as the network interface).


We can iterate the entire step 2 until we have achieved the desired granularity of partitions. The right granularity is a trade-off between the strength of isolation on the one hand and the increased overhead and cost of communication between partitions, as well as re-engineering cost, on the other.

In our example we end up with three partitions: a VM that implements the ground-station communication functionality running on Linux; another VM that implements camera-based navigation functionality (also running on Linux); and a native component that provides shared access to the network, as in Figure 8.

**Step 3. Native components.** Once the system has been decomposed into separate VM partitions, some or all of the individual partitions can be reimplemented as native components rather than as VMs. The aim is to significantly reduce the attack surface for the same functionality. An additional



**We intend the glue code proofs to be artifacts that do not need to be examined, meaning developers can focus on proving the correctness of their handwritten code.**



benefit of transforming a component into native code is a reduced footprint and better performance, removing the guest operating system and removing the execution and communication overhead of the VMM.

Using a native component also increases the potential for applying formal verification and other techniques for improving the assurance and trustworthiness of the component. Examples range from full functional verification of handwritten code to cogeneration of code and proofs, application of model checking, using type-safe programming languages, and static analysis or traditional thorough testing of a smaller codebase.

Due to the isolation provided by seL4 and the componentized architecture, it becomes possible for components of mixed assurance levels to coexist in the system without decreasing the overall assurance to that of the lowest-assurance component or increasing the verification burden of the lowest-assurance components to that of the highest-assurance ones.

In our example, we target the VM for mission manager and ground-station link, implementing the communications, cryptography, and mission-manager functionality as native components. We leave the camera and WiFi to run in a VM as an untrusted legacy component (see Figure 9). This split was a trade-off between the effort to reimplement the subsystems and the benefit gained by making them native from both a performance and an assurance perspective.

**Step 4. Overall assurance.** With all parts in place, the final step is to analyze the assurance of the overall system based on the assurance provided by the architecture and by individual components.


In HACMS, the communication, cryptography, and mission manager functionality were implemented in a provably type-safe, domain-specific language called Ivory,<sup>11</sup> with fixed heap-memory allocation. Without further verification, Ivory does not give us high assurance of functional correctness but does give us assurance about robustness and crash-safety. Given component isolation, we reason that these assurances are preserved in the presence of untrusted components (such as the camera VM).

The networking component is implemented in standard C code consisting of custom code for the platform and pre-existing library code. Its assurance level corresponds to that obtained through careful implementation of known code. Robustness could be increased without much cost through such techniques as driver synthesis<sup>32</sup> and type-safe languages, as with Ivory. However, in the overall security analysis of the system, any compromise of the network component would be able to inject or modify only network packets. Since the traffic is encrypted, such an attack would not compromise the guarantee that only authorized commands reach the flight computer.


The camera VM is the weakest part of the system, since it runs a stock Linux system and is expected to have vulnerabilities. However, as the VM is isolated, if attackers were to compromise the VM, they would not be able to escape to other components. The worst an attacker could do is send incorrect data to the mission-manager component. As in the quadcopter, the mission manager validates data it receives from the camera VM. This is the part of the system on the ULB that demonstrated containment of a compromise in the in-flight attack mentioned at the beginning of the article. This was a white-box attack, where the Red Team had access to all code and documentation, as well as to all external communication, and was intentionally given root access to the camera VM, simulating a successful attack against legacy software. Successfully containing the attack and being able to defend against this very powerful Red Team scenario served to validate the strength of our security claims and uncover any missed assumptions, interface issues, or other security issues the research team might have failed to recognize.

### Limitations and Future Work

This article has given an overview of a method for achieving very high levels of assurance for systems in which security property can be enforced through their component architecture. We have proved theorems for the kernel level and its correct configuration, as well as theorems that ensure the component platform correctly configures protec-



**The camera VM is the weakest part of the system, since it runs a stock Linux system and is expected to have vulnerabilities.**



tion boundaries according to its architecture description, and that it produces correct RPC communication code. The connection with a high-level security analysis of the system remains informal, and the communication code theorems do not cover all communication primitives the platform provides. While more work would be required to automatically arrive at an end-to-end system-level theorem, it is clear at this stage that one is feasible.

The main aim of the reported work is to dramatically reduce verification effort for specific system classes. While the purely architecture-based approach described here can be driven a good deal further than in the ULB example, it is clearly limited by the fact it can express only properties that are enforced by the component architecture of the system. If that architecture changes at runtime or if the properties of interest critically depend on the behavior of too many or too-large trusted components, returns will diminish.

The first step to loosen these limitations would be a library of pre-verified high-assurance components for use as trusted building blocks in such architectures. This library could include security patterns (such as input sanitizers, output filters, down-graders, and runtime monitors) potentially generated from higher-level specifications but also such infrastructure components as reusable crypto modules, key storage, file systems, network stacks, and high-assurance drivers. If the security property depends on more than one such component, it would become necessary to reason about the trustworthiness of their interaction and composition. The main technical challenges here are concurrency reasoning, protocols, and information-flow reasoning in the presence of trusted components. Despite these limitations, this work demonstrates that the rapid development of real high-assurance seL4-based systems is now a reality that can be achieved for a cost that is lower than traditional testing.

### Acknowledgments

We are grateful to Kathleen Fisher, John Launchbury, and Raymond Richards for their support as program managers in HACMS, in particular Kathleen



Fisher for having the vision to start the program. John Launchbury coined the term “seismic security retrofit.” We thank Lee Pike for feedback on an earlier draft. We would also like to acknowledge our HACMS project partners from Rockwell Collins, the University of Minnesota, Galois, and Boeing. While we concentrated on the operating system aspects of the HACMS project here, the rapid construction of high-assurance systems includes many further components, including a trusted build, as well as architecture and security-analysis tools. This material is based on research sponsored by the U.S. Air Force Research Laboratory and the Defense Advanced Research Projects Agency under agreement number FA8750-12-9-0179. The U.S. government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Research Laboratory, Defense Advanced Research Projects Agency, or U.S. government. ■

**References**

1. Alves-Foss, J., Oman, P.W., Taylor, C., and Harrison, S. The MILS architecture for high-assurance embedded systems. *International Journal of Embedded Systems* 2, 3-4 (2006), 239-247.
2. Blackham, B., Shi, Y., Chattopadhyay, S., Roychoudhury, A., and Heiser, G. Timing analysis of a protected operating system kernel. In *Proceedings of the 32<sup>nd</sup> IEEE Real-Time Systems Symposium* (Vienna, Austria, Nov. 29-Dec. 2), IEEE Computer Society Press, 2011, 339-348.
3. Boeing. Unmanned Little Bird H-6U; <http://www.boeing.com/defense/unmanned-little-bird-h-6u/>
4. Boyton, A., Andronick, J., Bannister, C., Fernandez, M., Gao, X., Greenaway, D., Klein, G., Lewis, C., and Sewell, T. Formally verified system initialisation. In *Proceedings of the 15<sup>th</sup> International Conference on Formal Engineering Methods* (Queenstown, New Zealand, Oct. 29-Nov. 1), Springer, Heidelberg, Germany, 2013 70-85.
5. Chen, H., Ziegler, D., Chajed, T., Chlipala, A., Frans Kaashoek, M., and Zeldovich, N. Using Crash Hoare logic for certifying the FSCQ file system. In *Proceedings of the 25<sup>th</sup> ACM Symposium on Operating Systems Principles* (Monterey, CA, Oct. 5-7), ACM Press, New York, 2015, 18-37.
6. Cock, D., Ge, Q., Murray, T., and Heiser, G. The last mile: An empirical study of some timing channels on seL4. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security* (Scottsdale, AZ, Nov. 3-7), ACM Press, New York, 2014, 570-581.
7. Cock, D., Klein, G., and Sewell, T. Secure microkernels, state monads and scalable refinement. In *Proceedings of the 21<sup>st</sup> International Conference on Theorem Proving in Higher Order Logics* (Montreal, Canada, Aug. 18-21), Springer, Heidelberg, Germany, 2008, 167-182.
8. Colbert, E. and Boehm, B. Cost estimation for secure software & systems. In *Proceedings of the International Society of Parametric Analysts / Society of Cost Estimating and Analysis 2008 Joint*

- International Conference* (Noordwijk, the Netherlands, May 12-14). Curran, Red Hook, NY, 2008.
9. Davis, J. and Myreen, M.O. The reflective Milawa theorem prover is sound (down to the machine code that runs it). *Journal of Automated Reasoning* 55, 2 (Aug. 2015), 117-183.
10. Dennis, J.B. and Van Horn, E.C. Programming semantics for multi-programmed computations. *Commun. ACM* 9, 3 (Mar. 1966), 143-155.
11. Elliott, T., Pike, L., Winwood, S., Hickey, P., Bielman, J., Sharp, J., Seidel, E., and Launchbury, J. Guilt-free Ivory. In *Proceedings of the ACM SIGPLAN Haskell Symposium* (Vancouver, Canada, Sept. 3-4), ACM Press, New York, 189-200.
12. Fernandez, M. *Formal Verification of a Component Platform*. Ph.D. thesis, School of Computer Science & Engineering, University of New South Wales, Sydney, Australia, July 2016.
13. Fernandez, M., Andronick, J., Klein, G., and Kuz, I. Automated verification of RPC stub code. In *Proceedings of the 20<sup>th</sup> International Symposium on Formal Methods* (Oslo, Norway, June 22-26), Springer, Heidelberg, Germany, 2015, 273-290.
14. Floyd, R.W. Assigning meanings to programs. *Mathematical Aspects of Computer Science* 19, (1967), 19-32.
15. Gonthier, G. *A Computer-Checked Proof of the Four-Colour Theorem*. Microsoft Research, Cambridge, U.K, 2005; <https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/gonthier-4colproof.pdf>
16. Gonthier, G., Asperti, A., Avigad, J., Bertot, Y., Cohen, C., Garillot, F., Le Roux, S., Mahboubi, A., O'Connor, R., Biha S.O., Pasca, I., Rideau, L., Solovyev, A., Tassi, E., and Théry, L. A machine-checked proof of the Odd Order Theorem. In *Proceedings of the Fourth International Conference on Interactive Theorem Proving, Volume 7998 of LNCS* (Rennes, France, July 22-26), Springer, Heidelberg, Germany, 2013, 163-179.
17. Gu, R., Shao, Z., Chen, H., Wu, X.(N.), Kim, J., Sjöberg, V., and Costanzo, C. CertiKOS: An extensible architecture for building certified concurrent OS kernels. In *Proceedings of the 12<sup>th</sup> USENIX Symposium on Operating Systems Design and Implementation* (Savannah, GA, Nov. 2-4), ACM Press, New York, 2016.
18. Hales, T.C., Adams, M., Bauer, G., Dang, D.T., Harrison, J., Le Hoang, T., Kaliszky, C., Magron, V., McLaughlin, S., Nguyen, T.T., Nguyen, T.Q., Nipkow, T., Obua, S., Pleso, J., Rute, J., Solovyev, A., Ta, A.H.T., Tran, T.N., Trieu, T.T., Urban, J., Vu, K.K., and Zunkeller, R. A formal proof of the Kepler Conjecture. *Forum of Mathematics, Pi, Volume 5*. Cambridge University Press, 2017.
19. Hawblitzel, C., Howell, J., Kapritsos, M., Lorch, J.R., Parno, B., Roberts, M.L., Setty, S.T.V., and Zill, B. IronFleet: Proving practical distributed systems correct. In *Proceedings of the 25<sup>th</sup> ACM Symposium on Operating Systems Principles* (Monterey, CA, Oct. 5-7), ACM Press, New York, 2015, 1-17.
20. Heiser, G. and Elphinstone, K. L4 microkernels: The lessons from 20 years of research and deployment. *ACM Transactions on Computer Systems* 34, 1 (Apr. 2016), 1:1-1:29.
21. Kanav, S., Lammich, P., and Popescu, A. A conference management system with verified document confidentiality. In *Proceedings of the 26<sup>th</sup> International Conference on Computer Aided Verification* (Vienna, Austria, July 18-22), ACM Press, New York, 2014, 167-183.
22. Klein, G., Andronick, J., Elphinstone, K., Murray, T., Sewell, T., Kolanski, R., and Heiser, G. Comprehensive formal verification of an OS microkernel. *ACM Transactions on Computer Systems* 32, 1 (Feb. 2014), 2:1-2:70.
23. Klein, G., Elphinstone, K., Heiser, G., Andronick, J., Cock, D., Derrin, P., Elkaduwe, D., Engelhardt, K., Kolanski, R., Norrish, M., Sewell, T., Tuch, H., and Winwood, S. seL4: Formal verification of an OS kernel. In *Proceedings of the 22<sup>nd</sup> ACM Symposium on Operating Systems Principles* (Big Sky, MT, Oct. 11-14), ACM Press, New York, 2009, 207-220.
24. Kumar, R., Arthan, R., Myreen, M.O., and Owens, S. Self-formalisation of higher-order logic: Semantics, soundness, and a verified implementation. *Journal of Automated Reasoning* 56, 3 (Apr. 2016), 221-259.
25. Kumar, R., Myreen, M., Norrish, M., and Owens, S. CakeML: A verified implementation of ML. In *Proceedings of the 41<sup>st</sup> ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages* (San Diego, CA, Jan. 22-24), ACM Press, New York, 2014, 179-191.
26. Kuz, I., Klein, G., Lewis, C., and Walker, A. capDL: A language for describing capability-based systems. In

- Proceedings of the First ACM Asia-Pacific Workshop on Systems* (New Delhi, India, Aug. 30-Sept. 3), ACM Press, New York, 2010, 31-35.
27. Kuz, I., Liu, Y., Gorton, I., and Heiser, G. CAmkES: A component model for secure microkernel-based embedded systems. *Journal of Systems and Software (Special Edition on Component-Based Software Engineering of Trustworthy Embedded Systems)* 80, 5 (May 2007), 687-699.
28. Leroy, X. Formal verification of a realistic compiler. *Commun. ACM* 52, 7 (July 2009), 107-115.
29. Murray, T., Maticuk, D., Brassil, M., Gammie, P., Bourke, T., Seefried, S., Lewis, C., Gao, X., and Klein, G. seL4: From general-purpose to a proof of information flow enforcement. In *Proceedings of the 2013 IEEE Symposium on Security and Privacy* (San Francisco, CA, May 19-22), IEEE Press, Los Alamitos, CA, 2013, 415-429.
30. Pnueli, A., Siegel, M., and Singerman, E. Translation validation. In *Proceedings of the Fourth International Conference on Tools and Algorithms for Construction and Analysis of Systems* (Lisbon, Portugal, Mar. 28-Apr. 4), Springer, Berlin, Germany, 1998, 151-166.
31. Rushby, J. Design and verification of secure systems. In *Proceedings of the Eighth Symposium on Operating System Principles* (Pacific Grove, CA, Dec. 14-16), ACM Press, New York, 1981, 12-21.
32. Ryzhyk, L., Chubb, P., Kuz, I., Le Sueur, E., and Heiser, G. Automatic device driver synthesis with Termite. In *Proceedings of the 22<sup>nd</sup> ACM Symposium on Operating Systems Principles* (Big Sky, MT, Oct. 11-14), ACM Press, New York, 2009, 73-86.
33. seL4 microkernel code and proofs; <https://github.com/seL4/>
34. Sewell, T., Kam, F., and Heiser, G. Complete, high-assurance determination of loop bounds and infeasible paths for WCET analysis. In *Proceedings of the 22<sup>nd</sup> IEEE Real Time and Embedded Technology and Applications Symposium* (Vienna, Austria, Apr. 11-14), IEEE Press, 2016.
35. Sewell, T., Myreen, M., and Klein, G. Translation validation for a verified OS kernel. In *Proceedings of the 34<sup>th</sup> Annual ACM SIGPLAN Conference on Programming Language Design and Implementation* (Seattle, WA, June 16-22), ACM Press, New York, 2013, 471-481.
36. Sewell, T., Winwood, S., Gammie, P., Murray, T., Andronick, J., and Klein, G. seL4 enforces integrity. In *Proceedings of the International Conference on Interactive Theorem Proving* (Nijmegen, the Netherlands, Aug. 22-25), Springer, Heidelberg, Germany, 2011, 325-340.

**Gerwin Klein** (gerwin.klein@data61.csiro.au) is a Chief Research Scientist at Data61, CSIRO, and Conjoint Professor at UNSW, Sydney, Australia.

**June Andronick** (june.andronick@data61.csiro.au) is a Principal Research Scientist at Data61, CSIRO, Conjoint Associate Professor at UNSW, Sydney, Australia, and the leader of the Trustworthy Systems group at Data61, known for the formal verification of the seL4 operating system microkernel.

**Matthew Fernandez** (matthew.fernandez@gmail.com) participated in this project while he was a Ph.D. student at UNSW, Sydney, Australia, and is today a researcher at Intel Labs, USA.

**Ihor Kuz** (ihor.kuz@data61.csiro.au) is a Principal Research Engineer at Data61, CSIRO, and also a Conjoint Associate Professor at UNSW, Sydney, Australia.

**Toby Murray** (toby.murray@unimelb.edu.au) is a lecturer at the University of Melbourne, Australia, and a Senior Research Scientist at Data61, CSIRO.

**Gernot Heiser** (gernot@unsw.edu.au) is a Scientia Professor and John Lions Chair of Computer Science at UNSW, Sydney, Australia, a Chief Research Scientist at Data61, CSIRO, and a fellow of the ACM, the IEEE, and the Australian Academy of Technology and Engineering.

DOI:10.1145/3183583

**New York State healthcare providers increased their use of the technology but delivered only mixed results for their patients.**

**BY QUANG “NEO” BUI, SEAN HANSEN, MANLU LIU, AND QIANG (JOHN) TU**

## The Productivity Paradox in Health Information Technology

“HEALTH INFORMATION TECHNOLOGY connects doctors and patients to more complete and accurate health records ... This technology is critical to improving patient care, enabling coordination between providers and patients, reducing the risk of dangerous drug interactions, and helping patients access prevention and disease management services.”

— President Barack Obama, Presidential Proclamation on National Health Information Technology Week, September 12, 2011

Health information technology (HIT)—the application of information technologies to enable and enhance the delivery of healthcare services—has been a central point of focus for U.S. healthcare policy since 2007. Both Presidents George W. Bush and Barack Obama

outlined bold goals for HIT adoption as a key facet of each of their healthcare reform efforts, promising significant benefits for healthcare providers and patients alike.<sup>20</sup> Clinical HIT systems, including electronic health records (EHRs), health information exchanges (HIEs), computerized provider order entry (CPOE), and telemedicine technologies, are seen as critical remedies to the complexity and inefficiency that have long plagued the U.S. healthcare industry.<sup>a</sup>

In 2009, the U.S. allocated more than \$30 billion, aiming to reduce healthcare costs and increase quality of care through adoption and use of HIT systems.<sup>1</sup> In that same year, the Office of the National Coordinator for Health Information Technology (ONC) was established as part of the Health Information Technology for Economic and Clinical Health (HITECH) Act of 2009 to drive HIT adoption and coordinate development of critical HIT infrastructure. The ONC oversees a range of programs (such as regional extension centers, HIEs, privacy and security policies, workforce development, and curriculum development). The HITECH Act introduced the principle of “meaningful use” of HIT, a set of guidelines for the substantive adoption and application of HIT, including

a HIT reflects a range of technologies that can be applied to the delivery and administration of healthcare service. In the present study, we focus primarily on clinical HIT systems, emphasizing EHR and HIE systems, as they have been the leading areas of emphasis in the ongoing wave of HIT adoption in the U.S.

### » key insights

- **No conclusive evidence has shown HIT contribution to health outcomes among New York State healthcare providers.**
- **Evidence indicates a HIT productivity paradox among healthcare providers that mirrors the earlier experience of the manufacturing sector.**
- **To address the paradox, a collective approach is needed involving multiple stakeholders and focusing on patient outcomes.**



corresponding incentives and penalties to motivate increased use.<sup>3</sup>

Despite aggressive investment and governmental support, evidence of HIT's contribution to health outcomes remains mixed.<sup>7</sup> A 2014 report from the U.S. Government Accountability Office (GAO) suggested that meaningful use requirements have had a modest effect, and a comprehensive strategy is needed to achieve better quality of care through HIT.<sup>14</sup> In addition, while several studies highlight perceived

benefits of HIT use (such as better clinical decision making and improved communications), other research suggests the observable effects are limited or even negative, marked by the risk of disrupted workflows, degradation of physician-patient relationships, and reduced clinical insight.<sup>25</sup> In light of these findings, many researchers and public-policy observers have called for additional studies to provide credible evidence of improved health outcomes through expanded use of HIT.<sup>26</sup>

### **Evidence from New York**

To explore the effect of HIT adoption on health outcomes, we consider the evidence from the State of New York. As the country's fourth most populous state and a national leader in HIT investment and adoption, New York offers a valuable context for assessing the effect of growing use of clinical HIT. Since 2007, New York has invested more than \$840 million<sup>b</sup>

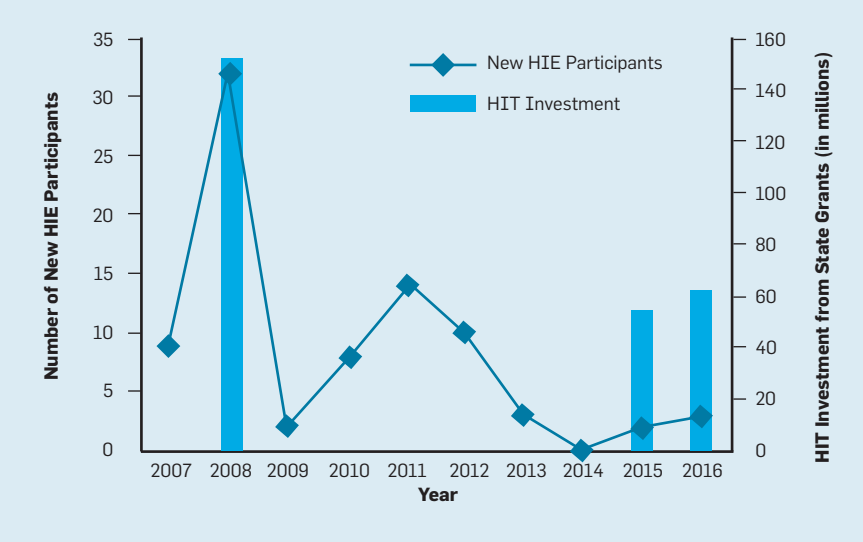
<sup>b</sup> <https://www.health.ny.gov/technology/>



**Figure 1. Adoption of EHR functionalities by hospitals in New York State.**



**Figure 2. HIE new participation rate and HIT investments from state grants in New York State, 2007–2017.**



in health information infrastructure. In that time, a variety of initiatives within the state have sought to foster information exchange, improve quality and outcomes of care, reduce healthcare costs, and engage constituents in their care.<sup>22</sup> Specifically, the state has focused on establishing governance and policies that increase participation in regional HIEs and encourage EHR system adoption by hospitals and individual providers. These efforts align with federal HIT meaningful-use initiatives aimed at creating better management of medical records and seamless coordi-

ination among healthcare providers across boundaries.<sup>c</sup>

To understand HIT effects among New York healthcare providers, we conducted a mixed-methods study using both quantitative and qualitative approaches. Our quantitative analyses used publicly available data from New York HIEs, New York State websites, and databases made available by the not-for-profit American Hospital Association and the U.S. Centers for Medicare and Medicaid Services. The dataset covered the period 2014–

<sup>c</sup> <https://www.healthit.gov/>

2015 for more than 180 hospitals across the state. We tested a structural model in which higher HIT investments would lead to increased adoption and use of EHR systems and HIEs that in turn would result in better health outcomes.<sup>d</sup> We tested the model using partial least squares software; for details, see the online appendix “Research Methodology”; [dl.acm.org/citation.cfm?doid=3183583&picked=formats](https://dl.acm.org/citation.cfm?doid=3183583&picked=formats). In addition to our quantitative analyses, we conducted a series of semi-structured interviews with more than 20 healthcare professionals from 2013 to 2016 to explore their experience around adoption and use of HIT systems. Respondents included multiple classes of clinicians (such as private practitioners, hospital physicians, and nurse-practitioners), managers, and IT professionals. The interviews were transcribed and coded in NVivo software to identify common patterns and themes.<sup>4</sup>

In general, we observed that in New York State, 2014–2015, substantial HIT investments led to the widespread acquisition and use of EHR systems, implementation of clinical decision-support functionality, and significant participation in HIEs. Specifically, New York healthcare providers implemented most EHR functionalities classified as “basic” (see Figure 1). On average, New York hospitals implemented 5.48 out of six basic EHR functions (such as electronic document viewing, results viewing, CPOE, and decision support); and hospitals differ only by the degree of implementation around other advanced EHR functionalities (such as barcode identification, telehealth, mobile device connections). Additionally, the number of new hospitals joining local HIEs corresponds to the surge in the state’s public funding for HIT investment in 2008, significantly augmented in 2015 and 2016 (see Figure 2).<sup>e</sup> As of 2018, over 80% of New York healthcare-provider

<sup>d</sup> Details of our research methodology is provided in the online appendix “Research Methodology”; [dl.acm.org/citation.cfm?doid=3183583&picked=formats](https://dl.acm.org/citation.cfm?doid=3183583&picked=formats)

<sup>e</sup> These local HIEs received public grants from New York State to increase information sharing among hospitals; [https://www.health.ny.gov/technology/financial\\_investment.htm](https://www.health.ny.gov/technology/financial_investment.htm)

organizations—162 out of 197—had joined HIEs and regularly exchange medical records data electronically.

While the majority of New York hospitals have implemented and used EHR and HIEs in their practice, the evidence is inconclusive with respect to how these initiatives have affected quality of care and broad health outcomes across the state. We found no evidence of a relationship between HIT use and such critical health outcomes as improved interpersonal care, customer satisfaction, customer loyalty, patient mortality, and reduced ER waiting times (see Figure 3). These results are in line with previous studies suggesting unclear evidence of HIT effects.<sup>15</sup>

While HIE participation and EHR use levels reveal no significant relationships with most outcome measures, we were surprised to find EHR use also does have a significant adverse relationship with patient readmission rates and complication rates. To further explore this counterintuitive result, we looked at the social-capital index in each county where the hospitals operate. The social-capital index<sup>27</sup> reflects the socio-economic growth of a community.<sup>f</sup> The post-analyses suggest areas with low social capital often see higher readmission rates and complication rates. This low score is due to such factors as rural market, low social support, and low educational rate. One possible explanation for our counterintuitive finding is that hospitals in areas with low social capital encounter inherent difficulties that in turn increase patient readmission and complication rates regardless of their use of HIT. We encourage future research into this relationship.

Augmenting our quantitative analysis, our conversations with healthcare providers suggest mixed feelings and skepticism toward the expected values of HIT. In particular, many clinicians were concerned that HIT initiatives were too often not motivated by patient-oriented objectives and might undermine

<sup>f</sup> The social capital index was developed by the Northeast Regional Center for Rural Development (<http://aese.psu.edu/nercd>) and uses an array of individual and community factors to measure the socioeconomic growth of a community.

rather than enhance the quality of care providers render. Prominent concerns include the perception that HIT adoption results in extra workload, ineffective communication, poor information quality, and ineffectiveness addressing operational needs. The following illustrative statements highlight the concerns shared by our respondents:

“This whole business about electronic medical records helping with communication I think is a total fallacy. I think it really hinders communication, unless you freehand-type or you dictate, which defeats the main purpose of electronic medical records.” — Physician, Pediatrics

“I hear complaints from patients saying, ‘They’re looking at the computer and not at me.’” — Physician, Pediatrics

“This is my issue with all electronic medical records: The notes that

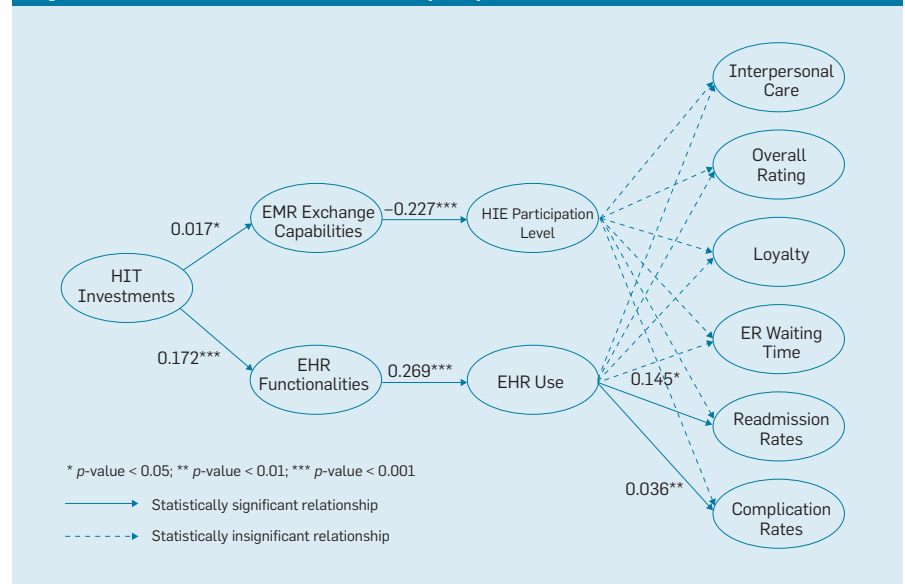
are generated have an awful lot of words but communicate very little.” — Physician, Family Practice

“The highlighted efficiency from reducing duplicate lab tests and cutting costs is just not there yet. I am not really sure that an EHR will provide the savings that are talked about.” — Physician, Internal Medicine

“I have charting at home. I ended up having to get a laptop through my work budget to bring home so that I wasn’t sitting at the office until ... I would see my last person around 4:20, and I would be there until 6:30 doing charting because of being slow with the system and be more attentive to the patient than I was to the computer.” — Nurse Practitioner

In summary, our mixed-methods analyses suggest strong evidence of increased adoption and use of EHR and HIE among New York healthcare

**Figure 3. Effects of HIT investment on hospital performance.**



**Explaining the IT productivity paradox in HIT contexts.**

Causes	Description
HIT mismeasurement	Most HIT measures focus on efficiency rather than effectiveness. Recent efforts like “meaningful use” level 2 are useful but far from satisfactory.
Delay delivering HIT benefits	HITs are complex systems that require an average of two to four years to deliver significant benefits to healthcare providers.
Redistribution of HIT benefits	HIT gains are offset by unintended consequences in healthcare processes and procedures, including extra work and lack of human-doctor interaction.
Mismanagement of HIT systems	Healthcare managers are not adequately trained to deal with the complexity of HIT systems.

providers but cast doubt on the claim of substantial HIT effects on health outcomes.

### Assessing HIT

The challenge of finding evidence of practical benefits accruing from IT investment is not unique to healthcare. Indeed, the IT productivity paradox,<sup>5</sup> an apparent disconnect between investment in IT resources and discernible impact on organizational performance, has been widely observed with earlier waves of IT adoption in manufacturing and other industrial sectors. In a seminal disposition on the phenomenon, Erik Brynjolfsson<sup>5</sup> summarized a number of concerns that emerged in the 1980s and early 1990s around a lack of productivity gains corresponding to rapid adoption of IT resources. Several analysts had noted significant growth in technological investment and innovation across developed economies had coincided with disappointing gains—or even declines—in productivity.<sup>11,23</sup> It appears that just as in the manufacturing sector, HIT is struggling to produce credible improvements in key measures of performance. The IT productivity paradox has once again surfaced in the healthcare industry.

In his exploration of the phenomenon, Brynjolfsson<sup>5</sup> suggested four possible explanations: mismeasurement, temporal lags, redistribution, and mismanagement. Mismeasurement refers to the idea that we lack appropriate measures for productivity in a service-based economy, with most traditional, manufacturing-oriented measures of productivity failing to account for indirect benefits (such as quality and customer satisfaction). The issue of a temporal lag centers on the possibility that gains from IT investment could take years to develop as organizations change their ways of working and the skills of their personnel. Less optimistically, redistribution suggests the dearth of productivity improvements could be the result of new IT resources merely shifting productivity gains (or losses) from some market participants to others. That is, IT may indeed create productivity gains for some players, but such gains are counterbalanced by losses for other individuals or organizations. Finally,

Brynjolfsson<sup>5</sup> said the productivity paradox could derive from the fact that “IT really is not productive at the firm level” or that managers have not been able to apply IT resources effectively.

“One of the health plans locally made an attempt at doing reporting [on provider efficiency]. They based it totally on cost. So [when they looked at the report] one of the physicians that was in the top had died six months before. He looked very efficient from a cost perspective. He hadn’t generated any cost to the system.”  
— Director, Medical Society

In the years since the initial explorations of the productivity paradox, the apparent disconnect between IT investment and organizational outcomes has been largely resolved; that is, researchers have concluded that the first two explanations—mismeasurement and lagged effects—were the primary drivers of the paradoxical observations<sup>6</sup> and that IT investment is indeed correlated with significant improvement in various measures of value at firm, industry, and country levels, but such gains might take years to materialize.<sup>12,13,28</sup> However, the idiosyncratic characteristics of the healthcare sector (such as institutional heterogeneity, combination of public and private influences, and comparatively late adoption of IT innovations) underscore important differences with the sectors explored previously. Consequently, a thorough consideration of diverse possible factors is warranted.<sup>17,19</sup> Indeed, the four proposed explanations associated with the IT productivity paradox suggest critical clues for considering the inconclusive effects of contemporary HIT investment (see the table here).

As our analysis highlights, the idiosyncratic nature of the healthcare domain introduces a range of relatively novel outcome measures for HIT investment, including quality of care, readmission rates, complication rates, and diagnostic accuracy. While these are well-established measures of effectiveness for health services, their appropriateness for evaluation of the efficiency and effectiveness of HIT remains to be seen. Interestingly, the concept of “meaningful use” that has driven adoption of much HIT since the passage of the HITECH Act focuses al-

most exclusively on measures of input, or use of a certified system for records capture, reporting, and data exchange. Despite incentivizing inputs, the ultimate objective of the meaningful-use guidelines is substantive improvement in health outcomes (such as quality of care and fewer medical errors). This disconnect suggests we may need better measures to capture the contribution of HIT investment to those ultimate objectives.<sup>29</sup>

With respect to the question of a temporal lag, a number of studies have suggested this is a critical issue in the healthcare context. For example, Menon et al.<sup>21</sup> found it takes, on average, from two to four years for HIT systems to improve health outcomes in a given healthcare-provider organization. Many providers lack the necessary IT skills to quickly get acquainted with new HIT tools and procedures, making implementation more challenging. Given the fact that the uptick of HIT investment commenced only in 2009, it may take many more years for HIT influence to ripple across healthcare providers.

The possibility of redistributive effects also warrants consideration in the HIT context. As the comments of our study respondents underscore, many healthcare providers fear the efficiency in reporting and data analysis HIT engenders for insurance firms and regulators comes at the expense of decreased efficiency for clinicians who actually deliver clinical care. Indeed, this shifting of efficiencies and burdens can be seen in one of the most common organizational responses to HIT adoption: dedicated “scribes” to capture data during a clinical encounter. The question of whether efficiency gains in one facet of the healthcare system are partially outweighed by efficiency or process losses elsewhere in the system thus requires additional analysis.

Finally, the mismanagement of IT resources may well play a role in the mixed results of HIT adoption. Concerns expressed to us by healthcare providers regarding the usefulness of HIT resources suggest the possibility of missteps in the design, implementation, and/or ongoing use of these systems. These concerns lead to negative perceptions of HIT that likely result in misuse and jeopardize overall perfor-



mance. Yet such concerns from multiple stakeholders are hardly captured in HIT development, and IT staff is inexperienced in helping and adjusting the new systems to local needs. In our interviews, several healthcare providers expressed their struggles in managing new systems due to their limited time and personal technology anxiety.

While each of the proposed mechanisms for paradoxical outcomes has some applicability in the healthcare context, the rich vein of research that grew out of the productivity paradox also offers some critical caveats for assessing the practical effect of IT investment and use.<sup>12,13,28</sup> First, significant variation exists across firms and industries with respect to the effect of IT investment on organization performance.<sup>9</sup> Second, this variation and the existence of temporal lags are tied to the fact that performance gains are often associated not merely with the adoption of new IT resources but with the concomitant redesign of business processes and investment in complementary assets and skills.<sup>6,28</sup> Finally, the healthcare literature reveals that measures of productivity or business value remain ambiguous and highly contingent on firm or industry conditions. Applying these lessons in the context of HIT, the evidence points to the need for more research to understand the complex nature of the healthcare industry and its business processes, along with interdependence among healthcare stakeholders in HIT development, adoption, and use.

### **Beyond the Paradox**

Based on our analyses of the effects of clinical HIT adoption, we find that a number of viable mechanisms are available for achieving enhanced health outcomes as a result of expanded HIT use, moving from meaningful use to meaningful results. The U.S. healthcare sector is an interdependent system. Leveraging and extending past insights from research on the productivity paradox and IT business value in general, we find it would benefit from a collective approach that brings together such diverse entities as hospitals, insurance companies, regulators, and HIT vendors to seek systemic improvements.



**We found no evidence of a relationship between HIT use and such critical health outcomes as improved interpersonal care, customer satisfaction, customer loyalty, patient mortality, and reduced ER waiting times.**



**Efforts by the healthcare community.** Resolution of the apparent HIT productivity paradox will require more than the isolated efforts of healthcare providers, calling for a community effort. To this end, we suggest a stronger leadership role for HIE-facilitating entities, including regional health information organizations (RHIOs). As the ONC acknowledges, RHIOs are central to data exchange across healthcare institutions.<sup>30</sup> Given the challenges in the healthcare industry, we propose that RHIOs should be more than mere data clearinghouses but formalized institutions that significantly improve HIT use, especially in two major roles:

*Encourage learning and adaptation mechanisms in HIT practices.* As with many enterprise IT systems, HIT platforms are frequently complex and rigid, requiring significant resources and enterprise-level effort to implement effectively. For such complex projects to yield tangible results, it takes time for users to adapt to new routines and practices, patients to get accustomed to new processes and functionality, and in-house IT staff to discern what system modifications would make the new system better fit with local needs. RHIOs can serve as a platform through which different parties can share resources, help others learn, and contribute back to the broader community. In addition to creating a mechanism for the development and exchange of a shared knowledgebase, these organizations represent a bridge between different types of hospitals: large/small, public/private, urban/rural. Managers can consider practices proposed in RHIO-based discourses to foster learning and adaptation in HIT adoption (such as using collaborative teams to explore HIT functionalities, rewards to enforce positive behaviors, and centers of excellence around HIT best practices).

*Put users at the center of the HIT experience.* Commonly found in our interviews and in the HIT literature is the concern that HIT policies have pushed healthcare providers toward a techno-centric perspective in which HIT is pursued “for IT’s sake” and HIT systems are designed without substantive input from prospective users.<sup>10</sup> It is critical not to lose sight of the most important HIT stakeholders—

the patients whose wellness is directly affected by HIT use and the healthcare providers who guide the patients through the treatment process. Healthcare providers thus need to encourage both policymakers and technology developers to emphasize inclusion of patients and healthcare providers in the design processes of HIT systems. To enable a coherent and seamless experiences across HIT systems, RHIOs can act as a forum for users' experiences to be heard, providers' suggestions to be noted, and community members' opinions to be constructively formed. Such a collaborative approach is essential to HIT success, because, despite the existence of competitive forces among healthcare providers, patient wellness should be regarded as the ultimate goal for all parties.

**Academic research.** Although the literature on HIT evaluation is expanding rapidly, there has not been a parallel increase in academic understanding of how HIT contributes to patient outcomes or how it can be used to improve health and healthcare. The related research should be adapted to meet the needs of clinicians, healthcare administrators, and health policymakers. We thus suggest the following actions for academic researchers:

*Develop enhanced measurements for clinical HIT impact.* As noted, the healthcare system today lacks adequate outcome-oriented measurements of the efficiency and effectiveness of HIT. For example, the U.S. Department of Health and Human Services released final criteria of "meaningful use" in 2010, aiming to improve quality and efficiency of care by encouraging clinicians and hospitals to use EHRs. However, as of 2017, the existing measurement of "meaningful use" focused exclusively on input metrics. Accordingly, researchers are well positioned to develop appropriate means of outcome measurement to connect HIT investment with productivity and clinical relevance. One important improvement that can be made in HIT evaluation is increased measurement of context, implementation, and context-sensitivity of effectiveness.<sup>18</sup> Exploring contextual and/or organizational factors would help address



**Many clinicians were concerned that HIT initiatives were too often not motivated by patient-oriented objectives and might undermine rather than enhance the quality of care providers render.**



lingering questions about potential mismeasurement in assessing the long-term impact of HIT. As we have noted in reference to the observed adverse effect of EHR use on patient re-admission and complication rates in the New York State context, a range of factors (such as urban/rural setting, social capital within a region, and academic vs. non-academic hospital adoption) can influence the contribution of HIT use on health outcomes. Clarifying the most relevant factors would thus aid the healthcare field in untangling the causal dynamics around HIT adoption and use. In addition, another important improvement regarding HIT evaluation would be increased use of evidence-based and clinical HIT research.<sup>24</sup> Using rich data generated through clinical HIT systems, future studies could examine how HIT as "informatic intervention" can significantly improve patients' health outcomes. Other initiatives (such as the Precision Medicine Initiative launched in 2016 by the U.S. National Institutes of Health) also underscore the need for more evidence-based HIT research in the future.<sup>g</sup>

*Learn how to realize value from HIT.* Early studies of HIT adoption and use focused largely on determining whether a particular HIT functionality created value and to what extent. With increasing adoption of EHRs and other forms of HIT, it is no longer sufficient for researchers to ask whether HIT creates value in terms of health outcomes.<sup>16</sup> As researchers, we need to help healthcare providers and policymakers learn how to realize value from HIT. That is, while HIT is being adopted, researchers should focus on exploring the causal mechanisms underlying its use to deliver health value to patients. Such theory-building research could help clarify the antecedents of the productive application of HIT resources. In particular, such research could leverage recent research shifting from consideration of simple IT use to ef-

<sup>g</sup> The Precision Medicine Initiative was launched in 2016 by the U.S. National Institutes of Health as a national, large-scale research participation group for the testing and study of evidence-based interventions; <https://allofus.nih.gov/>

fective, enhanced, or idiosyncratic use of IT resources.<sup>2,8</sup> Insight from the research could inform the aforementioned efforts among healthcare system participants to identify and disseminate best practices and foster more productive use patterns.

**Efforts by policymakers.** Policymakers play a significant role in each of the measures we have proposed, as in community building through RHIOs and advancing outcome-oriented measures of HIT use. While they should work with academic researchers and the industry to identify more relevant metrics for healthcare providers, it is equally important they maintain a holistic view of the healthcare value chain. Instead of focusing on policies that incentivize only EHR adoption or HIE participation, policymakers should also consider how to promote experimentation both within and across geographic boundaries. This might include more flexible use-style incentive programs that reward not only hospital-by-hospital efforts but also cross-hospital, cross-state, and cross-boundary initiatives. It is difficult today to promote technologies that provide value across geographical locations (such as telemedicine) or across institutional boundaries (such as healthcare supply-chain systems). In order to promote innovation and collaboration, policymakers might thus want to consider measures that target multiple parties in a healthcare value chain rather than a limited number of dominant players. This would include support for public-private partnerships that bring together healthcare providers, payer organizations, and HIT providers or initiatives that include large-scale participation groups (such as the Precision Medicine Initiative). Such efforts could leverage emergent technologies (such as big data analytics platforms, mobile health apps, and social media) to quickly assess the efficacy of a diverse set of HIT projects and channel resources toward the ones that show the greatest promise for bridging the gap between HIT use and health outcomes across populations.

## Conclusion

IT use in the healthcare industry has experienced tremendous growth and

attention since 2007. Yet concrete and credible evidence that HIT improves health outcomes remains inconclusive. Our investigation of New York State healthcare providers further indicates the healthcare industry may be experiencing an ongoing HIT productivity paradox, mirroring earlier patterns in manufacturing and other industrial sectors. While potential HIT contribution to health outcomes remains an open question, we suggest a collective approach is needed to address the many issues raised by the HIT productivity paradox and hope our research invites further inquiry into this important issue. **C**

## References

- Adler-Milstein, J., Bates, D.W., and Jha, A.K. A survey of health information exchange organizations in the United States: Implications for meaningful use. *Annals of Internal Medicine* 154, 10 (May 2011), 666–671.
- Bagayogo, F.F., Lapointe, L., and Bassellier, G. Enhanced use of IT: A new perspective on post-adoption. *Journal of the Association for Information Systems* 15, 7 (July 2014), 361–387.
- Blumenthal, D. and Tavenner, M. The 'meaningful use' regulation for electronic health records. *The New England Journal of Medicine* 363, 6 (Aug. 5, 2010), 501–504.
- Boyatzis, R.E. *Transforming Qualitative Information: Thematic Analysis and Code Development*. Sage Publications, Thousand Oaks, CA, 1998.
- Brynjolfsson, E. The productivity paradox of information technology. *Commun. ACM* 36, 12 (Dec. 1993), 66–77.
- Brynjolfsson, E. and Hitt, L.M. Beyond the productivity paradox. *Commun. ACM* 41, 8 (Aug. 1998), 49–55.
- Buntin, M.B., Burke, M.F., Hoaglin, M.C., and Blumenthal, D. The benefits of health information technology: A review of the recent literature shows predominantly positive results. *Health Affairs* 30, 3 (2011), 464–471.
- Burton-Jones, A. and Grange, C. From use to effective use: A representation theory perspective. *Information Systems Research* 24, 3 (Mar. 2012), 632–658.
- Chari, M.D., Devaraj, S., and David, P. The impact of information technology investments and diversification strategies on firm performance. *Management Science* 54, 1 (Jan. 2008), 224–234.
- Cho, K.W., Bae, S.-K., Ryu, J.-H., Kim, K.N., An, C.-H., and Chae, Y.M. Performance evaluation of public hospital information systems by the information system success model. *Healthcare Informatics Research* 21, 1 (Jan. 2015), 43–48.
- David, P.A. The dynamo and the computer: An historical perspective on the modern productivity paradox. *The American Economic Review* 80, 2 (May 1990), 355–361.
- Dedrick, J., Gurbaxani, V., and Kraemer, K.L. Information technology and economic performance: A critical review of the empirical evidence. *ACM Computing Surveys* 35, 1 (Mar. 2003), 1–28.
- Devaraj, S. and Kohli, R. Information technology payoff in the healthcare industry: A longitudinal study. *Journal of Management Information Systems* 16, 4 (Apr. 2000), 41–67.
- Government Accountability Office. *Electronic Health Record Programs: Participation Has Increased, but Action Is Needed to Achieve Goals, Including Improved Quality of Care*. Washington, D.C., 2014; <https://www.gao.gov/assets/670/661399.pdf>
- Harrison, M.I., Koppel, R., and Bar-Lev, S. Unintended consequences of information technologies in health care: An interactive sociotechnical analysis. *Journal of the American Medical Informatics Association* 14, 5 (Sept. 2007), 542–549.
- Jha, A.K., Burke, M.F., DesRoches, C., Joshi, M.S., Kralovec, P.D., Campbell, E.G., and Buntin, M.B. Progress toward meaningful use: Hospitals' adoption of electronic health records. *American Journal of*

*Managed Care* 17, 12 (Dec. 2011), 117–124.

- Jones, S.S., Heaton, P.S., Rudin, R.S., and Schneider, E.C. Unraveling the IT Productivity Paradox: Lessons for Health Care. *The New England Journal of Medicine* 366, 24 (June 14, 2012), 2243–2245.
- Jones, S.S., Rudin, R.S., Perry, T., and Shekelle, P.G. Health information technology: An updated systematic review with a focus on meaningful use. *Annals of Internal Medicine* 160, 1 (Jan. 2014), 48–54.
- Lapointe, L. The IT productivity paradox in health: A stakeholder's perspective. *International Journal of Medical Informatics* 80, 2 (Feb. 2011), 102–115.
- Leidner, D.E., Preston, D., and Chen, D. An examination of the antecedents and consequences of organizational IT innovation in hospitals. *Journal of Strategic Information Systems* 19, 3 (Sept. 2010), 154–170.
- Menon, N.M., Yaylacioglu, U., and Cezar, A. Differential effects of the two types of information systems: A hospital-based study. *Journal of Management Information Systems* 26, 1 (July 2009), 297–316.
- New York eHealth Collaborative. *State HIE Cooperative Agreement Program Strategic Plan*. New York, 2009; <https://www.healthit.gov/topic/onc-hitech-programs/state-health-information-exchange>
- Panko, R.R. Is office productivity stagnant? *MIS Quarterly* 15, 2 (June 1991), 191–203.
- Payne, P.R.O., Lussier, Y., Foraker, R.E., and Embi, P.J. Rethinking the role and impact of health information technology: Informatics as an interventional discipline. *BMC Medical Informatics and Decision Making* 16, 40 (Mar. 29, 2016), 1–7.
- Rosenbaum, L. Transitional chaos or enduring harm? The EHR and the disruption of medicine. *The New England Journal of Medicine* 373, 17 (Oct. 22, 2015), 1585–1588.
- Rudin, R.S., Motala, A., Goldzweig, C.L., and Shekelle, P.G. Usage and effect of health information exchange: A systematic review. *Annals of Internal Medicine* 161, 11 (Dec. 2014), 803–812.
- Rupasingha, A., Goetz, S.J., and Freshwater, D. The production of social capital in U.S. counties. *The Journal of Socio-Economics* 35, 1 (Feb. 2006), 83–101.
- Schryen, G. Revisiting IS business value research: What we already know, what we still need to know, and how we can get there. *European Journal of Information Systems* 22, 2 (Mar. 2013), 139–169.
- Sharma, L., Chandrasekaran, A., Boyer, K.K., and McDermott, C.M. The impact of health information technology bundles on hospital performance: An econometric study. *Journal of Operations Management* 41 (Jan. 2016), 25–41.
- Vest, J.R. and Gamm, L.D. Health information exchange: Persistent challenges and new strategies. *Journal of the American Medical Informatics Association* 17, 3 (May 2010), 288–294.

**Quang "Neo" Bui** (qnbui@saunders.rit.edu) is an assistant professor of management information systems in the Saunders College of Business of the Rochester Institute of Technology, Rochester, NY, USA.

**Sean Hansen** (shansen@saunders.rit.edu) is an associate professor of management information systems in the Saunders College of Business of the Rochester Institute of Technology, Rochester, NY, USA.

**Manlu Liu** (manliulu@saunders.rit.edu) is an associate professor of management information systems and accounting in the Saunders College of Business of the Rochester Institute of Technology, Rochester, NY, USA.

**Qiang (John) Tu** (jtu@saunders.rit.edu) is a professor of management information systems and the Senior Associate Dean in the Saunders College of Business of the Rochester Institute of Technology, Rochester, NY, USA.



## The future of computing research relies on addressing an array of limitations on a planetary scale.

BY BONNIE NARDI, BILL TOMLINSON, DONALD J. PATTERSON, JAY CHEN, DANIEL PARGMAN, BARATH RAGHAVAN, AND BIRGIT PENZENSTADLER

# Computing within Limits

COMPUTING RESEARCHERS AND practitioners are often seen as inventing the future. As such, we are implicitly also in the business of predicting the future. We plot trajectories for the future in the problems we select, the assumptions we make about technology and societal trends, and the ways we evaluate research.

However, a great deal of computing research focuses on one particular type of future, one very much like the present, only more so. This vision of the future assumes that current trajectories of ever-increasing production and consumption will continue. This focus is perhaps not surprising, since computing machinery as we know it has existed for only 80 years, in a period of remarkable industrial and technological expansion. But humanity is rapidly approaching, or has already exceeded, a variety of planet-scale limits related to the global climate system, fossil fuels, raw materials, and biocapacity.<sup>28,32,38</sup>

It is understandable that in computing we would not focus on limits. While planetary limits are obvious in areas such as extractive capacity in mining or fishing,

or the amount of pollution an ecosystem can bear, limits are less obvious in computing. Many believe the only limit worth considering is human ingenuity, and that we can surpass any and all other limits if we, as a global community, pool our creative resources. But we collectively face new global conditions that warrant our attention.

In this article we explore the relationship between these potential futures and computing research. What hidden assumptions about the future are embedded in most computing research? What possible or even probable futures are we ignoring? What work should we be doing to respond to fundamental planetary limits, and to the ecological and energy constraints that global society faces over the coming years and decades? Confronting such limits is likely to present challenges that we—humanity—have never before faced.

Given that computing underlies virtually all the infrastructure of global society—in commerce, communication, transportation, agriculture, manufacturing, education, science, healthcare, and governance—computing has an enormous role to play in responding to global limits and in shaping a society that meaningfully adapts to them. We contend that the root of much of computing research has been driven predominantly by growth-oriented visions

### » key insights

- **Most computing work is premised on industrial civilization's default worldview in which ongoing economic growth is both achievable and desirable.**
- **This growth-focused worldview, however, is at odds with findings from many other scientific fields, which see growth as deeply problematic for ecological and social reasons.**
- **We proposed that the computing field transition toward "computing within limits," exploring ways that new forms of computing supported well-being while enabling human civilizations to live within global ecological and material limits.**
- **Computing underlies virtually all the infrastructure of global society, and will therefore be critical in shaping a society that meaningfully adapts to global limits.**



of society's future.<sup>26,34,39</sup> If we broaden our view to a more diverse set of possible futures, including non growth-reliant futures, the societal challenges of ecological and energy limits can shape concrete technical challenges in computing research and practice.

In order to consider these futures, we have been building a community of scholars from computer science and engineering, information science, and social science, ecology, agriculture, and earth sciences to explore what we call “computing within limits” or “LIMITS” for short. The LIMITS research community integrates three topics: current and near-future ecological, material, and energy limits; the ways new forms of computing may help support well-being while living within these limits; and the impact these limits are likely to have on the field of computing. LIMITS is concerned with the material impacts of computation itself, but, more broadly and more importantly, it engages a deeper, transformative shift in computing research and practice to one that would use computing to contribute to the overall process of transitioning to a future in which the well-being of humans and other species is the primary objective.

The LIMITS perspective is related to Green IT,<sup>17</sup> sharing an interest in improvements in efficiency and other traditionally “green” research topics. However, LIMITS research questions Green IT's implicit assumption that we can “engineer around” the finiteness of the Earth's resources and waste

capacity. LIMITS sees ecological and environmental issues as a “predicament”—that is, a situation for which there are not likely to be clear-cut “solutions” but rather a constellation of complex issues that requires broad new assumptions and approaches. We seek to engage this predicament by adopting a new framing for computing research. We question the focus on ongoing economic growth that lies at the heart of industrial civilization and propose a shift from emphasis on standards of living and material productivity to an emphasis on long-term well-being. LIMITS research looks ahead to future scenarios cognizant of work such as that of Rockström et al.<sup>28</sup> that

draws attention to “planetary boundaries that must not be transgressed.” Each of these topics will be discussed in greater detail.

Here, we present background literature in ecological economics and archaeology that has informed LIMITS research, and then review computing research in sustainable human computer interaction, crisis informatics, and information and communication technology for development (ICTD). Although LIMITS researchers come from many subfields of computing including networking and software engineering, research in these three areas in particular is closely related to LIMITS with potential for deeper fu-




ture connections. We then briefly summarize the three annual workshops on LIMITS that began in 2015. Finally, we discuss several key principles that have arisen from LIMITS work to guide future research. We see work in this area as a subfield that is an important alternative to traditional growth-oriented computing research.


### Background

Since the beginning of computing, all research and development has taken place against a backdrop of exponential growth of, for example, transistors per integrated circuit (Moore's Law), disk storage density (Kryder's Law), bandwidth capacity (Nielsen's Law), and fiber-optic capacity (Keck's Law). These developments have led to the establishment of a "cornucopian paradigm"<sup>23</sup> where the design of new services stimulates demand, which drives growth of increased infrastructure capacity, which then cycles back to enable the design of new services in a self-perpetuating cycle. The idea that exponential growth of computing capacity and an ever-expanding infrastructure for computing will continue into the future is usually taken for granted. We draw from research in ecological economics and the historical record in archeology to question this assumption.

This research suggests that other futures are not just possible but probable. While most economists sidestep questions of finite resources,<sup>6</sup> economists in the subfield of ecological economics have grappled with these questions for decades. How can we maintain or increase well-being while staying within ecological limits? How can we promote well-being and not exceed the assimilative and regenerative capacities of the Earth's biochemical life-support systems? We have already exceeded many such limits through, for example, overfishing, deforestation, soil depletion, falling water tables, rising temperatures, and emitting CO<sub>2</sub> and other greenhouse gases at rates that dangerously increase their concentrations in the atmosphere.<sup>28,32,38</sup> Ecological economist Herman Daly has proposed that we abandon the idea of striving for economic growth in favor of a steady-state economy (in line with classical economist Adam Smith's idea that the economy would



## Computing has an enormous role to play in responding to global limits and in shaping a society that meaningfully adapts to them.



eventually reach a "stationary state"). A steady-state economy would maintain material throughput at a rate that is largely stable across time and that remains within ecological limits.<sup>7</sup> At the same time, Daly notes that culture and society need not be static: "Not only is quality free to evolve, but its development is positively encouraged in certain directions. If we use 'growth' to mean quantitative change, and 'development' to refer to qualitative change, then we may say that a steady-state economy develops but does not grow, just as the planet Earth, of which the human economy is a subsystem, develops but does not grow." Daly suggests that a single-minded focus on growing the economy comes at the eventual cost of decreasing human well-being and quality of life. Such growth results in, for example, charging for things that used to be free, the health consequences of polluting the environment, and decreasing long-term possibilities to produce food or earn a livelihood.

Looking at societal trends through the lens of human history, archaeologist Joseph Tainter's book *The Collapse of Complex Societies* argues that civilizations eventually collapse, declining over a period of decades or centuries.<sup>33</sup> Analyzing extensive historical and archaeological materials, Tainter presented collapse as a process that arises from increasing societal complexity, which, over time, creates burdens for systems that they eventually cannot sustain.

Decline will result in less material abundance as we push the limits of the Earth's resources necessary for economic activity. But it is not necessary for our society to end in abject collapse. The societies that Tainter studied—the Maya, the Mesopotamians, the Minoans, the Inca, the Romans, the Egyptians, and others—did not possess the resources of science, history, and technology that we have amassed in the last 500 years. These resources have the potential to be usefully deployed to fashion a transition from the current, unsustainable system to a new system based on today's realities. We optimistically assume that with advances in science and progress in philosophies of human rights, we have a good chance of transformative change to a system more like the



steady-state economy Herman Daly envisions. The implication of the work in ecological economics and archaeology is that we should endeavor to build computer systems that aim at increasing well-being and quality of life while contributing to staying within ecological limits. Foregrounding human well-being is supported by the ACM Code of Ethics and Professional Conduct, the first imperative of which states: “As an ACM member I will contribute to society and human well-being.” (<https://www.acm.org/aboutacm/acm-code-of-ethics-and-professional-conduct>)

We turn now to a review of computing literature that has been foundational for the development of computing within LIMITS perspectives.

### SCHI: Sustainable Human-Computer Interaction

The Sustainable Human-Computer Interaction community is about a decade old, and a number of LIMITS researchers have roots in this area. Eli Blevis’s “Sustainable Interaction Design”<sup>3</sup> is a primary source, offering a rubric to identify how interaction designs lead to material effects, as well as several principles for engaging in sustainable interaction design. Early papers that sparked interest among LIMITS researchers were Jeff Wong’s “Prepare for Descent: Interaction Design in Our New Future”<sup>40</sup> and Silberman and Tomlinson’s “Precarious Infrastructure and Postapocalyptic Computing.”<sup>31</sup> Several high-profile CHI papers drew attention to the challenges of sustainability and the shortcomings of SCHI work in failing to address questions of physical, material, and energy limits. DiSalvo et al.’s “Mapping the Landscape of Sustainable HCI”<sup>8</sup> sought to provide structure to the array of papers in SCHI, and identified gaps in the areas being studied, such as the need to focus on collectives and broader contexts, not just individuals, the importance of engaging with policy issues, and stronger connections to sustainability work in fields outside of computing.

From this context, Tomlinson et al.’s “Collapse Informatics”<sup>35</sup> was the first full treatment of LIMITS topics in the SCHI community. This paper explored “the study, design, and development of sociotechnical systems in the abundant present for use in a future

of scarcity.” This work helped lay the groundwork, along with papers from other subfields of computing<sup>24,37</sup> for LIMITS research.

LIMITS has drawn heavily from collapse informatics but shifts emphasis to planetary limits rather than societal decline. LIMITS focuses on exposing basic processes of resource use and waste management in complex human systems. The metrics used to assess sustainability must shift correspondingly. As examples, Pargman and Raghavan’s “Rethinking Sustainability in Computing: From Buzzword to Non-negotiable Limits”<sup>20</sup> and Raghavan and Pargman’s “Means and Ends in Human-Computer Interaction: Sustainability through Disintermediation,”<sup>25</sup> offer major contributions, arguing that “sustainability” must be grounded in rigorous metrics arising from planetary limits, and that the complexity of societal systems might be reduced, easing resource use and waste production. The forthcoming edited collection *Digital Technology and Sustainability: Engaging the Paradox*<sup>10</sup> incorporates influences from LIMITS research. Several of the papers mentioned here as well as Preist et al.<sup>23</sup> have won best paper awards, signaling interest in the issues.

### Crisis Informatics

We are often asked if computing within LIMITS is the same as crisis informatics. Crisis informatics is concerned with technology-based studies of disaster planning and response, and

constitutes an important subfield of human-computer interaction.<sup>19</sup> There are some key differences between crisis informatics and LIMITS, although we think that in the future the two may increasingly mutually inform one another. At present, crisis informatics research generally assumes an external entity that enacts a rescue when a disaster, such as a flood or earthquake, occurs. Events are conceived as localized, describing a space into which the surrounding society can pour resources to alleviate the resulting disorder and disruption. These scenarios accurately describe an important subset of possible issues confronting human civilizations. LIMITS, however, assumes long time frames and a global spatial scale. There is no external entity to provide relief. LIMITS emphasizes phenomena such as climate change, soil erosion, water pollution, civic instability, mass migration, reduced infrastructure, and an economy that requires continuous growth.<sup>4,5,14,20,21,24,30,36</sup>

Potentially there is a strong link between LIMITS and crisis informatics. Some crisis informatics researchers are beginning to examine long-term processes underlying crises, suggesting that when looked at more broadly, “crises” are often more than acute events of short duration, with roots in underlying processes that may have been developing over decades.<sup>1</sup> This understanding provides a bridge for future development and crossfertilization between the two subdisciplines.





### ICTD: Information and Communication Technology for Development

ICTD is a relatively young field that has explored the potential of computing for improving the socioeconomic situation of the poor. While computing within LIMITS typically focuses on the future, Tomlinson et al.<sup>35</sup> note that our imagined “future” LIMITS scenarios may already exist today in the conditions in which poor communities live around the world. However, few studies within the ICTD literature consider global ecological, material, and energy limits. Most research is situated in resource-constrained contexts and assumes the constraints will be relaxed in the future after sufficient economic growth has occurred.<sup>12,15</sup> The only paper so far that explicitly makes the link between LIMITS and ICTD in an ICTD venue is Tomlinson et al.’s DEV paper, “Toward alternative decentralized infrastructures.”<sup>36</sup> The vacuum regarding the implications of phenomena such as climate change in the ICTD literature could be filled by a LIMITS perspective.

There is, however, a tension between economic development in poor countries—the focus of ICTD—and sustainability. As Herman Daly points out, the total resource footprint of the Global North and the Global South combined together must stay within the boundaries of a global steady state economy that is sustainable in the long run. To ameliorate the problem of un-

equal distribution of wealth and the consequent problem of poverty in the Global South, the Global North must shrink its resource footprint enough that countries in the Global South are afforded some space for necessary economic growth. However, everyone—North and South—must operate within some absolute global limits. The ethical argument for improving the quality of life of the poor is easy to make, but reducing the Global North’s consumptive (and exploitative) practices to afford the Global South opportunities to grow, especially in the face of mounting resource and climate pressures, remains an enormous challenge, and one computing should be cognizant of.

Despite differing perspectives, LIMITS and ICTD have much in common and potential for integration and collaboration.<sup>4</sup> For example, LIMITS work has studied the use of digital technology to design habitations in refugee camps,<sup>29</sup> problems of networking in rural populations in Zambia and Guatemala<sup>30</sup> and infrastructure in conditions of scarcity in Haiti.<sup>21</sup> While these are classic ICTD topics, the authors in each case considered ecological, material, and energy limits in their analyses, unlike typical ICTD studies. The papers engage models of scarcity, examining the cases as possible future global LIMITS scenarios. Drought, flooding, environmental disasters, infrastructure disruption, mass migration, and permanent settlement in refugee camps in low-resource environments are seen

as highly relevant to global futures, not just as problems that will be solved through economic growth.

### Computing Within Limits Workshops

LIMITS ideas have been developed through three workshops (2015–2017) convened by the LIMITS community (the latter two in cooperation with ACM). The first two were held at the University of California, Irvine, and the third at Westmont College in Santa Barbara, with funding from the two universities as well as from Facebook and Google. Participants came from institutions in Abu Dhabi, Canada, Hong Kong, Pakistan, Spain, Sweden, Switzerland, the U.K., and the U.S., consistent with the global nature of LIMITS concerns and research. The 2018 workshop was held in Toronto, co-located with the Fifth International Conference on Information and Communication Technology for Sustainability (ICT4S). Sparked by discussions at the workshops, LIMITS participants have co-authored several papers published in mainstream conferences and a research grant. The LIMITS workshop papers are available at [computingwithinlimits.org](http://computingwithinlimits.org)

### Three Key Principles

We propose three principles that can help frame computing research and practice in a way that is consistent with the ideas described in this paper and the literature we have surveyed.

*Question growth.* The industrialized world’s current economic system, capitalism, is predicated on growth. Economic growth has brought more than an order of magnitude rise in per capita income from \$3 a day in 1800 to \$100 in the early 2000s for most of Europe and North America.<sup>16</sup> However, despite such unprecedented prosperity, global income inequality is increasing. Wealth is accumulating in the hands of fewer and fewer astoundingly rich persons.<sup>22</sup> Poverty is widespread. Such social dysfunction, along with the burdens on ecosystems produced by economic activity,<sup>28,32,38</sup> suggest we must rethink the growth paradigm. The ubiquity and power of computing make it well positioned to act as an agent of change to influence proposals for transformative economic systems




and methods of governance. While discussion of specific proposals is beyond the scope of this article, we point to the work of, for example, Daniel O'Neill,<sup>18</sup> Peter Frase,<sup>9</sup> and Tim Jackson<sup>13</sup> as thoughtful responses to current problems that might inform the ways we practice computing.

Daly's notion of promoting development rather than economic growth suggests a sound mechanism for moving civilization forward, deploying our creativity and capacity for innovation in LIMITS-compliant ways. An economy that demands endless growth entails a cycle of consequences that must be interrupted if we are to address massive problems such as climate change and resource depletion.<sup>20</sup> Exploring relations between computing and the economy will be an important direction for future development of the computing community and a considerable challenge.


Currently, the implicit organizing framework for a great deal of computing work puts a focus on increasing the proximate financial value of companies. Even when particular products, from a narrow perspective, are seeking to make people's lives better through new technology, these products are typically embedded in a rapid churn of objects and services that foster runaway consumption.<sup>23,27</sup> By shifting the explicit focus, first and foremost, to the pursuit of long-term well-being, we may finally escape the growth paradigm and build systems that more effectively lead to sustainable improvements in the quality of life for humans and other species.

To make this principle actionable, we encourage researchers and practitioners to consider whether their work is a) reliant on growth, b) seeking to make growth happen, c) contributing to growth. We encourage those working in computing to build systems and envision worlds that are neither reliant on nor contributing to runaway growth. A number of existing LIMITS relevant papers have addressed this principle.<sup>24,31,35</sup>

*Consider models of scarcity.* Clever technological fixes may help us defer catastrophes for some time, but not indefinitely, and especially not if events such as wildfires, hundred-year storms, and Category 5 hurricanes



**We encourage those working in computing to build systems and envision worlds that are neither reliant on nor contributing to runaway growth.**



become more numerous and more powerful as outcomes of global environmental changes. Our track record of being prepared for dealing with unpredictable catastrophic events is not encouraging. We would benefit from seriously considering LIMITS-related scenarios rather than blithely denying their possibility or treating their foreshocks as isolated incidents. Engaging with these difficult scenarios before they occur, rather than only in their aftermath, will help us evaluate our level of preparedness and perhaps prevent certain undesirable future scenarios from happening.<sup>21</sup>

To speak of LIMITS-scenarios only in the future tense, however, is misleading. These events are here now, as several climate-related catastrophes in the U.S. and Europe have shown, even during the writing of this article. Science fiction author William Gibson famously said, "The future is already here—it's just not evenly distributed." We see this future currently on display in places such as Flint, Michigan where toxic wastes have poisoned the water supply. It is thus possible to frame LIMITS scenarios (including, for example, heat waves, drought, rising sea levels, and floods) not in terms of random irregularities or threats that might afflict us in the future, but in terms of an increasing incidence of phenomena arising from intensive economic activity.

A concrete research strategy is to develop case studies of current changes that may model futures of relative scarcity. For example, a study of the continuing impact of the 2010 earthquake in Haiti found that the regrowth of infrastructures was occurring in a more distributed fashion than would be typical for countries with more resources.<sup>21</sup> Distribution networks for clean water, electricity, Internet, and gasoline were severely damaged in the earthquake. Corporate and government responses were hampered by political and financial obstacles. In many cases, survivors themselves began to rebuild the infrastructures in a bottom up manner. For example, large private water tanks were installed on local properties. Wealthier residences allowed adjacent poorer households to tap into power lines via jerry-rigged extension cords without paying for the service—a generous if somewhat precarious arrangement.




Such a re-arrangement certainly went against existing building codes, but recognized the low cost of alleviating some resource deprivation in exchange for neighborhood stability.

A case study such as this can be generative by revealing opportunities for developed and less developed regions to transfer technologies and schemes of sociotechnical organization that present a different set of economic incentives for actors. Being aware of the wide diversity of current and future potential contexts in which humans may find themselves, more than a few of which are characterized by scarcity, may help computing researchers and practitioners design technology that promotes global well-being.


Several other LIMITS-relevant papers have focused on aspects of this principle, including work found in Refs.<sup>4,14,29,30,40</sup>

*Reduce energy and material consumption.* Sticking to the dominant narrative of growth is riskier than just making a bad guess. It is dangerous because it creates a possibility that we will reach a point at which resources have precipitously dwindled and we may not have enough remaining resources to make the necessary corrections to avert catastrophic outcomes. Therefore, it is important to acknowledge that computing uses energy and material resources. If, as we have argued, these resources are declining, a threshold that LIMITS research should meet is that it is worth the resources it consumes. Put another way, LIMITS research, once applied, should reduce energy expenditures and material consumption. This reduction is difficult to assess, but not something we can sidestep.

More broadly speaking, attempts to limit resource usage in any human system are notoriously challenging. Most of us are well aware of the problems of CO<sub>2</sub> emissions, but less aware of more subtle dynamics such as the Jevons paradox, that is, that more efficient technologies often encourage greater use of a resource, reducing or eliminating savings. A more efficient gas engine may reduce fuel consumption by half, but stimulate more than twice as much driving (as well as more cars). A more efficient cryptocurrency mining chip effectively increases electricity consumption through competitive pres-



**LIMITS research, once applied, should reduce energy expenditures and material consumption. This reduction is difficult to assess, but not something we can sidestep.**



sure. Mitigating the Jevons paradox requires creative approaches that may include substitution of goods by services and dematerialization, for example, by virtualization.<sup>11</sup> Such changes have the potential to entail a drop in absolute consumption, although so far, most approaches have tended to focus on increasing efficiency, which may or may not result in absolute reductions.<sup>13</sup> However, there is scope for significant change; for example, the energy costs of a virtual meeting that transmits data to a large number of remote participants is tiny compared to the energy cost of a single airplane trip for a single participant. The energy needed for data transmission is decreasing at a fast pace, unlike the energy costs of air travel. Aslan et. al.<sup>2</sup> estimate that data transmission costs decrease by 50% every two years.

Accounting for resource use must be done thoughtfully, with long-term goals in mind, in view of the big picture. There is justification for spending resources during a time of relative abundance to prepare for a future of scarcity.<sup>12</sup> Not all investments need to pay off immediately. There is a place for experimenting when we don't know for sure if savings will be accrued. But such experimentation should fail fast, and have a plausible hope of saving resources. In this regard, we need to be cognizant of the power of capital markets in deciding what is a success and what is a failure. While markets are very good at optimizing the delivery of the goods and services that they incentivize, they tend not to be organized in such a way that promotes long-term returns or incorporates the costs of the externalities that push limits. Structural changes such as cap-and-trade markets, taxes, fees, rationing, and quotas are needed, in concert with technological changes, to address these issues.

Another key approach involves finding energy savings through disintermediation, that is, the process of leveraging technology to supplant "middleman" actors in resource chains.<sup>25</sup> Traditionally, in the absence of information technology, such middlemen provided value and extracted costs by creating markets and distribution centers for goods. For example, systems to directly connect small-scale worker/producer owned facilities


with consumers could be of value in a new economy. Such simplification is responsive to Tainter's argument that increasing complexity leads to increasing burdens for systems which at some point they cannot bear.<sup>33</sup> Technologies that provide services while reducing complexity at the same time, square conceptually with what we know from the historical and archaeological record about the relationship between increasing societal complexity and eventual societal decline. This and other efforts at disintermediation<sup>3,36</sup> could help reduce energy and material consumption.

## Conclusion

While we do not know for certain what the future holds, scientists from disciplines such as climate science and ecology have made evidence-based predictions about directions the future will likely take if current trends continue. However, what many computing researchers and practitioners do in practice is to assume there is only one possible likely future—that current trajectories of increased growth and consumption will continue. The burden of our message in this article is that science is telling us the kinds of growth we have recently experienced are unsustainable. Consequently, we believe the field of computing should be paying serious attention to futures in which we encounter planetary limits.

LIMITS thinking emphasizes incentivizing long-term returns. It seeks to align its efforts with the scientific disciplines documenting global transformations through climate change and numerous other global effects. LIMITS seeks to explore ways that computing may support long-term well-being. We see significant cause for concern in many science-based projections of the future, and we want to enable our work to be relevant and useful with respect to these potential realities.

## Acknowledgments

The authors thank several anonymous reviewers for their cogent comments, as well as the entire LIMITS community for helping shape the ideas in this article. This material is based in part on work supported by the NSF under Grants No. CCF-1442749 and IIS-0644415. 

## References

- Anderson, J. et al. Far far away in Far Rockaway: Responses to risks and impacts during Hurricane Sandy through first-person social media narratives. In *Proceedings of the 13<sup>th</sup> International Conference on Information Systems for Crisis Response and Management*, (2016).
- Aslan, J., Mayers, K., Koomey, J. and France, C. Electricity intensity of Internet data transmission: Untangling the estimates. *J. Industrial Ecology*, (2017).
- Blevis, E. Sustainable interaction design: Invention & disposal, renewal and reuse. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, (2007), ACM, 503–512.
- Chen, J. Computing within limits and ICTD. In *Proceedings of the 1<sup>st</sup> Workshop on Computing within Limits*, (2015), ACM.
- Chen, J. A strategy for limits-aware computing. In *Proceedings of the 2<sup>nd</sup> Workshop on Computing within Limits*, (2016), ACM.
- Costanza, R. *Ecological Economics: The Science and Management of Sustainability*. Columbia University Press, 1992.
- Daly, H. *Steady State Economy*. Island Press, 1977.
- DiSalvo, C., Sengers, P. and Brynjarsdóttir, H. Mapping the landscape of sustainable HCI. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, (2010), ACM, 1975–1984.
- Frase, P. *Four Futures*. Verso, London, U.K., 2016.
- Hazas, M. and Nathan, L. (Eds.). *Digital Technology and Sustainability: Engaging the Paradox*. Routledge, New York, 2018.
- Hilty, L.M. and Aebischer, B. (Eds.). *ICT Innovations for Sustainability*. Vol. 310. Springer International Publishing, 2015.
- Houston, L. and Jackson, S. Caring for the next billion mobile handsets: Opening proprietary closures through the work of repair. In *Proceedings of the 8<sup>th</sup> International Conference on Information and Communication Technologies and Development*, (2016), ACM.
- Jackson, T. *Prosperity without Growth*. Routledge, London, U.K., 2017.
- Jang, E., Johnson, M., Burnell, E. and Heimler, K. Unplanned obsolescence: Hardware and software after collapse. In *Proceedings of the 3<sup>rd</sup> Workshop on Computing within Limits*, (2017), ACM.
- Masinde, M., Bagula, A. and Muthama, N. The role of ICTs in downscaling and up-scaling integrated weather forecasts for farmers in sub-Saharan Africa. In *Proceedings of the 5<sup>th</sup> International Conference on Information and Communication Technologies and Development*, (2012), ACM, 122–129.
- McCloskey, D. *Bourgeois Dignity*. University of Chicago Press, Chicago, 2010.
- Murugesan, S. Harnessing green IT: Principles and practices. *IT Professional* 10, 1 (2008), 24–33.
- O'Neill, D. Measuring progress in the degrowth transition to a steady state economy. *Ecological Economics* 84 (2011), 1–11.
- Palen, L., Starbird, K., Vieweg, S. and Hughes, A. Twitter-based information distribution during the 2009 Red River Valley flood threat. *Bulletin of the American Society for Information Science and Technology* 36, 5 (2010), 13–17.
- Pargman, D. and Raghavan, B. Rethinking sustainability in computing: From buzzword to non-negotiable limits. In *Proceedings of the 8<sup>th</sup> Nordic Conference on Human-Computer Interaction*, (2014), ACM, 638–647.
- Patterson, D. 2015. Haitian resiliency: A case study in intermittent infrastructure. In *Proceedings of the 1<sup>st</sup> Workshop on Computing Within Limits*. ACM, 111–117.
- Piketty, T. *Capital in the 21<sup>st</sup> Century*. Belknap Press, Cambridge, MA, 2014.
- Preist, C., Schien, D. and Blevis, E. Understanding and mitigating the effects of device and cloud service design decisions on the environmental footprint of digital infrastructure. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 1324–1337.
- Raghavan, B. and Ma, J. Networking in the long emergency. *Proceedings of the 2<sup>nd</sup> ACM SIGCOMM Workshop on Green Networking*, (2011), ACM, 37–42.
- Raghavan, B. and Pargman, D. Means and ends in human-computer interaction: Sustainability through disintermediation. In *Proceedings of the 2017 SIGCHI Conference on Human Factors in Computing Systems*. ACM, 786–796.
- Ramchurn, S., Vytelingum, P., Rogers, A. and Jennings, N. Putting the 'smarts' into the smart grid: A grand challenge for artificial intelligence. *Commun. ACM* 55, 4 (Apr. 2012), 86–97.
- Remy, C. and Huang, E. Addressing the obsolescence of end-user devices. In *ICT Innovations for Sustainability*. Springer, New York, 2014.
- Rockström, J. et al. A safe operating space for humanity. *Nature* 461 (2009), 472–475.
- Sabie, S., Chen, J., Abouzied, A., Hashim, F., Kahlon, H. and Easterbrook, S. Shelter dynamics in refugee and IDP camps: Customization, permanency, and opportunities. In *Proceedings of the 3<sup>rd</sup> Workshop on Computing Within Limits*, (2017), ACM, 11–20.
- Schmitt, P. and Belding, E. Navigating connectivity in reduced infrastructure environments. In *Proceedings of the 2<sup>nd</sup> Workshop on Computing Within Limits*. ACM.
- Silberman, M.S. and Tomlinson, B. Precarious infrastructure and postapocalyptic computing. In *Proceedings of CHI 2010 Workshop on Examining Appropriation, Re-use, and Maintenance for Sustainability*.
- Steffen, W. et al. Planetary boundaries: Guiding human development on a changing planet. *Science* 347 (2015), 6223.
- Tainter, J. *The Collapse of Complex Societies*. Cambridge University Press, Cambridge, U.K., 1990.
- Thrun, S. et al. Stanley: The robot that won the DARPA Grand Challenge. *J. Field Robotics* 23, 9 (2006), 661–692.
- Tomlinson, B., Six Silberman, M., Patterson, D., Pan, Y., and Blevis, E. Collapse informatics: augmenting the sustainability & ICT4D discourse in HCI. In *Proceedings of the 2012 SIGCHI Conference on Human Factors in Computing Systems*. ACM, 655–664.
- Tomlinson, B., Nardi, B., Patterson, D., Raturi, A., Richardson, D., Saphores, J.-D. and Stokols, D. Toward alternative decentralized infrastructures. In *Proceedings of the 2015 Annual Symposium on Computing for Development*. ACM, 33–40.
- Vardi, M. The financial meltdown and computing. *Commun. ACM* 52, 9 (Sept. 2009), 5.
- Vitousek, P., Mooney, H., Lubchenco, J. and Melillo, J. Human domination of Earth's ecosystems. *Science* 277 (1997), 494–499.
- Weiser, M. The computer for the 21<sup>st</sup> century. *Scientific American* 265, 3 (1991), 94–104.
- Wong, J. Prepare for descent: Interaction design in our new future. In *Proceedings of the 2009 CHI Workshop on Defining the Role of HCI in the Challenges of Sustainability*.

**Bonnie Nardi** (nardi@ics.uci.edu) is a professor in the Department of Informatics at University of California, Irvine, USA.

**Bill Tomlinson** (bill.tomlinson@vuw.ac.nz) is a professor in the Department of Informatics at the University of California, Irvine, USA, and an adjunct professor in the School of Information Management, Victoria University of Wellington, New Zealand.

**Donald J. Patterson** (dpatterson@westmont.edu) is a professor in the Department of Math and Computer Science at Westmont College, Santa Barbara, CA, USA.

**Jay Chen** (jay.chen@nyu.edu) is an assistant professor in the Department of Computer Science at NYU Abu Dhabi, U.A.E.

**Daniel Pargman** (pargman@kth.se) is an associate professor in the Department of Media Technology and Interaction Design at KTH Royal Institute of Technology, Stockholm, Sweden.

**Barath Raghavan** (barath.raghavan@usc.edu) is an assistant professor of computer science at the University of Southern California, Los Angeles, USA.

**Birgit Penzenstadler** (Birgit.Penzenstadler@csulb.edu) is an assistant professor in the Department of Computer Engineering and Computer Science at California State University, Long Beach, USA.

# research highlights

---

P. 95

**Technical  
Perspective**  
**A Control Theorist's  
View on Reactive  
Control for  
Autonomous Drones**

By John Baillieul

P. 96

**Fundamental Concepts  
of Reactive Control for  
Autonomous Drones**

By Luca Mottola and Kamin Whitehouse

---

P. 105

**Technical  
Perspective**  
**The Future of MPI**

By Marc Snir

P. 106

**Enabling Highly Scalable  
Remote Memory Access  
Programming with  
MPI-3 One Sided**

By Robert Gerstenberger, Maciej Besta, and Torsten Hoefler

---



# Technical Perspective

## A Control Theorist's View on Reactive Control for Autonomous Drones

By John Baillieul

IN THE LATE 1990s, at about the time as an upsurge of interest among theorists in real-time control in which feedback loops were closed through rate-limited communication channels, the Bluetooth communication standard was introduced to enable “local area networks of things.” Various research groups, including my own, became interested in implementing feedback control using Bluetooth channels in order to evaluate the design principles that we and others had developed for communication-limited real-time systems. With device networks taking on ever increasing importance, our Bluetooth work was part of an emergent area within control theory that was aimed at systems using existing infrastructure rather than systems of sensors, actuators, and data links that were co-optimized to work together to meet performance objectives.


The main challenge of using infrastructure that was designed for purposes other than real-time applications was that none of the infrastructure-optimized computation and communication protocols are well suited to closing feedback loops of control systems. The work of Mottola and Whitehouse is somewhat along these lines—with the infrastructure in this case being the control logic and feedback control algorithms that are found on popular UAV autopilot platforms such as Ardupilot, Pixhawk, the Qualcomm Snapdragon, and the now discontinued OpenPilot. Several such autopilots are target platforms for the software described in the following paper.

The authors introduce the notion of “reactive control” in which an autopilot's control logic is run only intermittently based on whether readings from sensors indicate a need to react to something in the environment. Thus, they employ the off-the-shelf existing

control infrastructure, but only when their algorithms decide it is needed. For Mottola and Whitehouse, reactive control is distinguished from the more common approach to motion control that they refer to as “time-triggered” control. The meaning of the terminology is a bit different from the way it is used in most current work on mobile robot control where the term “reactive control” is used to distinguish fast, low-level, sensor-driven loops from slower “deliberative” control that involves path planning or goal seeking navigation. The deliberative parts of motion control involve high-level decisions and choices of ways to achieve an overall objective—say, obtaining food in the case of animals or finding areas of high concentration of a chemical species for an extremum-seeking robot. Reactive control in the robotic literature normally involves processing real-time streams of sensory data to guide low-level motor response to follow a preplanned path or a path created in the deliberative layer. There is always more urgency in executing the reactive layer of a control implementation, but a balance of reactive and deliberative is essential for achieving robot autonomy.

Reactive control in the following paper involves a protocol for determining when sensor readings call for the autopilot's control to function. Whereas classical feedback control corrects for deviations from a setpoint or desired trajectory at every tick of a system clock, reactive control in their paper takes control action only when a sensor input at a clock reading differs “significantly” from the previous reading. One of the contributions of this work is an algorithmic approach to deciding when sensor-reading differences are “significant.” The authors use a probabilistic logistic regression approach to decide when a sensor reading requires reaction.

Throughout flight experiments, the parameters of the logistic-based decision rule are tuned with the aim of minimizing false positive and—more importantly—false negative assessments of the significance of sensor reading differences. Although the concept of “act-only-when-necessary” is simple and intuitive, the fact there are multiple sensors and actuators means there are very complex data dependencies that must be accounted for in real-time execution.

How well does it work? The authors deserve a great deal of credit for meticulous testing. They have logged more than 260 hours of flight testing and experimental benchmarking on three different flight vehicles—a quadcopter, a hexacopter and a challenging tricopter. They also report work with three different off-the-shelf autopilot implementations. The applications to which reactive flight control is best suited are those where setpoints do not change dramatically over the path; for example, hovering and following relatively straight paths as opposed to, say, aerial acrobatics. Nevertheless, the experiments show convincingly that the approach can handle challenging situations, particularly in outdoor flights where wind gusts provide significant disturbances to which the control system must react. A thought that occurred to me after reading the paper is that animal movements are guided by neurological circuits that must continually refocus attention on the most relevant features in the environment. The current work may open a promising new thrust toward understanding such aspects of biological motor control. 

**John Baillieul** is Distinguished Professor of Engineering at Boston University. He is past editor-in-chief of the *IEEE Transactions on Automatic Control* and also past editor-in-chief of the *SIAM Journal of Control and Optimization*.

Copyright held by author.

# Fundamental Concepts of Reactive Control for Autonomous Drones

By Luca Mottola and Kamin Whitehouse

## Abstract

Autonomous drones represent a new breed of mobile computing system. Compared to smartphones and connected cars that only opportunistically sense or communicate, drones allow motion control to become part of the application logic. The efficiency of their movements is largely dictated by the low-level control enabling their autonomous operation based on high-level inputs. Existing implementations of such low-level control operate in a time-triggered fashion. In contrast, we conceive a notion of reactive control that allows drones to execute the low-level control logic only upon recognizing *the need to*, based on the influence of the environment onto the drone operation. As a result, reactive control can *dynamically adapt* the control rate. This brings fundamental benefits, including more accurate motion control, extended lifetime, and better quality of service in end-user applications. Based on 260+ hours of real-world experiments using three aerial drones, three different control logic, and three hardware platforms, we demonstrate, for example, up to 41% improvements in motion accuracy and up to 22% improvements in flight time.

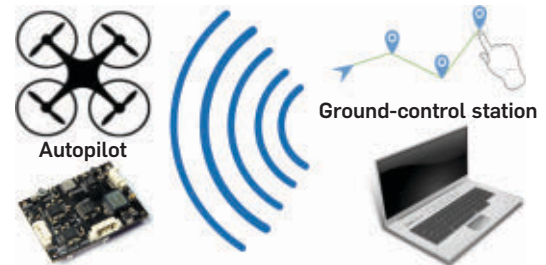
## 1. INTRODUCTION

Robot vehicle platforms, often called “drones,” offer exciting new opportunities for mobile computing. While many mobile systems, such as smartphones and connected cars, simply respond to device mobility, drones allow computer systems to actively control device location. Such a feature enables interactions with the physical world to happen in new ways and with new-found scale, efficiency, or precision.<sup>4,8,18</sup>

**Autopilots.** Figure 1 schematically illustrates the hardware and software components in modern drone platforms. Key to their operation is the *autopilot* software implementing the low-level motion control. The control loop processes high-level commands coming from a Ground-Control Station (GCS) as well as various sensor inputs, such as, accelerations and Global Positioning System (GPS) coordinates, to operate actuators such as electrical motors that set the 3D orientation of the drone.

Together with the mechanical design, the *autopilot* software is crucial to determine a drone’s performance along a number of essential metrics. For example, the low-level control directly influences the quality of the shots when using drones for imagery applications.<sup>17, 18</sup> Further, it is partly responsible for the overall energy efficiency, as a

**Figure 1. Hardware and software components in modern drone platforms. Users configure high-level mission parameters at the ground-control station (GCS), whereas the autopilot software implements the low-level motion control aboard the drone.**



drone’s lifetime is often a result of how streamlined is the autopilot operation.<sup>5,24</sup>

Unsurprisingly, most existing autopilots employ Proportional-Integral-Derivative (PID)<sup>2</sup> designs. Processing is thus time-triggered: every  $T$  time units, sensors are probed, control decisions are computed, and commands are sent to the actuators. Such a deterministic operation simplifies implementations and allows designers to directly rely on a vast body of existing literature.<sup>2</sup>

**Reactive control.** Based on a handful of *key observations*, a fundamental *leap of abstraction*, and an unconventional use of recent *advances in programming languages*, we conceive a notion of reactive control that allows autopilots to significantly improve a drone’s performance in both motion accuracy and energy consumption. Rather than periodically triggering the control logic, we only run the control logic upon recognizing *the need to*. Depending on the influence of the environment onto the drone operation, for example, due to wind gusts or pressure gradients, control may run more or less frequently, regardless of the the fixed rate of a corresponding time-triggered implementation. As a result, reactive control *dynamically adapts* the control rate.

Reactive control yields several advantages, including more timely and adaptive control decisions leading to improved motion accuracy and energy efficiency. As it

The original version of this paper is entitled “Reactive Control of Autonomous Drones” and was published in *Proceedings of the 14<sup>th</sup> ACM International Conference on Mobile Systems, Applications, and Services*, Singapore, June 2016.

exclusively works in software, reactive control also requires no hardware modifications. We provide concrete evidence of these benefits across different aerial drone applications, based on 260+ hours of test flights in three increasingly demanding environments, using a combination of three aerial drones, three autopilot software, and three embedded hardware platforms. Our results indicate, for example, that reactive control obtains up to 41% improvements in the accuracy of motion, and up to a 22% extension of flight times.

The remainder of the paper unfolds as follows. Section 2 provides the necessary background, elaborates on the fundamental intuitions behind reactive control, and outlines the issues that are to be solved to make it happen. Section 3 describes the specific techniques we employ to address these issues. Section 4 reports on the performance of reactive control compared with traditional time-triggered implementations, whereas Section 5 studies the impact of reactive control in a paradigmatic end-user application. We conclude the paper in Section 6 by discussing our current work towards obtaining official certifications to fly drones running reactive control over public ground.

## 2. BUILDING UP TO REACTIVE CONTROL

Reactive control relies on concepts and techniques germane to statistics, embedded software, programming languages, control, and low-power hardware. In the following, we try and smooth the waters for the readers by walking them through the characteristics of target platforms, the key observations leading to reactive control, and the issues that are to be solved to concretely realize it.

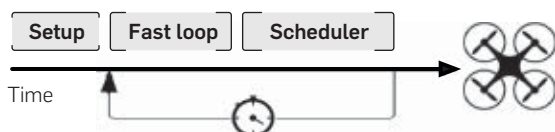
### 2.1. Autopilots

Drones can be regarded as a cruder form of modern robotics.<sup>9</sup> The high-level inputs coming from the GCS may be a waypoint or a trajectory. Autopilots implement the low-level control in charge of translating these inputs into commands for the drone actuators.

Ardupilot ([goo.gl/x2CHyM](http://goo.gl/x2CHyM)) is an example autopilot implementation, providing reliable low-level control for aerial drones and ground robots. The project boasts a large on-line community and is at the basis of many commercial products.

**Software.** Figure 2 shows the execution of Ardupilot’s low-level control loop, split in two parts. The *fast loop* only includes critical motion control functionality. The

**Figure 2. Ardupilot’s low-level control loop. The time for a single iteration of the loop is split between fast loop, which only includes critical motion control functionality, and an application-level scheduler that runs non-critical tasks.**



time left from the execution of *fast loop* is given to an application-level *scheduler* that distributes it among non-critical tasks that may not always execute, such as logging. The scheduler operates in a *best-effort* manner based on programmer-provided priorities. Many autopilots share similar designs.<sup>7</sup>

Initially, *fast loop* blocks waiting for a new value from the Inertial Measurement Unit (IMU). This provides an indication of the forces the drone is subject to, obtained by combining the readings of accelerometers, gyroscopes, magnetometers, and barometer. Once a new value is available, IMU information is combined with GPS readings to determine how the motors should operate to minimize the error between the desired and actual pitch, roll, and yaw, shown in Figure 3. Multiple PID controllers inside *fast loop* are used to this end.

In Ardupilot as well as the vast majority of autopilots, the control rate is statically set to strike a reasonable trade-off between motion accuracy and resource consumption, based on a few “rules of thumbs.”<sup>6,25</sup> For example, Ardupilot runs at a *fixed* 400Hz on the hardware we describe next. This rate is not necessarily the maximum the hardware supports. The 400Hz of Ardupilot, for example, are thought to leave enough room—on average—to the scheduler. In short bursts, control may run much faster than 400Hz, as long as some processing time is eventually allocated to the scheduler.

**Hardware.** Autopilots typically run on resource-constrained embedded hardware, for reasons of size and cost. A primary example is the Pixhawk family of autopilot boards ([goo.gl/wU4fmk](http://goo.gl/wU4fmk)), which feature a Cortex M4 core at 168MHz and a full sensor array for navigation, including a 16-bit gyroscope, a 14-bit accelerometer/magnetometer, a 16-bit 3-axis accelerometer/gyroscope, and a 24-bit barometer. Most often, at least a sonar and a GPS are added to provide positioning and altitude information, respectively.

Interestingly, the sensors on Pixhawk have similar capabilities as those on modern mobile phones. In fact, many argue that without the push to improve sensors due to the rise of mobile phones, drone technology would have not emerged.<sup>9</sup> Such sensors support energy-efficient high-frequency sampling and often provide interrupt-driven modes to generate a value upon verifying certain conditions. The ST LSM303D mounted on the Pixhawk, for example, can be programmed to generate an Serial Peripheral Interface (SPI) interrupt based on three thresholds. This is useful,

**Figure 3. Control based on raw, pitch, and yaw.**





for example, in human tracking applications for functionality such as fall detection.<sup>15</sup>

## 2.2. Intuition

Through our continuous work with drones as mobile computing platforms,<sup>16, 19</sup> we eventually noticed that the autopilots' PID controllers are mostly tuned so that it is the Proportional component to dictate the actual controller operation. The Derivative component can be kept to a minimum though a careful distribution of weights,<sup>6, 11</sup> whereas precise sensor calibration may spare the Integral component almost completely.<sup>6, 11, 22</sup>

As a result of this observation, we concluded that a simple relation exists between current inputs from the navigation sensors and the corresponding actuator settings. With little impact from the time-dependent Derivative and Integral components, and with the Proportional component dominating, small variations in the current sensor inputs likely correspond to small variations in the actuator settings. As an extreme case, as long as the sensor inputs do not change, the actuator settings should remain almost unaltered. In such a case, at least in principle, one may not run the control logic and simply retain the previous actuator settings.

Reactive control builds upon this intuition. We constantly monitor the navigation sensors to understand when the control logic does need to run as a function of the instantaneous environment conditions. These manifest as changes in the inputs of navigation sensors. If these are sufficiently significant to warrant a change in the physical drone behavior to be compensated, reactive control executes the control logic to compute new actuator settings. Otherwise, reactive control retains the existing configuration.

As we explain next, reactive control abstracts the problem of recognizing such significant changes in a way that makes it computationally tractable with little processing resources. Moreover, because of the aforementioned characteristics of sensor hardware on autopilot boards, monitoring the sensor readings at the maximum possible rate usually bears very little energy overhead. Reactive control, nonetheless, makes it possible to rely on the low-power interrupt-driven modes if available.

As a result, when sensor inputs change often, reactive control makes control run repeatedly, possibly at rates higher than the static settings of a time-triggered implementation. When sensor inputs exhibit small or no variations, the rate of control execution reduces, freeing up processing resources that may be needed at different times.

## 2.3. Challenge

Realizing reactive control is, however, non-trivial. Three issues are to be solved, as we illustrate in Section 3:

- 1) What is a “significant” change in the sensor input depends on several factors, including the accuracy of sensor hardware, the physical characteristics of the drone, the control logic, and the granularity of actuator output. We opt for a probabilistic approach to tackle this problem, which *abstracts* from all these

aspects by employing a form of auto-tuning of the conditions leading to running the control logic.

- 2) An indication for running the control logic may originate from different sensors, at different rates, and asynchronously with respect to each other. A problem is thus how to handle the possible interleavings. Moreover, not running the control loop for too long may negatively affect the drone's stability, possibly preventing to reclaim the correct behavior. We tackle these issues by only changing the *execution* of the control logic over time, rather than the logic itself.
- 3) Reactive control must run on resource-constrained embedded hardware. When implementing reactive control, however, the code quickly turns into a “callback hell”<sup>10</sup> as the operation becomes inherently event-driven. We experimentally find that, using standard languages and compilers, this negatively affects the execution speed, thus limiting the gains.<sup>7</sup> We design and implement a custom realization of Reactive Programming (RP) techniques<sup>3</sup> to tackle this problem.

The context where we are to address these issues shapes the challenge in unseen ways. For example, aerial drone demonstrations exist showing motion control in tasks such as throwing and catching balls,<sup>21</sup> flying in formation,<sup>23</sup> and carrying large payloads.<sup>14</sup> In these settings, the low-level control does not operate aboard the drone. At 100Hz or more, a powerful computer receives accurate localization data from high-end motion capture systems, runs sophisticated control algorithms based on drone-specific mechanical models expressed through differential equations, and sends actuator commands to the drones. Differently, we aim at improving the performance of mainstream low-level control on embedded hardware, targeting mobile sensing applications that operate in the wild.

On the surface, reactive control may also resemble the notion of event-based control.<sup>1</sup> Here, however, the control logic is often expressly redesigned for settings different than ours; for example, in distributed control systems to cope with limited communication bandwidth or unpredictable latency. This requires a different theoretical framework.<sup>1</sup> In contrast, we aim at re-using existing control logic, whose properties are well understood, and at doing so with little or no knowledge of its corresponding implementation and its parameter tuning. Different than event-based control, in addition, reactive control is mainly applicable only to PID-like controllers where the Proportional component dominates.

## 3. REACTIVE CONTROL

The key issues we discussed require dedicated solutions, as we explain here.

### 3.1. Conditions for reacting

**Problem.** It may seem intuitive that the more “significant” is a change in a sensor reading, the more likely is the necessity to run the control loop. Such a condition would indicate that

something just happened in the environment that requires the drone to react. However, what is a “significant” change in the sensor readings depends on several factors, including the accuracy of sensor hardware, the physical characteristics of the drone, and the actual control logic.

**Approach.** Our solution abstracts away from these aspects: despite the control logic is deterministic, we consider a change in the control output as a random phenomena. The input to this phenomena is the difference between consecutive samples of the same navigation sensor; the output is a binary value indicating whether the actuator settings need to change. If so, we need to run the control loop to compute the new settings. Therefore, an accurate statistical estimator of such random phenomena would allow us to take an informed decision on whether to run the control loop.

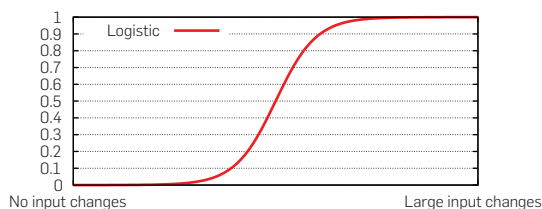
Among estimators with a *binary* dependent variable, *logistic regression*,<sup>12</sup> shown in Figure 4 in its general form, closely matches this intuition. For small changes in the sensor inputs, the probability of changes in the actuator settings is small. When changes in sensor inputs are large, a change in the actuator settings becomes (almost) certain. It also turns out it is possible estimate the parameters shaping the curve of Figure 4 efficiently, because logistic regression allows one to employ traditional estimators, such as least squares.<sup>12</sup>

**Operation.** We employ one logistic regression model per navigation sensor. Given a change in the sensor readings, we compute the probability that the change corresponds to new control decisions, according to a corresponding logistic regression model. If this is greater than a threshold  $P_{run}$ , we execute the control logic, with all other inputs set to the most recent value; otherwise, we maintain the earlier output to the actuators.

This approach assumes that changes in a sensor’s inputs at different times are statistically independent. This is justified because the time-dependent I, D components of the PID controllers bear little influence in our setting, as discussed earlier. Moreover, maintaining the earlier output to the actuators is possible only as long as the control set-point does not change in the mean time. This is most often the case when drones hover or perform waypoint navigation, but rarely happens in applications such as aerial acrobatics, where this approach would probably be inefficient.

Parameter  $P_{run}$  offers a knob to trade processing resources with the tightness of control. Large values of  $P_{run}$  spare a significant fraction of control executions.

**Figure 4. Example logistic function.**



However, the drone may require drastic corrections whenever the control loop does run; in a sense, motion becomes more “nervous.” Small values of  $P_{run}$  limit the processing gains. However, control runs more often, ensuring the drone operates smoothly. We demonstrate that gains over time-triggered control are seen for many different settings of  $P_{run}$ ,<sup>7</sup> therefore, tuning this parameter is typically no major issue.

**Run-time<sup>a</sup>.** The question is now how to realize the functionality above at run-time, and especially how to gather the data required to tune the logistic regression models. To that end, we initially run the control loop at fixed rate for a predefined limited time, tracking whether the actuator settings change. This gives us an initial data set to employ least square estimators to compute the parameters of logistic regression. From this point on, reactive control kicks in and drives the execution of the control logic based on whether the probability of new actuator settings, according to the logistic regression models, surpasses  $P_{run}$ .

False positives may occur when logistic regression triggers the execution of the control logic, yet the newly computed actuator settings stay the same. In this case, the change in the sensor reading is added to the data set initially used for tuning the regression models. The least square estimation repeats throughout the execution, as part of the best-effort *scheduler* part of the autopilot control loop, shown in Figure 2, taking false positives into account. Such a simple form of *auto-tuning*<sup>25</sup> progressively improves the estimation accuracy over time. We discuss the case of false negatives next.

### 3.2. Dealing with time

**Problem.** PID controllers used in autopilots are conceived under the assumption that sensors are sampled almost simultaneously and at a fixed rate. In reality, the time of sampling, and therefore of possibly recognizing the need to execute the control loop, is not necessarily aligned across sensors. Drastic changes in the sensor inputs may also be correlated. For example, when the accelerometers record a sudden increase because of a wind gust, a gyroscope also likely records significant changes. A traditional implementation would process these inputs together.

**Approach.** We take a conservative approach to address these issues. Based on the sampling frequency of every sensor in the system, we compute the system’s *hyperperiod* as the smallest interval of time after which the sampling of all sensors repeats. Upon recognizing first the conditions requiring the execution of the control loop, we wait until the current hyperperiod completes. This allows us to “accumulate” all inputs on different sensors, giving the most up-to-date inputs to the control logic at once.

Moreover, we need to cater for situations where false negatives happen in a row, potentially threatening

<sup>a</sup> Note that this design considers the initial drone execution as representative of the rest of the flight. Should this not be the case, a fail-over mechanism kicks in that recomputes the logistic regression parameters from scratch.

dependability. To address this issue, we run the control loop anyways at very low frequency, typically in the range of a few Hz. If such executions compute new actuator settings, the drone most likely applies some significant correction to the flight operation that causes reactive control to be triggered immediately after. If logistic regression originally indicated that the current changes in sensor readings did not demand to run the control logic, the current iteration is considered a false negative and feed back to the data set used for tuning the regression parameters. The next time the least square estimation executes, as explained above, these false negatives are also taken into account.

Note that the techniques hitherto described do *not* require one to alter the control logic itself; they solely drive its execution differently over time. The single iteration remains essentially the same as in a traditional time-triggered implementation. This means reactive control does not require to conceive a new control logic; the existing ones can be re-used provided an efficient implementation of such asynchronous processing is possible, as we discuss next.

### 3.3. Implementation

**Problem.** The control logic is implemented as multiple processing steps arranged in a complex multi-branch pipeline. Moreover, each such processing step may—in addition to producing an output *immediately* useful to take control decisions—update global state used *at a different iteration* elsewhere in the control pipeline.

Using reactive control, depending on what sensor indicates the need to execute the control loop, different slices of the code may need to run while other parts may not. The parts of the control pipeline that do not run at a given iteration, however, may need to run later because of new updates to global state. Thus, *any* arbitrary processing step—not just those directly connected to the sensors’ inputs—might potentially need to execute upon recognizing a significant change in given sensor inputs.

Employing standard programming techniques in these circumstances quickly turns implementations into a “call-back hell”.<sup>10</sup> This fragments the program’s control flow across numerous syntactically-independent fragments of code, hampering compile-time optimizations. We experimentally found that this causes an overhead that limits the benefits of reactive control.<sup>7</sup>

**Approach.** We tackle this issue using *RP*.<sup>3</sup> *RP* is increasingly employed in applications where it is generally impossible to predict when interesting events arrive.<sup>3</sup> It provides abstractions to automatically manage data dependencies in programs where updates to variables happen unpredictably. Consider for example:

```
a = 2;
b = 3;
c = a + b;
```

In sequential programming, variable *c* retains the value 5 regardless of any future update to variable *a* or *b*. Updating

*c* requires an explicit assignment following the changes in *a* or *b*. It becomes an issue to determine *where* to place such an assignment without knowing when *a* or *b* might change.

Using *RP*, one declaratively describes the data dependencies between variables *a*, *b*, and *c*. As variables *a* and *b* change, the value of *c* is constantly kept up-to-date. Then, variable *c* may be input to the computation of further state variables. The data dependencies thus take the form of an (acyclic) graph, where the nodes represent individual values, and edges represent input/output relations.

The *RP* run-time support traverses the data dependency graph every time a data change occurs, stopping whenever a variable does not change its value as a result of changes in its inputs. Any further processing would be unnecessary because the other values in the graph would remain the same. This is precisely what we need to efficiently implement reactive control; however, *RP* is rarely employed in embedded computing because of resource constraints.

**RP-Embedded.** We rely on a few key characteristics of reactive control to realize a highly efficient *RP* implementation. First, the data dependency graph encodes the control logic; therefore, its layout is known at compile-time. Second, the sensors we wish to use as initial inputs are only a handful. Finally, the highest frequency of data changes is known; for each sensors, we are aware or can safely approximate the highest sampling frequency.

Based on these, we design and implement *RP-EMBEDDED*: a C++ library to support *RP* on embedded resource-constrained hardware. *RP-EMBEDDED* trades generality for efficiency, both in terms of memory consumption and processing speed, which are limited on our target platforms. We achieve this by relying heavily on statically-allocated compact data structures to encode the data dependency graph. These reduce memory occupation compared with container classes of the *STD* library used in many existing C++ *RP* implementations, and improve processing speed by sparing pointer dereferences and indirection operation during the traversal. This comes at the cost of reduced flexibility: at run-time, the data dependency graph can only change within strict bounds determined at compile-time.

In addition, *RP-EMBEDDED* provides custom time semantics to handle the issues described in Section 3.2. The traditional *RP* semantics would trigger a traversal of the data dependency graph for any change of the inputs. With reactive control, however, the traversal caused by changes in a high-frequency sensor may be immediately superseded by the traversal caused by changes in another sensor within the same hyperperiod. The output that matters, however, is only the one produced by the second traversal.

To avoid unnecessary processing, *RP-EMBEDDED* allows one to characterize the inputs to the data dependency graph with their maximum rate of change. This information is used to compute the system’s hyperperiod. Every time a value is updated in the data dependency graph, *RP-EMBEDDED* waits for the completion of the current



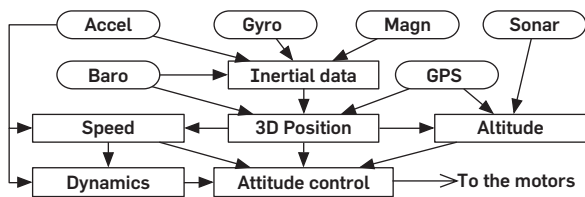
hyperperiod before triggering the traversal, which allows all inputs in the current hyperperiod to be considered together. To the best of our knowledge, such a semantics is not available in any RP implementation, regardless of the language.

**Using RP-Embedded.** Using RP-EMBEDDED for implementing reactive control requires to reformulate the implementation of the control logic in the form of a dependency graph. Sensor inputs remain the same as in the original time-triggered implementation, as well as control outputs directed to the actuators. The key modification is in processing changes in the sensor inputs: rather than immediately updating the inputs to the data dependency graph, we first check whether the corresponding logistic regression model would indicate the need to execute the control logic, as explained in Section 3.1.

Other than that, turning the control logic into a data dependency graph essentially boils down to a problem of code refactoring. Software engineering offers a wide literature on the subject.<sup>13</sup> Even in the absence of dedicated support, our experience indicates that the needed transformations can be implemented with little manual effort. Figure 5 shows the data dependency graph of the Ardupilot control loop for copters, which a single person on our team realized and tested in *three* days of work. Ardupilot is one of the most complex autopilot implementations. The other autopilots we test in Section 4 are simpler, and it took from one to two work days to refactor them.

<sup>b</sup> The OpenPilot project is currently discontinued. The community behind OpenPilot, however, forked a new project called LibrePilot ([goo.gl/KnZ3hG](http://goo.gl/KnZ3hG)) that shares most of the original codebase. Reactive control is thus equally applicable to LibrePilot, and we expect the performance to be similar to that we measure with OpenPilot.

**Figure 5. Ardupilot’s control loop for copters after refactoring to use RP-Embedded. Squashed rectangles indicate sensor inputs, squared rectangles indicate global state information.**



#### 4. PERFORMANCE<sup>b</sup>

We measure the performance of reactive control against the original Ardupilot. We also apply reactive control to two other autopilot implementations, namely OpenPilot and Cleanflight, and repeat the same comparison.

**Setup.** We use two custom drones, shown in Figure 6, and a 3D Robotics Y6 drone. The latter is peculiar as it is equipped with only three arms with two co-axial motor-propellers assemblies at each end, requiring a drastically different control logic.

We test three environments: (i) a 20×20m indoor lab, termed LAB, where localization happens using visual techniques; (ii) a rugby field termed RUGBY, using GPS; and (iii) an archaeological site in Aquileia (Italy) termed ARCH,<sup>16</sup> again using GPS. The sites exhibit increasing environment influence, from the mere air conditioning in LAB to average wind speeds of 8+ knots in ARCH. The variety of software, hardware, and test environments demonstrates the general applicability of reactive control.

We test OpenPilot and Cleanflight by replacing Ardupilot and the Pixhawk board on either the quadcopter or the hexacopter of Figure 6; however, only Ardupilot supports the Y6. The original time-triggered implementation of OpenPilot and Cleanflight resembles the design of Ardupilot shown in Figure 2, but the control logic differs substantially in both sophistication and tuning. Further, the autopilot hardware for OpenPilot and Cleanflight differ in processing capabilities and sensor equipment, compared with the Pixhawk. These differences are instrumental to investigate the general applicability of reactive control.

To study the accuracy of motion, we measure the *attitude error*, that is, the difference between the desired and actual 3D orientation of the drone. The former is determined by the autopilot as the desired setpoint, whereas the actual 3D orientation is recorded through the on-board sensors. Their difference is the figure the control logic aims at minimizing. If the error was constantly zero, the control would attain perfect performance; the larger this figure, the less effective is the autopilot. Measuring these figures in a minimally-invasive way requires dedicated hardware and software.<sup>7</sup>

To understand how the accuracy of flight control impacts the drone lifetime, we also record the *flight time* as the time between the start of an experiment and the time when the battery falls below a 20% threshold. For

**Figure 6. Aerial drones for performance evaluation.**



safety, most GCS implementations instruct the drone to return to the launch point upon reaching this threshold. In general, the lifetime of aerial drones is currently extremely limited. State of the art technology usually provides at most half an hour of operation. This aspect is thus widely perceived as a major hampering factor.

In the following, we describe an excerpt of the results we collect based on 260+ hours of test flights performing way-point navigation in the three environments.<sup>7</sup>

**Results.** As an example, Figure 7(a) shows the average improvements in pitch error; these are significant, ranging from a 41% reduction with Cleanflight in LAB to a 27% reduction with Ardupilot in ARCH. We obtain similar results, sometimes better, for yaw and roll.<sup>7</sup> Comparing this performance with earlier experiments, we confirm that it is the ability to shift processing resources in time that enables more accurate control decisions.<sup>7</sup> Not running the control loop unnecessarily frees resources, increasing their availability whenever there is actually the need to use them. In these circumstances, reactive control dynamically increases the rate of control, possibly beyond the pre-set rate.

Evidence of this is shown in Figure 8, showing an example trace that indicates the average control rate at second scale using Ardupilot and the hexacopter. In ARCH, reactive control results in rapid adaptations of the control rate in response to the environment influence, for example, wind gusts. On average, the control rate starts slightly below the 400Hz used in time-triggered control and slowly increases. An anemometer we deploy in the middle of the field confirms that the average wind speed is growing during this experiment.

In contrast, Figure 8 shows reactive control in LAB exhibiting more limited short-term adaptations. The average control rate stays below the rate of time-triggered control, with occasional bursts whenever corrections are needed to respond to environmental events, for example, when passing close to a ventilation duct. The trends in Figure 8 demonstrate reactive control’s adaptation abilities both in the short and long term.

Still in Figure 7(a), the improvements of reactive control apply to the Y6 as well; in fact, these are highest in a

given environment. This cannot be attributed to its structural robustness; the Y6 is definitely the least “sturdy” of the three. We conjecture that the different control logic of the Y6 offers additional opportunities to reactive control. A similar reasoning applies to Cleanflight, as shown in Figure 7(a). Being the youngest of the autopilot we test, it is fair to expect the control logic to be the least refined. Reactive control is still able to drastically improve the pitch error, by a 32% (37%) factor with the quadcopter (hexacopter) in ARCH.

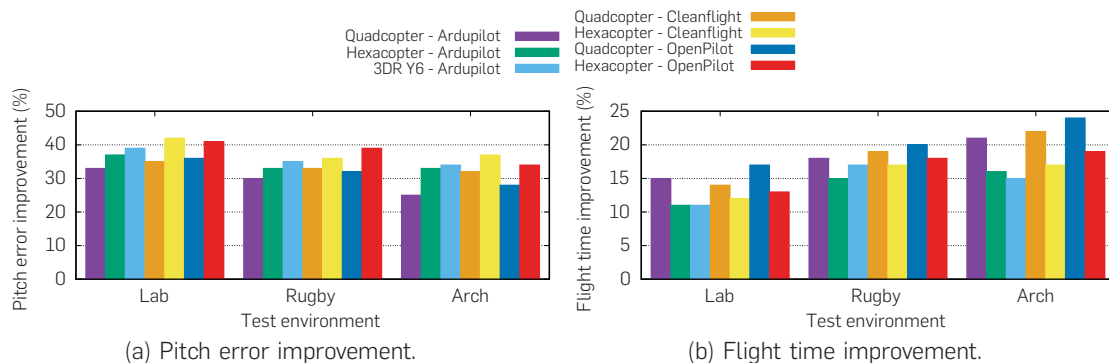
The improvements in attitude error translate into more accurate motion control and fewer attitude corrections. As a result, energy utilization improves. Figure 7(b) shows the results we obtain in this respect. Reactive control reaches up to a 24% improvement. This means flying more than 27min instead of 22min with OpenPilot in ARCH. This figure is crucial for aerial drones; the improvements reactive control enables are thus extremely valuable. Most importantly, these improvements are higher in the more demanding settings. Figure 7(b) shows that the better resource utilization of reactive control becomes more important as the environment is harsher. Similarly, the quadcopter shows higher improvements than the hexacopter. The mechanical design of the latter already makes it physically resilient. Differently, the quadcopter offers more ample margin to cope with the environment influence in software.

### 5. END-USER APPLICATIONS

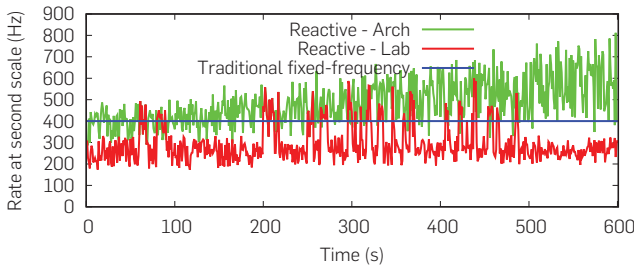
The performance improvements of reactive control reflect in more efficient operation of end-user drone applications ranging from 3D reconstruction to search-and-rescue.<sup>18</sup> The latter is a paradigmatic example of active sensing functionality, whereby data gathered by application-specific sensors guides the execution of the application logic, which includes here the drone movements. We build a prototype system to investigate the impact of reactive control in this kind of applications.

**System.** Professional alpine skiers are used to carry a device called Appareil de Recherche de Victimes en Avalanche (ARVA)<sup>20</sup> during their excursions. ARVA is nothing but a 457KHz radio transmitter expressly designed

Figure 7. Performance improvements with reactive control.



**Figure 8. Average rate of control at second scale in two example Ardupilot runs. Reactive control adapts the rate of control executions both in the short and long term, and according to the perceived environment influence.**



for finding people under snow. The device emits a radio beacon a rescue team can pick up using another ARVA receiver device. The latter essentially operates as a direction finding device, generating a “U-turn” signal whenever it detects the person carrying it starts moving away from the emitter. Modern ARVA devices are able to reach a 5m accuracy in locating an emitter under 10m of snow.<sup>20</sup>

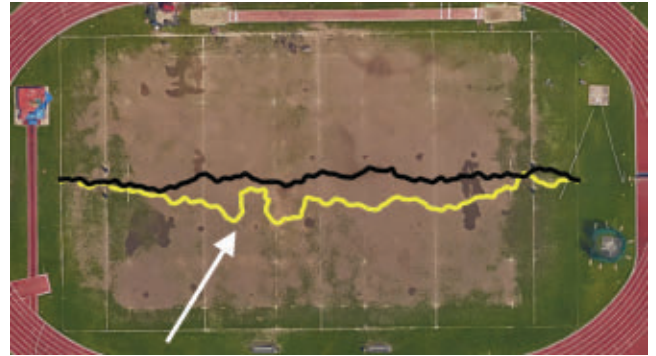
Our goal is to control the drone so that it reaches the supposed location of an ARVA emitter. To that end, we integrate a Pieps DSP PRO<sup>20</sup> ARVA receiver with the Pixhawk board. A custom PID controller aligns the drone’s yaw with the direction pointed by the on-board ARVA receiver. Roll and pitch, instead, are determined to fly at constant speed along the direction indicated by the ARVA receiver. Navigation is thus entirely determined by the ARVA inputs. We implement this controller both using reactive control by probing the ARVA device as fast as possible, and with time-triggered control at 400Hz, that is, the same as in the original time-triggered implementation of Ardupilot.

We place an ARVA transmitter at one end of RUGBY, and set up the quadcopter at 100m distance facing opposite to it. Even though GPS does not provide any inputs for navigation, we use it to track the path until the first time the ARVA device generates the “U-turn” signal. We compare the duration and length of the flight when using reactive or time-triggered control. We repeat this experiment 20 times in comparable environmental conditions.

**Results.** Reactive control results in a 21% (11%) reduction in the duration (length) of the flight, on average. Time-triggered control also shows higher variance in the results, occasionally producing quite inefficient paths. Figure 9 shows an example. The path followed by reactive control appears fairly smooth. In contrast, time-triggered control shows a convoluted trajectory at about one-third of the distance, where the yaw is almost  $\pm 90^\circ$  compared to the target.

The logs we collect during the experiment indicate that the reason for this behavior is essentially the inability of time-triggered control to promptly react. Probably because of a sudden wind gust, at some point the drone gains a lateral momentum. Time-triggered control is unable to react fast enough; a higher than 400Hz rate would probably be

**Figure 9. Example of ARVA-driven navigation when using reactive (black) and time-triggered control (yellow). Time-triggered control occasionally produces highly inefficient paths, whereas we never observe similar behaviors with reactive control.**



needed in this case, and the drone turns almost  $90^\circ$ . We never observe this behavior with reactive control, which better manages available processing resources against environment influences.

## 6. CONCLUSION AND OUTLOOK

Reactive control replaces the traditional time-triggered implementation of drone autopilots by governing the execution of the control logic based on changes in the navigation sensors. This allows the system to dynamically adapt the control rate to varying environment dynamics. To that end, we conceived a probabilistic approach to trigger the execution of the control logic, a way to carefully regulate the control executions over time, and an efficient implementation on resource-constrained embedded hardware. The benefits provided by reactive control include higher accuracy in motion control and longer flight times.

We are currently working toward obtaining official certifications from the Italian civil aviation authority to fly drones running reactive control over public ground. Surprisingly, the major hampering factor is turning out not to be reactive control per se. The evidence we collected during our experiments, plus (i) additional fallback mechanisms we implement to switch back to time-triggered control in case of problems and (ii) extensive tests conducted by independent technicians and professional pilots, were sufficient to convince the authority on the efficient and dependable operation of reactive control.

Rather, the authority would like to obtain a precise specification of what kind of drone, intended in its physical parts, can support reactive control. In computing terms, this essentially means a specification of the target platform. This task represents a multi-disciplinary challenge, in that it requires skills and expertise beyond the computing domain and reaching into electronics, aeronautics, and mechanics. We believe much of the future of computer science rests here, at the confluence with other disciplines. □



References

1. Åström, K.J. Event based control. In *Analysis and Design of Nonlinear Control Systems*. Springer Verlag, 2007.
2. Åström, K.J., Häggglund, T. *Advanced PID control*. ISA—The Instrumentation, Systems, and Automation Society, 2006.
3. Bainomugisha, E., et al. A survey on reactive programming. *ACM Comput. Surv.*, 45, 4 (2013).
4. BBC News. Disaster drones: How robot teams can help in a crisis. [goo.gl/6efliV](http://goo.gl/6efliV).
5. Bekey, G.A. *Autonomous Robots: From Biological Inspiration to Implementation and Control*. The MIT Press, 2005.
6. Bouabdallah, S., Noth, A., Siegwart, R. PID vs LQ control techniques applied to an indoor micro quadrotor. In *Proceedings of IROS* (2004).
7. Bregu, E., Casamassima, N., Cantoni, D., Mottola, L., Whitehouse, K. Reactive control of autonomous drones. In *Proceedings of ACM MOBISYS* (2016).
8. Burgard, W., et al. Collaborative multi-robot exploration. In *Proceedings of ICRA* (2000).
9. Anderson, C. How I accidentally kickstarted the domestic drone boom. [goo.gl/SPOIR](http://goo.gl/SPOIR).
10. Edwards, J. Coherent reaction. In *Proceedings of the ACM Conference on Object Oriented Programming Systems Languages and Applications (OOPSLA)* (2009).
11. Faragher, R.M., et al. Captain Buzz: An all-smartphone autonomous delta-wing drone. In *Workshop on Micro Aerial Vehicle Networks, Systems, and Applications (colocated with ACM MOBISYS)* (2015).
12. Hosmer, D.W. Jr., Lemeshow, S., Sturdivant, R.X. *Applied Logistic Regression*, vol. 398. John Wiley & Sons, 2013.
13. Mens, T., Tourwé, T. A survey of software refactoring. *IEEE Transactions on Software Engineering* 30, 2 (2004).
14. Michael, N., et al. Cooperative manipulation and transportation with aerial robots. *Autonomous Robots* 30, 1 (2011).
15. Miluzzo, E., et al. Sensing meets mobile social networks: The design, implementation and evaluation of the CenceMe application. In *Proceedings of ACM SENSYS* (2008).
16. Mottola, L., Moretta, M., Ghezzi, C., Whitehouse, K. Team-level programming of drone sensor networks. In *Proceedings of ACM SENSYS* (2014).
17. Natalizio, E., Surace, R., Loscri, V., Guerriero, F., Melodia, T. Filming sport events with mobile camera drones: Mathematical modeling and algorithms. [goo.gl/v7Qo80](http://goo.gl/v7Qo80), 2012. Technical report.
18. Nex, F., Remondino, F. UAV for 3D mapping applications: A review. *Applied Geomatics* (2003).
19. Patelli, A., Mottola, L. Model-based real-time testing of drone autopilots. In *Proceedings of DRONET (colocated with ACM MOBISYS)* (2016).
20. Pieps. ARVA Transceivers. [goo.gl/tPywra](http://goo.gl/tPywra).
21. Ritz, R., et al. Cooperative quadcopter ball throwing and catching. In *Proceedings of IROS* (2012).
22. Sadeghzadeh, I., Zhang, Y. Actuator fault-tolerant control based on gain-scheduled PID with application to fixed-wing unmanned aerial vehicles. In *IEEE International Conference on Control and Fault-Tolerant Systems* (2013).
23. Turpin, M., Michael, N., Kumar, V. Decentralized formation control with variable shapes for aerial robots. In *Proceedings of ICRA* (2012).
24. Yim, M., et al. Modular self-reconfigurable robot systems. *IEEE Robotics Automation Magazine* 14, 1 (2007).
25. Zhuang, M., Atherton, D. Automatic tuning of optimum PID controllers. *IEEE Proceedings on Control Theory and Applications* 140, 3 (1993).

**Luca Mottola** ([luca.mottola@polimi.it](mailto:luca.mottola@polimi.it)), Politecnico di Milano, Italy and SICS Swedish ICT.

**Kamin Whitehouse** ([whitehouse@virginia.edu](mailto:whitehouse@virginia.edu)), University of Virginia, USA.

© 2018 ACM 0001-0782/18/10 \$15.00



## 12 rising-stars from different subfields of multimedia research discuss the challenges and state-of-the-art developments of their prospective research areas in a general manner to the broad community.

The field of multimedia is unique in offering a rich and dynamic forum for researchers from “traditional” fields to collaborate and develop new solutions and knowledge that transcend the boundaries of individual disciplines. Despite the prolific research activities and outcomes, however, few efforts have been made to develop books that serve as an introduction to the rich spectrum of topics covered by this broad field. A few books are available that either focus on specific subfields or basic background in multimedia. Tutorial-style materials covering the active topics being pursued by the leading researchers at frontiers of the field are currently lacking...UNTIL NOW.



ISBN: 978-1-970001-044 DOI: 10.1145/3122865  
<http://books.acm.org>  
<http://www.morganclaypoolpublishers.com/chang>

# Technical Perspective

## The Future of MPI

By Marc Snir

THE MPI COMMUNITY recently celebrated 25 years since the start of the MPI standardization effort. This early-1990s effort was due to the emergence of commodity clusters as a replacement to vector machines, in what was dubbed by Eugene Brooks as “The attack of the killer micros.” Commodity clusters needed very different software than vector systems, and two efforts were started to satisfy this need: The first effort, developed by High Performance Fortran Forum, was HPF—a data parallel extension to Fortran 90 that would provide portability across vector, SIMD, and cluster systems. The more modest second effort, developed by the Message Passing Interface Forum, was MPI—a portable message-passing library aimed specifically at clusters.

The MPI effort succeeded beyond the dreams of the early forum members. Today, all large supercomputers are commodity clusters, all support MPI, and basically all large scientific application codes; as well as an increasing number of data analytics codes, use MPI. The same will be true for the coming generation of exascale systems.

Early competitors to MPI, including HPF, have disappeared. This success has multiple reasons: Some good choices made in the MPI design, the relative ease of its implementation, the early availability of high-quality implementations, the confidence that an MPI library will continue to be available on future HPC systems, and the malleability of a library solution that can support multiple programming styles.

One critical cause of this success has been the continued evolution of the MPI specification, in support of evolving architectures and application needs: The MPI 1.1 specification, released in June 1995, was a document of 231 pages describing 128 functions; the MPI 3.1 specification, released June 2015, is an 836-page document describing 451 functions. Over time, MPI came to accommodate threads, parallel I/O, and an extensive set of collective operations, including non-blocking ones.


### The following paper convincingly shows that the potential of MPI one-sided communication can be realized.

One major extension to MPI has been the introduction of one-sided communication, first in MPI 2.0, and then, with major additions, in MPI 3.0. The main communication paradigm for MPI point-to-point communication has been two-sided communication, where a send call at the source is matched by a receive call at the destination. This paradigm has weaknesses: The complex matching rules of sends to receives result in significant software overheads, especially for receive operations; overlap of communication and computation requires the presence of an asynchronous communication agent that can poll queues concurrently with ongoing computation; and send-receive communication either requires an extra copying of messages (eager protocol) or extra handshakes between sender and receiver (rendezvous protocol).

One-sided communication requires the involvement of only one process: the source process (for Put) or the destination process (for Get). This already enables a significant reduction of software overheads. It requires the involved process to provide the location of both the local and remote communication buffers; this is rarely a problem since the same association between local and remote buffer tends to be reused multiple times. It separates between communication and synchronization as only one of the two communicating processes will know the communication occurred; this is often an advantage as one synchronization can cover multiple communications. Most importantly, one-sided communication,

especially Put, is a very good match to the capabilities of modern Network Interface Controllers (NICs): They very often support remote direct memory access (rDMA) operations whereby local and remote NICs collaborate in copying data from local memory to remote memory with no software involvement, aside from the call that initiates the transfer at the source node. Therefore, one-sided communication has the potential to significantly reduce the software overheads for communication.

This is extremely important as the next generation of networks and NICs will have the capability of handling tens or hundreds of millions of messages per second: With current communication protocols, this would mean that tens of GigaOps would be consumed by communication.

The following paper convincingly shows that the potential of MPI one-sided communication can be realized. It provides both a general framework for the efficient implementation of MPI one-sided communication on modern architectures, and an experimental proof that such an implementation can significantly reduce communication overheads and improve the performance of large-scale applications. The paper is timely and important for two reasons: First, users tend to avoid new features in MPI (or other software) unless they have a convincing proof of their advantages and a solid implementation; the paper provides such a proof and provides guidance for new releases of the MPI library. Second, hardware vendors are often focused on optimizing their future systems for past applications; NIC designers are focused on accelerating two-sided communication as it is currently the main communication paradigm. The paper provides a timely warning that more attention must be devoted to one-sided communication. 

Marc Snir is the Michael Faiman Professor in the Department of Computer Science at the University of Illinois at Urbana-Champaign, IL, USA.

Copyright held by author.

# Enabling Highly Scalable Remote Memory Access Programming with MPI-3 One Sided

By Robert Gerstenberger,\* Maciej Besta, and Torsten Hoefler

## Abstract

Modern high-performance networks offer remote direct memory access (RDMA) that exposes a process' virtual address space to other processes in the network. The *Message Passing Interface* (MPI) specification has recently been extended with a programming interface called MPI-3 Remote Memory Access (MPI-3 RMA) for efficiently exploiting state-of-the-art RDMA features. MPI-3 RMA enables a powerful programming model that alleviates many message passing downsides. In this work, we design and develop bufferless protocols that demonstrate how to implement this interface and support scaling to millions of cores with negligible memory consumption while providing highest performance and minimal overheads. To arm programmers, we provide a spectrum of performance models for RMA functions that enable rigorous mathematical analysis of application performance and facilitate the development of codes that solve given tasks within specified time and energy budgets. We validate the usability of our library and models with several application studies with up to half a million processes. In a wider sense, our work illustrates how to use RMA principles to accelerate computation- and data-intensive codes.

## 1. INTRODUCTION

Supercomputers have driven the progress of various society's domains by solving challenging and computationally intensive problems in fields such as climate modeling, weather prediction, engineering, or computational physics. More recently, the emergence of the "Big Data" problems resulted in the increasing focus on designing high-performance architectures that are able to process enormous amounts of data in domains such as personalized medicine, computational biology, graph analytics, and data mining in general. For example, the recently established Graph500 list ranks supercomputers based on their ability to traverse enormous graphs; the results from November 2014 illustrate that the most efficient machines can process up to 23 trillion edges per second in graphs with more than 2 trillion vertices.

\* RG performed much of the implementation during an internship at UIUC/NCSA while the analysis and documentation was performed during a scientific visit at ETH Zurich. RG's primary email address is gerstenberger.robert@gmail.com.

Supercomputers consist of massively parallel nodes, each supporting up to hundreds of hardware threads in a single shared-memory domain. Up to tens of thousands of such nodes can be connected with a high-performance network, providing large-scale distributed-memory parallelism. For example, the Blue Waters machine has >700,000 cores and a peak computational bandwidth of >13 petaflops.

Programming such large distributed computers is far from trivial: an ideal programming model should tame the complexity of the underlying hardware and offer an easy abstraction for the programmer to facilitate the development of high-performance codes. Yet, it should also be able to effectively utilize the available massive parallelism and various heterogeneous processing units to ensure highest scalability and speedups. Moreover, there has been a growing need for the support for *performance modeling*: a rigorous mathematical analysis of application performance. Such formal reasoning facilitates developing codes that solve given tasks within the assumed time and energy budget.

The *Message Passing Interface* (MPI)<sup>11</sup> is the *de facto* standard API used to develop applications for distributed-memory supercomputers. MPI specifies message passing as well as remote memory access semantics and offers a rich set of features that facilitate developing highly scalable and portable codes; message passing has been the prevalent model so far. MPI's message passing specification does not prescribe specific ways how to exchange messages and thus enables flexibility in the choice of algorithms and protocols. Specifically, to exchange messages, senders and receivers may use eager or rendezvous protocols. In the former, the sender sends a message without coordinating with the receiver; unexpected messages are typically buffered. In the latter, the sender waits until the receiver specifies the target buffer; this may require additional control messages for synchronization.

Despite its popularity, message passing often introduces time and energy overheads caused by the rendezvous control messages or copying of eager buffers; eager messaging may also require additional space at the receiver. Finally, the fundamental feature of message passing is that it *ouples communication and synchronization*: a message both transfers the data and synchronizes the receiver with the sender. This may prevent effective overlap of computation and

The original version of this paper was published in the *Proceedings of the Supercomputing Conference 2013 (SC'13)*, Nov. 2013, ACM.



communication and thus degrade performance.

The dominance of message passing has recently been questioned as novel hardware mechanisms are introduced, enabling new high-performance programming models. Specifically, network interfaces evolve rapidly to implement a growing set of features directly in hardware. A key feature of today's high-performance networks is remote direct memory access (RDMA), enabling a process to directly access virtual memory at remote processes without involvement of the operating system or activities at the remote side. RDMA is supported by on-chip networks in, for example, Intel's SCC and IBM's Cell systems, as well as off-chip networks such as InfiniBand, IBM's PERCS or BlueGene/Q, Cray's Gemini and Aries, or even RDMA over Ethernet/TCP (RoCE/iWARP).

The RDMA support gave rise to *Remote Memory Access* (RMA), a powerful programming model that provides the programmer with a Partitioned Global Address Space (PGAS) abstraction that unifies separate address spaces of processors while preserving the information on which parts are local and which are remote. A fundamental principle behind RMA is that it *relaxes synchronization and communication* and allows them to be managed independently. Here, processes use independent calls to initiate data transfer and to ensure the consistency of data in remote memories and the notification of processes. Thus, RMA generalizes the principles from shared memory programming to distributed memory computers where data coherency is explicitly managed by the programmer to ensure highest speedups.

Hardware-supported RMA has benefits over message passing in the following three dimensions: (1) *time* by avoiding synchronization overheads and additional messages in rendezvous protocols, (2) *energy* by eliminating excessive copying of eager messages, and (3) *space* by removing the need for receiver-side buffering. Several programming environments embrace RMA principles: PGAS languages such as Unified Parallel C (UPC) or Fortran 2008 Coarrays and libraries such as Cray SHMEM or MPI-2 One Sided. Significant experience with these models has been gained in the past years<sup>1,12,17</sup> and several key design principles for RMA programming evolved. Based on this experience, MPI's standardization body, the MPI Forum, has revamped the RMA (or One Sided) interface in the latest MPI-3 specification.<sup>11</sup> MPI-3 RMA supports the

newest generation of RDMA hardware and codifies existing RMA practice. A recent textbook<sup>4</sup> illustrates how to use this interface to develop high-performance large-scale codes.

However, it has yet to be shown how to implement the new library interface to deliver highest performance at lowest memory overheads. In this work, we design and develop scalable protocols for implementing MPI-3 RMA over RDMA networks, requiring  $\mathcal{O}(\log p)$  time and space per process on  $p$  processes. We demonstrate that the MPI-3 RMA interface can be implemented adding negligible overheads to the performance of the utilized hardware primitives.

In a wider sense, our work answers the question if the MPI-3 RMA interface is a viable candidate for moving towards exascale computing. Moreover, it illustrates that RMA principles provide significant speedups over message passing in both microbenchmarks and full production codes running on more than half a million processes. Finally, our work helps programmers to rigorously reason about application performance by providing a set of asymptotic as well as detailed performance models of RMA functions.

## 2. SCALABLE PROTOCOLS FOR RMA

We now describe protocols to implement MPI-3 RMA based on low-level RDMA functions. In all our protocols, we assume that we only have small bounded buffer space at each process ( $\mathcal{O}(\log p)$  for synchronization,  $\mathcal{O}(1)$  for communication), no remote software agent, and only put, get, and some basic atomic operations (atomics) for remote accesses. Thus, our protocols are applicable to all current RDMA networks and are forward-looking towards exascale network architectures.

We divide the RMA functionality of MPI into three separate concepts: (1) window creation, (2) communication functions, and (3) synchronization functions.

Figure 1a shows an overview of MPI's synchronization functions. They can be split into active target mode, in which the target process participates in the synchronization, and passive target mode, in which the target process is passive. Figure 1b shows a similar overview of MPI's communication functions. Several functions can be completed in bulk with bulk synchronization operations or using fine-grained request objects and test/wait functions. However, we observed that the completion model only minimally affects local overheads and is thus not considered separately in the rest of this work.

**Figure 1. An overview of MPI-3 RMA and associated cost functions. The figure shows abstract cost functions for all operations in terms of their input domains. (a) Synchronization and (b) Communication. The symbol  $p$  denotes the number of processes,  $s$  is the data size,  $k$  is the maximum number of neighbors, and  $o$  defines an MPI operation. The notation  $\mathcal{P}: \{p\} \rightarrow \mathcal{T}$  defines the input space for the performance (cost) function  $\mathcal{P}$ . In this case, it indicates, for a specific MPI function, that the execution time depends only on  $p$ . We provide asymptotic cost functions in Section 2 and parametrized cost functions for our implementation in Section 3.**

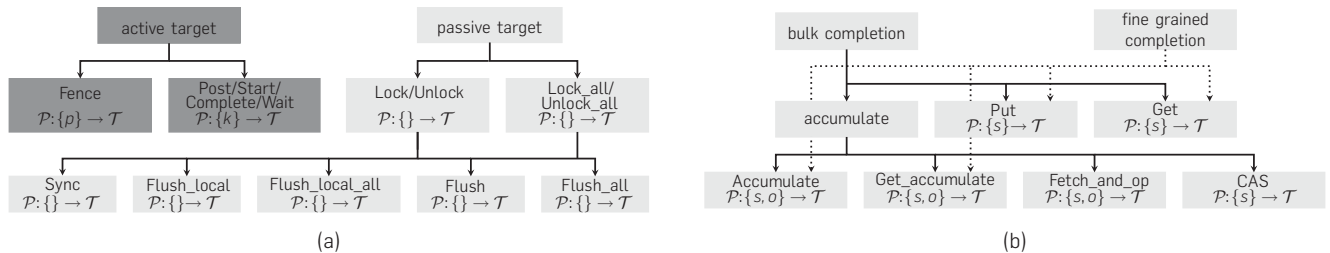


Figure 1 also shows abstract definitions of the performance models for each synchronization and communication operation. The performance model for each function depends on the exact implementation. We provide a detailed overview of the *asymptotic* as well as *exact* performance properties of our protocols and our implementation in the next sections. The different performance characteristics of communication and synchronization functions make a unique combination of implementation options for each specific use-case optimal. Yet, it is not always easy to choose this best variant. The exact models can be used to design close-to-optimal implementations (or as input for model-guided autotuning) while the simpler asymptotic models can be used in the algorithm design phase as exemplified by Karp et al.<sup>7</sup>

To support post-petascale computers, all protocols need to implement each function in a scalable way, that is, consuming  $\mathcal{O}(\log p)$  memory and time on  $p$  processes. For the purpose of explanation and illustration, we choose to discuss a reference implementation as a use-case. However, all protocols and schemes discussed in the following can be used on any RDMA-capable network.

### 2.1. Use-case: Cray DMAPP and XPMEM

Our reference implementation used to describe RMA protocols and principles is called FOMPI (fast one sided MPI). FOMPI is a fully functional MPI-3 RMA library implementation for Cray Gemini (XK5, XE6) and Aries (XC30)<sup>3</sup> systems. In order to maximize asynchronous progression and minimize overhead, FOMPI interfaces to the lowest-level available hardware APIs.

For inter-node (network) communication, FOMPI uses the RDMA API of Gemini and Aries networks: Distributed Memory Application (DMAPP). DMAPP offers put, get, and a limited set of atomic memory operations for certain 8 Byte datatypes. For intra-node communication, we use XPMEM,<sup>16</sup> a portable Linux kernel module that allows to map the memory of one process into the virtual address space of another. All operations can be directly implemented with load and store instructions, as well as CPU atomics (e.g., using the x86 lock prefix).

FOMPI's performance properties are self-consistent (i.e., respective FOMPI functions perform no worse than a combination of other FOMPI functions that implement the same functionality) and thus avoid surprises for users. We now proceed to develop algorithms to implement the window creation routines that expose local memory for remote access. After this, we describe protocols for communication and synchronization functions over RDMA networks.

### 2.2. Scalable window creation

An MPI *window* is a region of process memory that is made accessible to remote processes. We assume that communication memory needs to be registered with the communication subsystem and that remote processes require a remote descriptor that is returned from the registration to access the memory. This is true for most of today's RDMA interfaces including DMAPP and XPMEM.

FOMPI can be downloaded from [http://spcl.inf.ethz.ch/Research/Parallel\\_Programming/foMPI](http://spcl.inf.ethz.ch/Research/Parallel_Programming/foMPI).

**Traditional Windows.** These windows expose existing user-memory by specifying an arbitrary local base address. All remote accesses are relative to this address. Traditional windows are not scalable as they require  $\Omega(p)$  storage on each of the  $p$  processes in the worst case. Yet, they are useful when the library can only access user-specified memory. Memory addresses are exchanged with two MPI\_Allgather operations: one for DMAPP and one for XPMEM.

**Allocated Windows.** These windows allow the MPI library to allocate window memory and thus use identical base addresses on all nodes requiring only  $\mathcal{O}(1)$  storage. This can be done with a system-wide *symmetric heap* or with the following POSIX-compliant protocol: (1) a leader process chooses a random address and broadcasts it to other processes in the window, and (2) each process tries to allocate the memory with this specific address using `mmap()`. Those two steps are repeated until the allocation was successful on all the processes (this can be checked with `MPI_Allreduce`). This mechanism requires  $\mathcal{O}(\log p)$  time (with high probability).

**Dynamic Windows.** Here, windows can be dynamically resized by attaching or detaching memory regions with local `MPI_Win_attach` and `MPI_Win_detach` calls. They can be used in, for example, dynamic RMA-based data structures. In our implementation, the former call registers a memory region and inserts the information into a linked list; the latter removes a region from the list. Both calls require  $\mathcal{O}(1)$  memory per region. The access to the list on a target is purely one sided. We use a local cache to reduce the number of remote accesses; a simple protocol uses gets to ensure the cache validity and to update local information if necessary.

**Shared Memory Windows.** These windows are only valid for intra-node communication, enabling efficient load and store accesses. They can be implemented with POSIX shared memory or XPMEM with constant memory overhead per core.<sup>5</sup> We implement the intra-node case as a variant of allocated windows, providing identical performance and full compatibility with shared memory windows.

### 2.3. Communication functions

Communication functions map nearly directly to low-level hardware functions, enabling significant speedups over message passing. This is a major strength of RMA programming. In FOMPI, put and get simply use DMAPP put and get for remote accesses or local `memcpy` for XPMEM accesses. Accumulates either use DMAPP atomics (for common integer operations on 8 Byte data) or fall back to a simple protocol that locks the remote window, gets the data, accumulates it locally, and writes it back. This fallback protocol ensures that the target is not involved in the communication for true passive mode. It can be improved if we allow buffering (enabling a space-time trade-off<sup>18</sup>) and active messages to perform the remote operations atomically.

We now show novel protocols to implement synchronization modes in a scalable way on pure RDMA networks without remote buffering.

### 2.4. Scalable window synchronization

MPI defines *exposure* and *access* epochs. A process starts an exposure epoch to allow other processes access to its

memory. To access exposed memory at a remote target, the origin process has to be in an access epoch. Processes can be in access and exposure epochs simultaneously. Exposure epochs are only defined for active target synchronization (in passive target, window memory is always exposed).

**Fence.** `MPI_Win_fence`, called collectively by all processes, finishes the previous exposure and access epoch and opens the next exposure and access epoch for the whole window. All remote memory operations must be committed before leaving the fence call. We use an x86 `m fence` instruction (XPMEM) and DMAPP bulk synchronization (`gsync`) followed by an MPI barrier to ensure global completion. The asymptotic memory bound is  $\mathcal{O}(1)$  and, assuming a good barrier implementation, the time bound is  $\mathcal{O}(\log p)$ .

**General Active Target Synchronization.** This mode (also called “PSCW”) synchronizes a subset of processes of a window and thus enables synchronization at a finer granularity than that possible with fences. Exposure (`MPI_Win_post/MPI_Win_wait`) and access epochs (`MPI_Win_start/MPI_Win_complete`) can be opened and closed independently. A group argument is associated with each call that starts an epoch; it states all processes participating in the epoch. The calls have to ensure correct *matching*: if a process  $i$  specifies a process  $j$  in the group argument of the post call, then the next start call at process  $j$  with  $i$  in the group argument *matches* the post call.

Since our RMA implementation cannot assume buffer space for remote operations, it has to ensure that all processes in the group argument of the start call have issued a matching post before the start returns. Similarly, the wait call has to ensure that all matching processes have issued complete. Thus, calls to `MPI_Win_start` and `MPI_Win_wait` may block, waiting for the remote process. Both synchronizations are required to ensure integrity of the accessed data during the epochs. The MPI specification forbids matching configurations where processes wait cyclically (deadlocks).

We now describe a scalable matching protocol with a time and memory complexity of  $\mathcal{O}(k)$  if each process has at most  $k$  neighbors across all epochs. We assume  $k$  is known to the protocol. We start with a high-level description: process  $i$  that *posts* an epoch announces itself to all processes  $j_1, \dots, j_l$  in the group argument by adding  $i$  to a list local to the processes  $j_1, \dots, j_l$ . Each process  $j$  that tries to *start* an access epoch waits until all processes  $i_1, \dots, i_m$  in the group argument are present in its local list. The main complexity lies in the scalable storage of this neighbor list, needed for *start*, which requires a remote free-storage management scheme. The *wait* call can simply be synchronized with a completion counter. A process calling *wait* will not return until the completion counter reaches the number of processes in the specified group. To enable this, the *complete* call first guarantees remote visibility of all issued RMA operations (by calling `mfence` or DMAPP’s `gsync`) and then increases the completion counter at all processes of the specified group.

If  $k$  is the size of the group, then the number of operations issued by post and complete is  $\mathcal{O}(k)$  and zero for start and wait. We assume that  $k \in \mathcal{O}(\log p)$  in scalable programs.

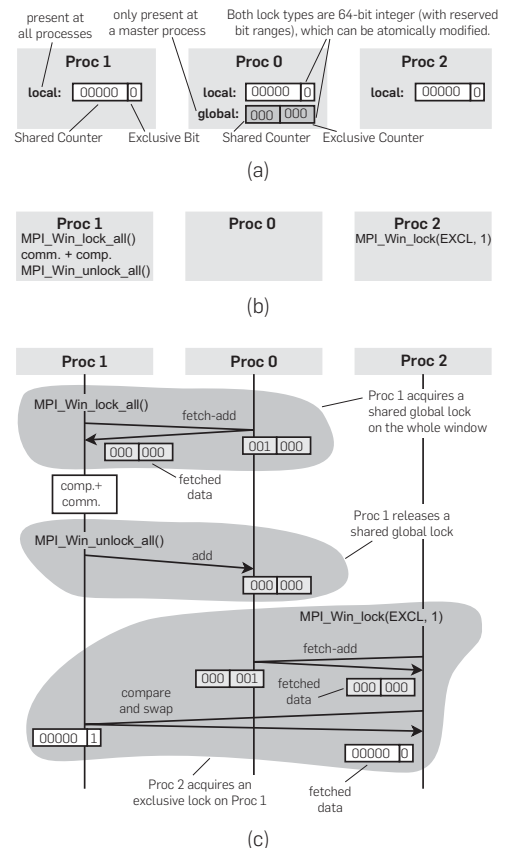
A more detailed explanation can be found in our SC13 paper.

**Lock Synchronization.** We now sketch a low-overhead and scalable strategy to implement shared global, shared process-local, and exclusive process-local locks on RMA systems (the MPI specification does not allow exclusive global locks). These mechanisms allow to synchronize processes and memories at very fine granularities. We utilize a two-level lock hierarchy: one global lock variable (at a designated process, called *master*) and  $p$  local lock variables (one lock on each process).

Each local lock variable is used to implement a reader-writer lock that allows one writer (*exclusive* lock), but many readers (*shared* locks). The highest order bit of the variable indicates a write access; the other bits are used to count the number of shared locks (cf. Ref.<sup>8</sup>). The global lock variable is split into two parts; they count the number of processes holding a shared global lock in the window and the number of exclusively locked processes, respectively. These variables enable all lock operations to complete in  $\mathcal{O}(1)$  steps if a lock can be acquired immediately; they are pictured in Figure 2a.

Figure 2b shows an exemplary lock scenario for three processes. We omit a detailed description of the protocol due to the lack of space (the source code is available online); we describe a locking scenario to illustrate the core idea behind the protocol. Figure 2c shows a possible execution schedule for the scenario from Figure 2b. Please note that we permuted the order of processes to (1, 0, 2) instead of the intuitive (0, 1, 2) to minimize overlapping lines in the figure.

**Figure 2. Example of lock synchronization. (a) Data structures, (b) Source code, and (c) A possible schedule.**





An acquisition of a shared global lock (MPI\_Win\_lock\_all) only involves the global lock on the master. The origin (Process 1) fetches and increases the lock in one atomic operation. Since there is no exclusive lock present, Process 1 can proceed. Otherwise, it would repeatedly (remotely) read the lock until no writer was present; exponential back off can be used to avoid congestion.

For a local exclusive lock, the origin needs to ensure two invariants: (1) no shared global lock **and** (2) no local shared or exclusive lock can be held or acquired during the local exclusive lock. For the first part, the origin (Process 2) atomically fetches the global lock from the master and increases the writer part to register for an exclusive lock. If the fetched value indicates lock all accesses, the origin backs off. As there is no global reader, Process 2 proceeds to the second invariant and tries to acquire an exclusive local lock on Process 1 using a compare-and-swap (CAS) with zero (cf. Ref.<sup>8</sup>). It succeeds and acquires the lock. If one of the two steps fails, the origin backs off and repeats the operation.

When unlocking (MPI\_Win\_unlock\_all) a shared global lock, the origin only atomically decreases the global lock on the master. The unlocking of an exclusive lock requires two steps: clearing the exclusive bit of the local lock, and then atomically decreasing the writer part of the global lock.

The acquisition or release of a shared local lock (MPI\_Win\_lock/MPI\_Win\_unlock) is similar to the shared global case, except it targets a local lock.

If no exclusive locks exist, then shared locks (both local and global) only take one remote atomic operation. The number of remote requests while waiting can be bound by using MCS locks.<sup>9</sup> An exclusive lock will take in the best case two atomic communication operations. Unlock operations always cost one atomic operation, except for the exclusive case with one extra atomic operation for releasing the global lock. The memory overhead for all functions is  $\mathcal{O}(1)$ .

**Flush.** Flush guarantees remote completion and is thus one of the most performance-critical functions on MPI-3 RMA programming. FOMPI's flush implementation relies on the underlying interfaces and simply issues a DMAPP remote bulk completion and an x86 mfence. All

flush operations (MPI\_Win\_flush, MPI\_Win\_flush\_local, MPI\_Win\_flush\_all, and MPI\_Win\_flush\_all\_local) share the same implementation and add only 78 CPU instructions (on x86) to the critical path.

### 3. DETAILED PERFORMANCE MODELING AND EVALUATION

We now analyze the performance of our protocols and implementation and compare it to Cray MPI's highly tuned point-to-point as well as its relatively untuned one sided communication. In addition, we compare FOMPI with two major HPC PGAS languages: UPC and Fortran 2008 Coarrays, both specially tuned for Cray systems. We execute all benchmarks on the Blue Waters supercomputer, using Cray XE6 nodes. Each node contains four 8-core AMD Opteron 6276 (Interlagos) 2.3GHz CPUs and is connected to other nodes through a 3D-Torus Gemini network. Additional results can be found in the original SC13 paper.

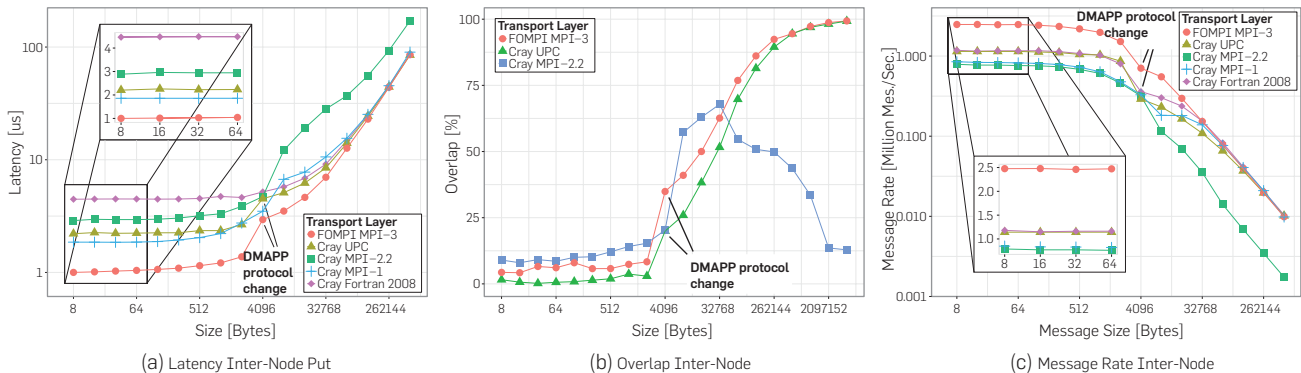
#### 3.1. Communication

Comparing latency and bandwidth between RMA and point-to-point communication is not always fair since RMA communication may require extra synchronization to notify the target. For all RMA latency results we ensure remote completion (the data is committed in remote memory) but no synchronization. We analyze synchronization costs separately in Section 3.2.

**Latency and Bandwidth.** We start with the analysis of latency and bandwidth. The former is important in various latency-constrained codes such as interactive graph processing frameworks and search engines. The latter represents a broad class of communication-intensive applications such as graph analytics engines or distributed key-value stores.

We measure point-to-point latency with standard ping-pong techniques. Figure 3a shows the latency for varying message sizes for inter-node put. Due to the highly optimized fast-path, FOMPI has >50% lower latency than other PGAS models while achieving the same bandwidth for larger messages. The performance functions (cf. Figure 1) are:  $\mathcal{P}_{put} = 0.16ns \cdot s + 1\mu s$  and  $\mathcal{P}_{get} = 0.17ns \cdot s + 1.9\mu s$ .

**Figure 3. Microbenchmarks:** (a) Latency comparison for put with DMAPP communication. Note that message passing (MPI-1) implies remote synchronization while UPC, Fortran 2008 Coarrays, and MPI-2.2/3 only guarantee consistency. (b) Communication/computation overlap for put over DMAPP, Cray MPI-2.2 has much higher latency up to 64 KB (cf. a), thus allows higher overlap. (c) Message rate for put communication.



**Overlapping Computation.** Overlapping computation with communication is a technique in which computation is progressed while waiting for communication to be finished. Thus, it reduces the number of idle CPU cycles. Here, we measure how much of such overlap can be achieved with the compared libraries and languages. The benchmark calibrates a computation loop to consume slightly more time than the latency. Then it places computation between communication and synchronization and measures the combined time. The ratio of overlapped computation is then computed from the measured communication, computation, and combined times. Figure 3b shows the ratio of the overlapped communication for Cray’s MPI-2.2, UPC, and FOMPI.

**Message Rate.** This benchmark is similar to the latency benchmark. However, it benchmarks the start of 1000 transactions without synchronization to determine the overhead for starting a single operation. Figure 3c presents the results for the inter-node case. Here, injecting a single 8 Byte operation costs only 416ns.

**Atomics.** As the next step we analyze the performance of various atomics that are used in a broad class of lock-free and wait-free codes. Figure 4a shows the performance of the DMAPP-accelerated MPI\_SUM of 8 Byte elements, a non-accelerated MPI\_MIN, and 8 Byte CAS. The performance functions are  $\mathcal{P}_{acc,sum} = 28ns \cdot s + 2.4\mu s$ ,  $\mathcal{P}_{acc,min} = 0.8ns \cdot s + 7.3\mu s$ , and  $\mathcal{P}_{CAS} = 2.4\mu s$ . The DMAPP acceleration lowers the latency for small operations while the locked implementation exhibits a higher bandwidth. However, this does not consider the serialization due to the locking.

### 3.2. Synchronization schemes

Finally, we evaluate synchronization schemes utilized in numerous parallel protocols and systems. The different synchronization modes have nontrivial trade-offs. For example PSCW performs better for small groups of processes and fence performs best for groups that are essentially as big as the full group attached to the window. However, the exact crossover point is a function of the implementation and system. While the active target mode notifies the target implicitly that its memory is consistent, in passive target mode, the user has to do this

explicitly or rely on synchronization side effects of other functions (e.g., allreduce).

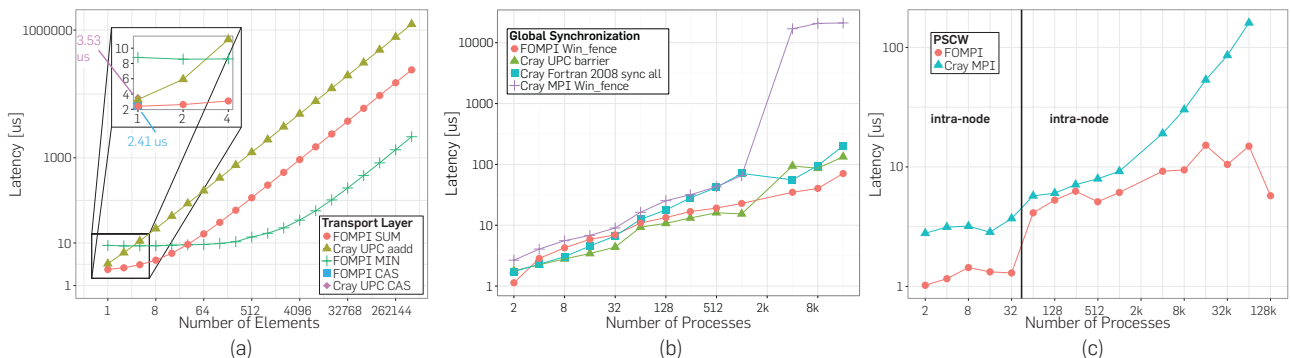
**Global Synchronization.** Global synchronization is performed in applications based on the Bulk Synchronous Parallel (BSP) model. It is offered by fences in MPI. It can be directly compared to Fortran 2008 Coarrays sync all and UPC’s upc\_barrier which also synchronize the memory at all processes. Figure 4b compares the performance of FOMPI with Cray’s MPI-2.2, UPC, and Fortran 2008 Coarrays implementations. The performance function for FOMPI’s fence implementation is:  $\mathcal{P}_{fence} = 2.9\mu s \cdot \log_2(p)$ .

**General Active Target Synchronization (PSCW).** This mode may accelerate codes where the communication graph is static or changes infrequently, for example stencil computations. Only MPI offers PSCW. Figure 4c shows the performance for Cray MPI-2.2 and FOMPI when synchronizing a ring where each process has exactly two neighbors ( $k = 2$ ). An ideal implementation would exhibit constant time. We observe systematically growing overheads in Cray’s MPI as well as system noise (due to network congestion, OS interrupts and daemons, and others) on runs with >1000 processes with FOMPI. We model the performance with varying numbers of neighbors and FOMPI’s PSCW synchronization costs involving  $k$  off-node neighbor are  $\mathcal{P}_{post} = \mathcal{P}_{complete} = 350ns \cdot k$ , and  $\mathcal{P}_{start} = 0.7\mu s$ ,  $\mathcal{P}_{wait} = 1.8\mu s$  (without noise).

**Passive Target Synchronization.** Finally, we evaluate lock-based synchronization that can be utilized to develop high-performance distributed-memory variants of shared-memory lock-based codes. The performance of lock/unlock is constant in the number of processes as ensured by our protocols and thus not graphed. The performance functions are  $\mathcal{P}_{lock,excl} = 5.4\mu s$ ,  $\mathcal{P}_{lock,shrd} = \mathcal{P}_{lock,all} = 2.7\mu s$ ,  $\mathcal{P}_{unlock,shrd} = \mathcal{P}_{unlock,all} = 0.4\mu s$ ,  $\mathcal{P}_{unlock,excl} = 4.0\mu s$ ,  $\mathcal{P}_{flush} = 76ns$ , and  $\mathcal{P}_{sync} = 17ns$ .

We demonstrated the performance of our protocols and implementation using microbenchmarks comparing to other RMA and message passing codes. The exact performance models can be utilized to design and optimize parallel applications, however, this is outside the scope of the paper. To demonstrate the usability and performance of our design for real codes, we continue with a large-scale application study.

**Figure 4. Performance of atomic accumulate operations and synchronization latencies. (a) Atomic Operation Performance, (b) Latency for Global Synchronization, and (c) Latency for PSCW (Ring Topology).**



#### 4. ACCELERATING FULL CODES WITH RMA

To compare our protocols and implementation with the state of the art, we analyze a 3D FFT code as well as the MIMD Lattice Computation (MILC) full production application with several hundred thousand lines of source code that performs quantum field theory computations. Other application case-studies can be found in the original SC13 paper, they include a distributed hashtable representing many big data and analytics applications and a dynamic sparse data exchange representing graph traversals and complex modern scientific codes such as n-body methods.

In all the codes, we keep most parameters constant to compare the performance of PGAS languages, message passing, and MPI RMA. Thus, we did not employ advanced concepts, such as MPI datatypes or process topologies, which are not available in all designs (e.g., UPC and Fortran 2008).

##### 4.1. 3D fast Fourier transform

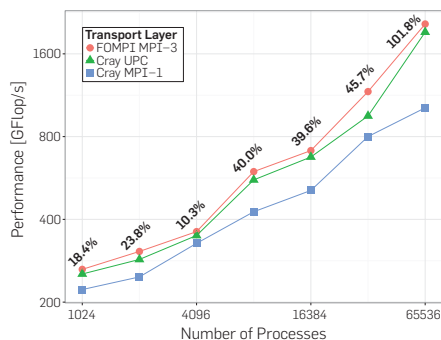
We now discuss how to exploit overlap of computation and communication in a 3D Fast Fourier Transformation. We use Cray's MPI and UPC versions of the NAS 3D FFT benchmark. Nishtala et al.<sup>12</sup> and Bell et al.<sup>1</sup> demonstrated that overlap of computation and communication can be used to improve the performance of a 2D-decomposed 3D FFT. We compare the default "nonblocking MPI" with the "UPC slab" decomposition, which starts to communicate the data of a plane as soon as it is available and completes the communication as late as possible. For a fair comparison, our FOMPI implementation uses the same decomposition and communication scheme like the UPC version and required minimal code changes resulting in the same code complexity.

Figure 5 illustrates the results for the strong scaling class D benchmark ( $2048 \times 1024 \times 1024$ ). UPC achieves a consistent speedup over message passing, mostly due to the communication and computation overlap. FOMPI has a somewhat lower static overhead than UPC and thus enables better overlap (cf. Figure 3b) and slightly higher performance.

##### 4.2. MIMD lattice computation

The MIMD Lattice Computation (MILC) Collaboration studies Quantum Chromodynamics (QCD), the theory of strong interaction.<sup>2</sup> The group develops a set of applications, known as the MILC code, which regularly gets one of the largest allocations at US NSF supercomputer centers. The

**Figure 5. 3D FFT Performance.** The annotations represent the improvement of foMPI over message passing.



su3\_rmd module, which is part of the SPEC CPU2006 and SPEC MPI benchmarks, is included in the MILC code.

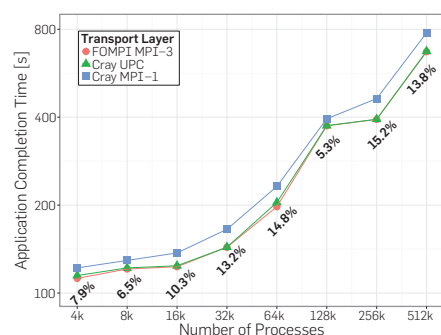
The program performs a stencil computation on a 4D rectangular grid and it decomposes the domain in all four dimensions to minimize the surface-to-volume ratio. To keep data consistent, neighbor communication is performed in all eight directions. Global allreductions are done regularly to check the solver convergence. The most time-consuming part of MILC is the conjugate gradient solver which uses nonblocking communication overlapped with local computations.

Figure 6 shows the execution time of the whole application for a weak-scaling problem with a local lattice of  $4^3 \times 8$ , a size very similar to the original Blue Waters Petascale benchmark. Some computation phases (e.g., CG) execute up to 45% faster, yet, we chose to report full-code performance. Cray's UPC and FOMPI exhibit essentially the same performance, while the UPC code uses Cray-specific tuning<sup>15</sup> and the MPI-3 code is portable to different architectures. The full-application performance gain over Cray's MPI-1 version is more than 15% for some configurations. The application was scaled successfully to up to 524,288 processes with all implementations. This result and our microbenchmarks demonstrate the scalability and performance of our protocols and that the MPI-3 RMA library interface can achieve speedups competitive to compiled languages such as UPC and Fortran 2008 Coarrays while offering all of MPI's convenient functionalities (e.g., Topologies and Datatypes). Finally, we illustrate that the new MPI-3 RMA semantics enable full applications to achieve significant speedups over message passing in a fully portable way. Since most of those existing codes are written in MPI, a step-wise transformation can be used to optimize most critical parts first.

#### 5. RELATED WORK

PGAS programming has been investigated in the context of UPC and Fortran 2008 Coarrays. For example, an optimized UPC Barnes Hut implementation shows similarities to MPI-3 RMA programming by using bulk vectorized memory transfers combined with vector reductions instead of shared pointer accesses.<sup>17</sup> Highly optimized PGAS applications often use a style that can easily be adapted to MPI-3 RMA.

**Figure 6. Full MILC code execution time.** The annotations represent the improvement of foMPI over message passing.





The intricacies of MPI-2.2 RMA implementations over InfiniBand networks have been discussed by Jian et al.<sup>6</sup> and Santhanaraman et al.<sup>14</sup> Zhao et al.<sup>18</sup> describe an adaptive strategy to switch from eager to lazy modes in active target synchronizations in MPICH 2. This mode could be used to speed up these of FoMPI's atomics that are not supported in hardware.

The applicability of MPI-2.2 RMA has also been demonstrated for some applications. Mirin and Sawyer<sup>10</sup> discuss the usage of MPI-2.2 RMA coupled with threading to improve the Community Atmosphere Model (CAM). Potluri et al.<sup>13</sup> show that MPI-2.2 RMA with overlap can improve the communication in a Seismic Modeling application. However, we demonstrated new MPI-3 features, such as lock-all epochs, flushes, and allocated windows, which can be used to further improve performance by utilizing state-of-the-art RDMA features and simplified implementations.


## 6. DISCUSSION AND CONCLUSION

In this work, we demonstrate how the MPI-3 RMA library interface can be implemented over RDMA networks to achieve highest performance and lowest memory overheads. We provide detailed performance models that help choosing among the multiple options. For example, a user can decide whether to use Fence or PSCW synchronization (if  $\mathcal{P}_{fence} > \mathcal{P}_{post} + \mathcal{P}_{complete} + \mathcal{P}_{start} + \mathcal{P}_{wait}$ , which is true for large  $k$ ). This is just one example for the possible uses of the provided detailed performance models.

We study all overheads in detail and provide performance evaluations for all critical RMA functions. Our implementation proved to be scalable and robust while running on 524,288 processes on Blue Waters speeding up a full application run by 13.8% and a 3D FFT on 65,536 processes by a factor of two. These gains will directly translate to significant energy savings in big data and HPC computations.

We expect that the principles and scalable synchronization algorithms developed in this work will act as a blueprint for optimized RMA implementations over future large-scale RDMA networks. We also conjecture that the demonstration of highest performance to users will quickly increase the number of RMA programs. Finally, as the presented techniques can be applied to data-centric codes, we expect that RMA programming will also accelerate emerging data center computations.

## Acknowledgments

We thank Timo Schneider for early help in the project, Greg Bauer and Bill Kramer for support with Blue Waters, Cray's Duncan Roweth, and Roberto Ansaloni for help with Cray's PGAS environment, Nick Wright for the UPC version of MILC, and Paul Hargrove for the UPC version of NASFT. This work was supported in part by the DOE Office of Science, Advanced Scientific Computing Research, under award number DE-FC02-10ER26011, program manager Lucy Nowell. This work is partially supported by the Blue Waters sustained-petascale computing project, which is supported by the National Science Foundation (award number OCI 07-25070) and the state of Illinois. MB is supported by the 2013 Google European Doctoral Fellowship in Parallel Computing. 

## References

- Bell, C., Bonachea, D., Nishtala, R., Yelick, K. Optimizing bandwidth limited problems using one-sided communication and overlap. In *Proceedings of the International Conference on Parallel and Distributed Processing (IPDPS'06)* (2006). IEEE Computer Society, 1–10.
- Bernard, C., Ogilvie, M.C., DeGrand, T.A., DeTar, C.E., Gottlieb, S.A., Krasnitz, A., Sugar, R., Toussaint, D. Studying quarks and gluons on MIMD parallel computers. *J. High Perform. Comput. Appl.* 5, 4 (1991), 61–70.
- Faanee, G., Bataineh, A., Roweth, D., Court, T., Froese, E., Alverson, B., Johnson, T., Kopnick, J., Higgins, M., Reinhard, J. Cray Cascade: A Scalable HPC System Based on a Dragonfly Network. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC'12)* (2012). IEEE Computer Society, Los Alamitos, CA, 103:1–103:9.
- Gropp, W., Hoefler, T., Thakur, R., Lusk, E. *Using Advanced MPI: Modern Features of the Message-Passing Interface*. MIT Press, Cambridge, MA, Nov. (2014).
- Hoefler, T., Dinan, J., Buntinas, D., Balaji, P., Barrett, B., Brightwell, R., Gropp, W., Kale, V., Thakur, R. Leveraging MPI's one-sided communication interface for shared-memory programming. In *Recent Advances in the Message Passing Interface (EuroMPI'12)*, Volume LNCS 7490 (2012). Springer, 132–141.
- Jiang, W., Liu, J., Jin, H.-W., Panda, D.K., Gropp, W., Thakur, R. High performance MPI-2 one-sided communication over InfiniBand. In *Proceedings of the IEEE International Symposium on Cluster Computing and the Grid (CCGRID'04)* (2004). IEEE Computer Society, 531–538.
- Karp, R.M., Sahay, A., Santos, E.E., Schauer, K.E. Optimal broadcast and summation in the LogP model. In *Proceedings of the ACM Symposium on Parallel Algorithms and Architectures (SPAA'93)* (1993). ACM, New York, NY, USA, 142–153.
- Mellor-Crummey, J.M., Scott, M.L. Scalable reader-writer synchronization for shared-memory multiprocessors. *SIGPLAN Notices* 26, 7 (1991), 106–113.
- Mellor-Crummey, J.M., Scott, M.L. Synchronization without contention. *SIGPLAN Notices* 26, 4 (1991), 269–278.
- Mirin, A.A., Sawyer, W.B. A scalable implementation of a finite-volume dynamical core in the community atmosphere model. *J. High Perform. Comput. Appl.* 19, 3 (2005), 203–212.
- MPI Forum. *MPI: A Message-Passing Interface Standard. Version 3.0* (2012).
- Nishtala, R., Hargrove, P.H., Bonachea, D.O., Yelick, K.A. Scaling communication-intensive applications on BlueGene/P using one-sided communication and overlap. In *Proceedings of the IEEE International Parallel and Distributed Processing Symposium (IPDPS'09)* (2009). IEEE Computer Society, 1–12.
- Potturi, S., Lai, P., Tomko, K., Sur, S., Cui, Y., Tatineni, M., Schulz, K.W., Barth, W.L., Majumdar, A., Panda, D.K. Quantifying performance benefits of overlap using MPI-2 in a seismic modeling application. In *Proceedings of the ACM International Conference on Supercomputing (ICS'10)* (2010). ACM 17–25.
- Santhanaraman, G., Balaji, P., Gopalakrishnan, K., Thakur, R., Gropp, W., Panda, D.K. Natively supporting true one-sided communication in MPI on multi-core systems with InfiniBand. In *Proceedings of the IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID'09)* (2009), 380–387.
- Shan, H., Austin, B., Wright, N., Strohmaier, E., Shalf, J., Yelick, K. Accelerating applications at scale using one-sided communication. In *Proceedings of the Conference on Partitioned Global Address Space Programming Models (PGAS'12)* (2012).
- Woodacre, M., Robb, D., Roe, D., Feind, K. *The SGI Altix TM 3000 Global Shared-Memory Architecture* (2003). SGI HPC White Papers.
- Zhang, J., Behzad, B., Snir, M. Optimizing the Barnes-Hut algorithm in UPC. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC'11)* (2011). ACM, 75:1–75:11.
- Zhao, X., Santhanaraman, G., Gropp, W. Adaptive strategy for one-sided communication in MPICH2. In *Recent Advances in the Message Passing Interface (EuroMPI'12)* (2012). Springer, 16–26.

Robert Gerstenberger, Maciej Besta, and Torsten Hoefler ([robertge, maciej.best, htor}@inf.ethz.ch) ETH Zurich, Switzerland.

# CAREERS

## **Bentley University**

### **Tenure Track Assistant/Associate Professor, User Experience**

Bentley University invites applications for a tenure-track position in the areas of user experience, HCI, product design, or related fields to start in fall 2019. We seek a creative scholar who would join the interdisciplinary faculty of Bentley's internationally renowned user experience graduate program. We are also open to candidates in industry in these areas.

The reputation of our graduate program is grounded in educating leaders in user experience research and design. Central to our program's success is the strong connections it has forged with leading technology groups throughout the United States and internationally. We support that network with programs on our main campus, San Francisco and online. Further reinforcing this reputation is our well-known User Experience Center, a research and consulting group contracting with tech organizations around the world. Finally, most of our classes have research and design problems sponsored by startups, non-profit organizations, and large tech companies. Being part of a business university, our graduate program has a strong focus on the strategic role of the user experience in the competitive positioning of products in the marketplace and fostering continuous product innovation.

Bentley University is an AACSB and EQUIS accredited business university located 11 miles outside of Boston. Bentley leads higher education in the integration of global business with the arts and sciences. We seek faculty and staff who represent diverse backgrounds, interests, and talents. We strive to create a campus community that welcomes the exchange of ideas, and fosters a culture that values differences and views them as a strength in our community.

Candidates are encouraged to learn more about our UX community at Bentley by visiting <https://admissions.bentley.edu/graduate/masters-in-human-factors>.

#### **Required Qualifications**

Candidates must have evidence of research ability and experience with teaching. A Ph.D. is required.

#### **Special Instructions to Applicants**

Interested candidates should submit a CV, cover letter and research statement <https://bentley.wd1.myworkdayjobs.com/faculty>. Names and contact information for three references will be required upon application. Bentley will contact these confidential references for those candidates moving forward in the process.

#### **Application Instructions**

To apply to this position, please submit an online application directly to: [https://bentley.wd1.myworkdayjobs.com/faculty/job/Bentley-Campus/Tenure-Track-Assistant-Associate-Professor--User-Experience\\_R0000006](https://bentley.wd1.myworkdayjobs.com/faculty/job/Bentley-Campus/Tenure-Track-Assistant-Associate-Professor--User-Experience_R0000006)

## **Bilkent University**

### **Assistant Professor, Associate Professor and Full Professor**

The Computer Engineering Department at Bilkent University invites applications for multiple faculty positions. Appointments may be made at Assistant Professor, as well as at Associate Professor or Full Professor rank for candidates with commensurate experiences and accomplishments.

The department emphasizes both high quality research and teaching. Faculty duties include teaching at the graduate and undergraduate levels, research, supervision of these and other related tasks. The department will consider candidates with backgrounds and interests in all areas in computer science and engineering. We are particularly interested in the areas of data science, intelligent systems and data analytics.

Bilkent University is an equal opportunity employer. We place great emphasis on interdisciplinary research, university-industry collaboration, and innovative teaching. The university offers internationally competitive salaries and benefit packages including furnished housing, health insurance and pension funding.

Each application should describe the professional interests and goals of the candidate in both teaching and research, include a resume and the names and contact information of three or more individuals who will provide letters of recommendation. Review of applications will begin immediately and will continue until the positions are filled.

**Please apply through <http://stars.bilkent.edu.tr/staffapp/CS2018>**

#### **Requirements**

Ph.D. in computer science or a related discipline and evidence of exceptional research promise.

## **Massachusetts Institute of Technology Faculty Positions**

The Massachusetts Institute of Technology (MIT) Department of Electrical Engineering and Computer Science (EECS) seeks candidates for faculty positions starting in September 1, 2019, or on a mutually agreed date thereafter. Appointment will be at the assistant or untenured associate professor level. In special cases, a senior faculty appointment may be possible. Faculty duties include teaching at the undergraduate and graduate levels, advising students, conducting original scholarly research, and developing course materials at the graduate and undergraduate levels. Candidates should hold a Ph.D. in electrical engineering and computer science or a related field by the start of employment. We will consider candidates with research and teaching interests in any area of electrical engineering and computer science.

Candidates must register with the EECS search website at <https://school-of-engineering-faculty-search.mit.edu/eecs/>, and must submit application materials electronically to this website. Candi-

date applications should include a description of professional interests and goals in both teaching and research. Each application should include a curriculum vitae and the names and addresses of three or more individuals who will provide letters of recommendation. Letter writers should submit their letters directly to MIT, preferably on the website or by mailing to the address below. Complete applications should be received by December 1, 2018. Applications will be considered complete only when both the applicant materials and **at least three letters of recommendation are received.**

**It is the responsibility of the candidate to arrange reference letters to be uploaded at <https://school-of-engineering-faculty-search.mit.edu/eecs/> by December 1, 2018.**

Send all materials not submitted on the website to:

Professor Asu Ozdaglar  
Department Head, Electrical Engineering  
and Computer Science  
Massachusetts Institute of Technology  
Room 38-403  
77 Massachusetts Avenue  
Cambridge, MA 02139

M.I.T. is an equal opportunity/affirmative action employer. Women and minorities are encouraged to apply.

## **National University of Singapore**

### **Department of Computer Science**

#### **Tenure-track Assistant Professor Positions in Computer Science**

The Department of Computer Science at the National University of Singapore (NUS) invites applications for tenure-track Assistant Professor positions in all areas of computer science. Candidates should be early in their academic careers and yet demonstrate outstanding research potential, and a strong commitment to teaching.

The Department enjoys ample research funding, moderate teaching loads, excellent facilities, and extensive international collaborations. We have a full range of faculty covering all major research areas in computer science and boasts a thriving PhD program that attracts the brightest students from the region and beyond. More information is available at [www.comp.nus.edu.sg/careers](http://www.comp.nus.edu.sg/careers).

NUS is an equal opportunity employer that offers highly competitive salaries, and is situated in Singapore, an English-speaking cosmopolitan city that is a melting pot of many cultures, both the east and the west. Singapore offers high-quality education and healthcare at all levels, as well as very low tax rates.

#### **Application Details:**

- ▶ Submit the following documents (in a single PDF) online via: <https://faces.comp.nus.edu.sg>
  - A cover letter that indicates the position applied for and the main research interests
  - Curriculum Vitae

- A teaching statement
- A research statement

► Provide the contact information of 3 referees when submitting your online application, or, arrange for at least 3 references to be sent directly to [csrec@comp.nus.edu.sg](mailto:csrec@comp.nus.edu.sg)

► Application reviews will commence immediately and continue until the positions are filled

If you have further enquiries, please contact the Search Committee Chair, Weng-Fai Wong, at [csrec@comp.nus.edu.sg](mailto:csrec@comp.nus.edu.sg).

## Ohio University

### Assistant/Associate/Full Professor of Computer Science

The School of Electrical Engineering and Computer Science at Ohio University invites candidates to apply for a tenure-track position in computer science. The selected applicant will be expected to perform excellent research, teaching, and service in computer science. Departmental support will include initial reduced teaching loads, competitive salary and generous start-up funds. Candidates must have an earned doctorate computer science or a related discipline by start date of the appointment. Candidates from all relevant research areas are welcomed, but special consideration will be given to candidates with a record of high-quality research and scholarship in computer security, formal methods, artificial intelligence, machine learning, data analytics or software engineering.

The School of Electrical Engineering and

Computer Science is in the Russ College of Engineering and Technology at Ohio University. The School of EECS offers bachelor's, master's, and doctoral degrees. At present there are roughly 500 undergraduate majors, 170 master's degree students and 40 PhD students in the School of EECS. We employ 22 full-time tenured and tenure-track faculty. New sponsored research awards in the School of EECS have averaged roughly \$5M per year over the last five years, and numerous research collaboration opportunities exist within the School's Center for Scientific Computing and Immersive Technologies (CSCIT) and within the Avionics Engineering Center.

Ohio University is a public, comprehensive university that conducts high quality research across many disciplines, and emphasizes an excellent, learner-centered educational experience by providing undergraduate, graduate, and professional programs to approximately 20,000 students in a residential setting. The Ohio University area features a national forest, state parks and recreation opportunities such as hiking, bicycling, camping, and canoeing.

To apply, complete the online application and attach required documents. Required documents include: CV, cover letter, a statement of research interests and priorities, a statement of teaching philosophy and priorities, and a list of three references with current contact information.

<http://www.ohiouniversityjobs.com/postings/28181>

Review of applications will begin immediately and continue until the position is filled. For full consideration, please apply by December 2, 2018.

## San Diego State University

### Department of Computer Science

#### Two Tenure-Track Assistant Professor Positions

The Department of Computer Science at SDSU seeks to hire two tenure-track Assistant Professors starting Fall 2019. The candidates should have PhD degrees in Computer Science or closely related fields. One position is in Cybersecurity (see <https://apply.interfolio.com/53552>); the other position is in Algorithms & Computation (see <https://apply.interfolio.com/53547>). Questions about the position may be directed to [COS-CS-Search@sdsu.edu](mailto:COS-CS-Search@sdsu.edu). Top candidates in other areas will also be considered. **SDSU is an equal opportunity/Title IX employer.**

## Southern University of Science and Technology (SUSTech)

### Tenure-Track Faculty Positions

The Department of Computer Science and Engineering (CSE, <http://cse.sustc.edu.cn/en/>), Southern University of Science and Technology (SUSTech) has multiple Tenure-track faculty openings at all ranks, including Professor/Associate Professor/Assistant Professor. We are looking for outstanding candidates with demonstrated research achievements and keen interest in teaching, in the following areas (but are not restricted to):

- Data Science
- Artificial Intelligence



SAARLAND UNIVERSITY

SAARBRÜCKEN GRADUATE SCHOOL OF COMPUTER SCIENCE

SIC Saarland Informatics Campus

**study – research – excellence**

**JOIN OUR WORLD**

Saarland Informatics Campus is one of the top locations for Computer Science. If you have a passion for research, this an ideal place to earn your PhD in Computer Science. Full funding included.

**Apply at: [www.graduateschool-computerscience.de](http://www.graduateschool-computerscience.de)**

**APPLY BY APRIL 30<sup>TH</sup> OR OCTOBER 31<sup>ST</sup> EACH YEAR**



- ▶ Computer Systems (including Networks, Cloud Computing, IoT, Software Engineering, etc.)
- ▶ Cognitive Robotics and Autonomous Systems
- ▶ Cybersecurity (including Cryptography)

Applicants should have an earned Ph.D. degree and demonstrated achievements in both research and teaching. The teaching language at SUSTech is bilingual, either English or Putonghua. It is perfectly acceptable to use English in all lectures, assignments, exams. In fact, our existing faculty members include several non-Chinese speaking professors.

As a State-level innovative city, Shenzhen has identified innovation as the key strategy for its development. It is home to some of China's most successful high-tech companies, such as Huawei and Tencent. SUSTech considers entrepreneurship as one of the main directions of the university. Strong supports will be provided to possible new initiatives. SUSTech encourages candidates with experience in entrepreneurship to apply.

The Department of Computer Science and Engineering at SUSTech was founded in 2016. It has 17 professors, all of whom hold doctoral degrees or have years of experience in overseas universities. Among them, two were elected into the "1000 Talents" Program in China; three are IEEE fellows; one IET fellow. The department is expected to grow to 50 tenure track faculty members eventually, in addition to teaching-only professors and research-only professors.

SUSTech is committed to increase the diversity of its faculty, and has a range of family-friendly policies in place. The university offers competitive salaries and fringe benefits including medi-

cal insurance, retirement and housing subsidy, which are among the best in China. Salary and rank will commensurate with qualifications and experience. More information can be found at <http://talent.sustc.edu.cn/en>.

We provide some of the best start-up packages in the sector to our faculty members, including one PhD studentship per year, in addition to a significant amount of start-up funding (which can be used to fund additional PhD students and post-docs, research travels, and research equipments).

To apply, please provide a cover letter identifying the primary area of research, curriculum vitae, and research and teaching statements, and forward them to [cshire@sustc.edu.cn](mailto:cshire@sustc.edu.cn).

### Stanford University Faculty positions in Operations, Information and Technology

The Operations, Information and Technology (OIT) area at the Graduate School of Business, Stanford University, is seeking qualified applicants for full-time, tenure-track positions, starting September 1, 2019. All ranks and relevant disciplines will be considered. Applicants are considered in all areas of Operations, Information and Technology (OIT) that are broadly defined to include the analytical and empirical study of technological systems, in which technology, people, and markets interact. It thus includes operations, information systems/technology, and management of technology. Applicants are expected to have rigorous training in management science,

engineering, computer science, economics, and/or statistical modeling methodologies. Candidates with strong empirical training in economics, behavioral science or computer science are encouraged to apply. The appointed will be expected to do innovative research in the OIT field, to participate in the school's PhD program, and to teach both required and elective courses in the MBA program. Junior applicants should have or expect to complete a PhD by September 1, 2019.

While the Graduate School of Business will not be conducting any interviews at the INFORMS meeting in Phoenix, AZ, some members of the OIT faculty will be attending. Candidates who will be presenting at INFORMS are strongly encouraged to submit their CV, a research abstract and any supporting information by October 28, 2018. We will continue to accept applications until November 15, 2018.

Applicants should submit their applications electronically by visiting the web site <http://www.gsb.stanford.edu/recruiting> and uploading their curriculum vitae, research papers and publications, and teaching evaluations, if applicable, on that site. **For an application to be considered complete, all applicants must submit a CV, a job market paper and arrange for three letters of recommendation to be submitted by November 15, 2018.** For questions regarding the application process, please send an email to [Faculty\\_Recruiter@gsb.stanford.edu](mailto:Faculty_Recruiter@gsb.stanford.edu).

Stanford is an equal employment opportunity and affirmative action employer. All qualified applicants will receive consideration for employment without regard to race, color, religion,



Faculty of Computer Science  
and Biomedical Engineering



The Department of Computer Science and Biomedical Engineering at Graz University of Technology invites applications for the following faculty positions.

- Full Professor (tenured) in **Bioinformatics**
- Full Professor (tenured) in **Information Security**
- Full Professor (5-years contract) in **Intelligent and Adaptive User Interfaces**
- Tenure Track Professor in **Cryptography** for women only.
- Tenure Track Professor in **Health Care Engineering** for women only.
- Tenure Track Professor in **Natural Language Processing**

The Department of Computer Science and Biomedical Engineering at Graz University of Technology is committed to excellence in research and teaching. The research at the department covers a broad spectrum of topics that are reflected in our main research areas "Biomedical Engineering," "Safety and Security," "Intelligent Systems," and "Visual Computing." We are proud of our outstanding applied and basic research and stimulate interdisciplinary projects. Moreover, our faculty fosters a close dialogue with local industry and encourages the establishment of spin-offs.

Graz University of Technology aims to increase the proportion of women and therefore qualified female applicants are explicitly encouraged to apply. Graz University of Technology actively promotes diversity and equal opportunities. People with disabilities who have the relevant qualifications are expressly invited to apply.

Application deadline: **3 December 2018**. For details, see <https://www.tugraz.at/fakultaeten/infbio/news/vacancies> or contact us at [applications.csbme@tugraz.at](mailto:applications.csbme@tugraz.at)



## ADVERTISING IN CAREER OPPORTUNITIES

**How to Submit a Classified Line Ad: Send an e-mail to [acmm mediasales@acm.org](mailto:acmm mediasales@acm.org). Please include text, and indicate the issue/or issues where the ad will appear, and a contact name and number.**

**Estimates: An insertion order will then be e-mailed back to you. The ad will be typeset according to CACM guidelines. NO PROOFS can be sent. Classified line ads are NOT commissionable.**

**Deadlines: 20th of the month/2 months prior to issue date. For latest deadline info, please contact: [acmm mediasales@acm.org](mailto:acmm mediasales@acm.org)**

**Career Opportunities Online: Classified and recruitment display ads receive a free duplicate listing on our website at: <http://jobs.acm.org>**

**Ads are listed for a period of 30 days.  
For More Information Contact:**

**ACM Media Sales  
at 212-626-0686 or  
[acmm mediasales@acm.org](mailto:acmm mediasales@acm.org)**

sex, sexual orientation, gender identity, national origin, disability, protected veteran status, or any other characteristic protected by law. Stanford also welcomes applications from others who would bring additional dimensions to the University's research, teaching and clinical missions.

---

### University of Southern California, Information Sciences Institute Multiple Computer Scientist/Research Positions

The Information Sciences Institute (ISI) at the University of Southern California (USC) is a world leader in the research and development of advanced information processing, computing and communications technologies. ISI played a pivotal role in the information revolution, developing and managing the early internet and its predecessor, ARPANet. Today, its research spans artificial intelligence (AI), cybersecurity, grid computing, quantum computing, microelectronics, supercomputing, nanosatellites and many other areas.

ISI has three research campuses: one in Marina Del Rey, CA; one in Arlington, VA; and one in Waltham, MA. For detailed information about each position or to apply, please visit the web pages listed below.

#### Arlington, Virginia

- Computer Scientist – EDA Algorithm  
Researcher: <http://ow.ly/gwhd30l28oN>
- Computer Scientist – Reconfigurable  
Abstraction Researcher:  
<http://ow.ly/54nP30l28r4>
- Postdoctoral Scholar Research Associate –  
Reconfigurable Computing:  
<http://ow.ly/XZTs30l28ul>
- Research Programmer II – Vision/AI:  
<http://ow.ly/uwH630l28gZ4>
- Computer Scientist – HPC:  
<http://ow.ly/qrGb30l1woV>
- Computer Scientist – Virtualization:  
<http://ow.ly/9bkv30l1wPI6>
- Computer Scientist – Real Time:  
<http://ow.ly/h9AH30l1ma2K>

#### Waltham, Massachusetts

- Research Programmer II – Natural Language  
Processing: <http://ow.ly/uvBE30l1wPO9>
- Research Programmer I – Natural Language  
Processing: <http://ow.ly/KSq030l28P6>
- Computer Scientist – Natural Language  
Processing: <http://ow.ly/YS9e30l28Xl>

#### Marina del Rey, California

- Postdoctoral Scholar – Research Associate:  
<http://ow.ly/yXga30l2938>

---

### The University of Texas at San Antonio (UTSA) Faculty Position in Computer Science

The Department of Computer Science at The University of Texas at San Antonio (UTSA) invites applications for **one tenure-track or tenured open rank** (Assistant, Associate or Full Professor) position, starting in Fall 2019. This position is targeted towards faculty with expertise and interest in artificial intelligence (AI). Outstanding candi-

dates from all areas of AI will be considered, and preference will be given to applicants with expertise in cyber adversarial learning, AI for resource-constrained systems (such as IoTs and embedded systems), or AI (such as natural language processing, computer vision and deep learning) as it relates to health-related applications. This position is part of the university-wide cluster hiring in Artificial Intelligence.

See <http://www.cs.utsa.edu/fsearch> for information on the Department and application instructions. Screening of applications will begin immediately.

Application received by **January 2, 2019** will be given full consideration. The search will continue until the positions are filled or the search is closed. The University of Texas at San Antonio is an Affirmative Action/Equal Opportunity Employer. Women, minorities, veterans, and individuals with disabilities are encouraged to apply.

---

**Department of Computer Science**  
**RE: Faculty Search**  
**The University of Texas at San Antonio**  
**One UTSA Circle**  
**San Antonio, TX 78249-0667**  
**Phone: 210-458-4436**

---

### University of Toronto Assistant Professor, Teaching Stream

The Edward S. Rogers Sr. Department of Electrical and Computer Engineering (ECE) at the University of Toronto invites applications for a full-time teaching-stream faculty appointment at the rank of Assistant Professor, Teaching Stream, in the general area of Computer Systems and Software. The appointment will commence on July 1, 2019.

Applicants are expected to have a Ph.D. in Electrical and Computer Engineering, or a related field, at the time of appointment or soon after.

Successful candidates will have demonstrated excellence in teaching and pedagogical inquiry, including in the development and delivery of undergraduate courses and laboratories and supervision of undergraduate design projects. This will be demonstrated by strong communication skills, a compelling statement of teaching submitted as part of the application highlighting areas of interest, awards and accomplishments and teaching philosophy; sample course syllabi and materials; and teaching evaluations, as well as strong letters of reference from referees of high standing endorsing excellent teaching and commitment to excellent pedagogical practices and teaching innovation.

Eligibility and willingness to register as a Professional Engineer in Ontario is highly desirable.

Salary will be commensurate with qualifications and experience.

The Edward S. Rogers Sr. Department of Electrical and Computer Engineering at the University of Toronto ranks among the best in North America. It attracts outstanding students, has excellent facilities, and is ideally located in the middle of a vibrant, artistic, diverse and cosmopolitan city. Additional information may be found at <http://www.ece.utoronto.ca>.

Review of applications will begin after September 1, 2018, however, the position will remain open until November 29, 2018.

As part of your online application, please

include a cover letter, a curriculum vitae, and a teaching dossier including a summary of your previous teaching experience, your teaching philosophy and accomplishments, your future teaching plans and interests, sample course syllabi and materials, and teaching evaluations. Applicants must arrange for three letters of reference to be sent directly by the referees (on letterhead, signed and scanned), by email to the ECE department at [search2018@ece.utoronto.ca](mailto:search2018@ece.utoronto.ca). Applications without any reference letters will not be considered; it is your responsibility to make sure your referees send us the letters while the position remains open.

You must submit your application online while the position is open, by following the submission guidelines given at <http://uoft.me/how-to-apply>. Applications submitted in any other way will not be considered. We recommend combining attached documents into one or two files in PDF/MS Word format. If you have any questions about this position, please contact the ECE department at [search2018@ece.utoronto.ca](mailto:search2018@ece.utoronto.ca).

The University of Toronto is strongly committed to diversity within its community and especially welcomes applications from racialized persons / persons of colour, women, Indigenous / Aboriginal People of North America, persons with disabilities, LGBTQ persons, and others who may contribute to the further diversification of ideas.

As part of your application, you will be asked to complete a brief Diversity Survey. This survey is voluntary. Any information directly related to you is confidential and cannot be accessed by search committees or human resources staff. Results will be aggregated for institutional planning purposes. For more information, please see <http://uoft.me/UP>.

All qualified candidates are encouraged to apply; however, Canadians and permanent residents will be given priority.

---

### University of Toronto Assistant Professor, Tenure Stream

The Edward S. Rogers Sr. Department of Electrical and Computer Engineering (ECE) at the University of Toronto invites applications for up to four full-time tenure-stream faculty appointments at the rank of Assistant Professor. The appointments will commence on July 1, 2019.

Within the general field of electrical and computer engineering, we seek applications from candidates with expertise in one or more of the following strategic research areas: 1. Computer Systems and Software; 2. Electrical Power Systems; 3. Systems Control, including but not limited to autonomous and robotic systems.

Applicants are expected to have a Ph.D. in Electrical and Computer Engineering, or a related field, at the time of appointment or soon after.

Successful candidates will be expected to initiate and lead an outstanding, innovative, independent, competitive, and externally funded research program of international calibre, and to teach at both the undergraduate and graduate levels. Candidates should have demonstrated excellence in research and teaching. Excellence in research is evidenced primarily by publications or forthcoming publications in leading journals or conferences in the field, presentations at significant conferences, awards and accolades, and

strong endorsements by referees of high international standing. Evidence of excellence in teaching will be demonstrated by strong communication skills; a compelling statement of teaching submitted as part of the application highlighting areas of interest, awards and accomplishments, and teaching philosophy; sample course syllabi and materials; and teaching evaluations, as well as strong letters of recommendation.

Eligibility and willingness to register as a Professional Engineer in Ontario is highly desirable.

Salary will be commensurate with qualifications and experience.

The Edward S. Rogers Sr. Department of Electrical and Computer Engineering at the University of Toronto ranks among the best in North America. It attracts outstanding students, has excellent facilities, and is ideally located in the middle of a vibrant, artistic, diverse and cosmopolitan city.

Additional information may be found at <http://www.ece.utoronto.ca>.

Review of applications will begin after September 1, 2018, however, the position will remain open until November 29, 2018.

As part of your online application, please include a cover letter, a curriculum vitae, a summary of your previous research and future research plans, as well as a teaching dossier including a statement of teaching experience and interests, your teaching philosophy and accomplishments, and teaching evaluations. Applicants must arrange for three letters of reference to be sent directly by the referees (on letterhead, signed and scanned), by email to the ECE department at [search2018@ece.utoronto.ca](mailto:search2018@ece.utoronto.ca). Applications without any reference letters will not be considered; it is your responsibility to make sure your referees send us the letters while the position remains open.

You must submit your application online while the position is open, by following the submission guidelines given at <http://uoft.me/how-to-apply>. Applications submitted in any other way will not be considered. We recommend combining attached documents into one or two files in PDF/MS Word format. If you have any questions about this position, please contact the ECE department at [search2018@ece.utoronto.ca](mailto:search2018@ece.utoronto.ca).

The University of Toronto is strongly committed to diversity within its community and especially welcomes applications from racialized persons / persons of colour, women, Indigenous / Aboriginal People of North America, persons with disabilities, LGBTQ persons, and others who may contribute to the further diversification of ideas.

As part of your application, you will be asked to complete a brief Diversity Survey. This survey is voluntary. Any information directly related to you is confidential and cannot be accessed by search committees or human resources staff. Results will be aggregated for institutional planning purposes. For more information, please see <http://uoft.me/UP>.

All qualified candidates are encouraged to apply; however, Canadians and permanent residents will be given priority.

### **University of Toronto** **Associate Professor, Tenure Stream**

The Edward S. Rogers Sr. Department of Electrical and Computer Engineering (ECE) at the

University of Toronto invites applications for up to four full-time tenure-stream faculty appointments at the rank of Associate Professor. The appointments will commence on July 1, 2019.

Within the general field of electrical and computer engineering, we seek applications from candidates with expertise in one or more of the following strategic research areas: 1. Computer Systems and Software; 2. Electrical Power Systems; 3. Systems Control, including but not limited to autonomous and robotic systems.

Applicants are expected to have a Ph.D. in Electrical and Computer Engineering, or a related field, and have at least five years of academic or relevant industrial experience.

Successful candidates will be expected to maintain and lead an outstanding, independent, competitive, innovative, and externally funded research program of international calibre, and to teach at both the undergraduate and graduate levels. Candidates should have demonstrated excellence in research and teaching. Excellence in research is evidenced primarily by sustained and impactful publications in leading journals or conferences in the field, awards and accolades, presentations at significant conferences and a high profile in the field with strong endorsements by referees of high international standing. Evidence of excellence in teaching will be demonstrated by strong communication skills, a compelling statement of teaching submitted as part of the application highlighting areas of interest, awards and accomplishments, and teaching philosophy; sample course syllabi and materials; and teaching evaluations, as well as strong letters of recommendation.

Eligibility and willingness to register as a Professional Engineer in Ontario is highly desirable.

Salary will be commensurate with qualifications and experience.

The Edward S. Rogers Sr. Department of Electrical and Computer Engineering at the University of Toronto ranks among the best in North America. It attracts outstanding students, has excellent facilities, and is ideally located in the middle of a vibrant, artistic, diverse and cosmopolitan city. Additional information may be found at <http://www.ece.utoronto.ca>.

Review of applications will begin after September 1, 2018, however, the position will remain open until November 29, 2018.

As part of your online application, please include a cover letter, a curriculum vitae, a summary of your previous research and future research plans, as well as a teaching dossier including a statement of teaching experience and interests, your teaching philosophy and accomplishments, and teaching evaluations. Applicants must arrange for three letters of reference to be sent directly by the referees (on letterhead, signed and scanned), by email to the ECE department at [search2018@ece.utoronto.ca](mailto:search2018@ece.utoronto.ca). Applications without any reference letters will not be considered; it is your responsibility to make sure your referees send us the letters while the position remains open.

You must submit your application online while the position is open, by following the submission guidelines given at <http://uoft.me/how-to-apply>. Applications submitted in any other way will not be considered. We recommend combining attached documents into one or two files in PDF/MS Word format. If you have any questions about this position, please contact the ECE de-

partment at [search2018@ece.utoronto.ca](mailto:search2018@ece.utoronto.ca).

The University of Toronto is strongly committed to diversity within its community and especially welcomes applications from racialized persons / persons of colour, women, Indigenous / Aboriginal People of North America, persons with disabilities, LGBTQ persons, and others who may contribute to the further diversification of ideas.

As part of your application, you will be asked to complete a brief Diversity Survey. This survey is voluntary. Any information directly related to you is confidential and cannot be accessed by search committees or human resources staff. Results will be aggregated for institutional planning purposes. For more information, please see <http://uoft.me/UP>.

All qualified candidates are encouraged to apply; however, Canadians and permanent residents will be given priority.

### **University of Zurich** **Assistant Professorship in Interacting with Data (Non-tenure Track)**

The Faculty of Business, Economics and Informatics of the University of Zurich invites applications for an Assistant Professorship in Interacting with Data (Non-tenure Track) starting in 2019.

Candidates should hold a Ph.D. degree in Computer Science with specialization in Interactive Data Analysis, Visual Analytics, Information Visualization or related areas and have an excellent record of academic achievements in the relevant fields. A strong motivation to teach both at the undergraduate and the graduate levels as well as an interest in human and societal aspects of managing data are highly beneficial.

The successful candidate is expected to establish her or his research group within the Department of Informatics, actively interface with the other groups at the department and the faculty, and seek collaboration with researchers across faculties within the Digital Society Initiative of the University of Zurich.

Through its educational and research objectives, the University of Zurich aims at attracting leading international researchers who are willing to contribute to its development and to strengthening its reputation. The University of Zurich is an equal opportunity employer and strongly encourages applications from female candidates.

Please submit your application at <https://www.facultyhiring.oec.uzh.ch/position/9633792> before October 15, 2018.

Documents should be addressed to Prof. Dr. Harald Gall; Dean of the Faculty of Business, Economics and Informatics; University of Zurich; Switzerland.

For further questions regarding the profile of the open position please contact Prof. Renato Pajarola ([pajarola@ifi.uzh.ch](mailto:pajarola@ifi.uzh.ch))



[CONTINUED FROM P. 120] coding was defined in the context of something called multicast.

Multicast is a communications protocol in which you deliver the same information to a group of destinations simultaneously.

But in networking, typically, that's not how it works. In networking, you typically have unicast, where one sender transmits to a single destination. Even when you are sending something like broadcast television over the Internet, your broadcast is actually using unicast. You have your server turning that traffic to all the individuals who are interested in it.

What Muriel and I did was try to take that really beautiful, elegant theory, and think about it in the context of real networks. I felt wireless networks, in particular, might be the right environment for this technology. Wireless is way more limited in terms of data rate and bandwidth than wired networks, and it's also less reliable. So network coding is an ideal solution when you make an error in your transmission.

**In your recent work, you've used wireless signals to track people's motions—even through walls. How did you get that idea?**

When we began, it was really curiosity. Let's say there is a room and you don't have access to it. Can you tell if there are people in the room? If you can tell there are people, can you tell how many people? When we tried that, we didn't really know whether or not it was possible, and we certainly didn't know what kind of application you'd use it for. All we knew is that we have been able to track people using their cellphones—so, using a wireless signal, but a wireless signal that is emitted from a device. And we have some understanding of how wireless works in an indoor environment and propagates through walls and materials.

**After your initial demonstrations were successful, the questions got more complex, and practical applications began to present themselves.**

Once we started working with it, we began to have all these ideas—why stop at just being able to see if people are moving? Can you tell how

they are moving? Can you tell how many people there are? It turns out you can, because they are breathing... So what other physiological signals can you extract?

And these questions are extremely intellectually interesting, but it's not just that; they have very practical and useful applications to people's lives!

**You're now working, through a start-up called Emerald, to commercialize the technology and develop some of those applications—for instance, remotely monitoring people's health.**

We talk a lot about the smart home, but really the smartest thing a home can do is to take care of us and our health. Our vision is to have a technology that disappears into the environment; I don't have to enter information about my heartrate, or put some device on myself and remember to charge it. I don't need to change my behavior in any way, but still there is a home that's watching over my health and keeping track of problems early on—or even before they occur—and alerting doctors or the hospital or a caregiver.

**That sounds promising. Where are you in your efforts?**

At this early stage, our focus is to work with healthcare providers, on the one hand, and with the biotech and pharma industry, on the other. It turns out there are many deep physiological signals we can extract, so we need to connect with people who understand what those signals mean in the context of diseases. I can tell you that my mom is walking well or that she fell—that's the extent of it. I couldn't tell you if the patterns of information indicate we should change the dose on her Parkinson's medication.

**One of the most consistently cited features of your work is creativity.**

In general, in almost all the stuff I do, I'm driven by curiosity. I'm always interested in trying something where I don't know the answer, or where I'm not sure whether the answer is "yes" or "no."

*Leah Hoffmann* is a technology writer based in Piermont, NY, USA.

© 2018 ACM 0001-0782/18/10 \$15.00



## Distinguished Speakers Program

<http://dsp.acm.org>

Students and faculty can take advantage of ACM's Distinguished Speakers Program to invite renowned thought leaders in academia, industry and government to deliver compelling and insightful talks on the most important topics in computing and IT today. ACM covers the cost of transportation for the speaker to travel to your event.



Association for Computing Machinery

## Q&amp;A

# Reaping the Benefits of a Diverse Background

*Earlier this year, ACM named Dina Katabi of the Massachusetts Institute of Technology's Computer Science and Artificial Intelligence Laboratory recipient of the 2017 ACM Prize in Computing for her creative contributions to wireless systems.*

DINA KATABI, RECIPIENT of the 2017 ACM Prize in Computing, took a winding road to computing, and it paid off. Now a professor of electrical engineering and computer science at the Massachusetts Institute of Technology (MIT), Katabi began her career in medicine. Since making the transition, she has made numerous creative contributions to wireless network design. Today, she is helping to develop medical applications for a technology she pioneered, which uses wireless signals to sense humans and their movements through walls—her early training coming full circle.

**Your undergraduate degree is in electrical engineering, but you began by studying medicine.**

In Syria, after high school, there is a nationwide exam, and the expectation is that the top people will go to medical school. I took the exam, and I ranked very high. I also come from a family of doctors. So I went to med school, but after the first year, I decided I could not continue. I wanted to do math and engineering, so I decided to switch.

**You then came to the U.S., and did your Ph.D. in computer science.**

At the time, computer science (CS) was a very new field in Syria. In fact, at the school I attended, there was no such thing as a CS school or department. But I was always fascinated with computers, and when I came



to the U.S., I wanted to learn more about algorithms.

**Having experience with different fields seems to have proven beneficial to your work.**

I've benefited from having a very diverse background, which has enabled me to see beyond the field I am in. Particularly when I was working on wireless systems, my background gave me the expertise I needed to design the circuit, the signal, and also the algorithm that extracts information from that signal. You can design many systems with electrical engineering, but the ability to add intelligence to them using CS is much more powerful than if it was just pure signal processing.

**In some of your earliest work at MIT, you collaborated with David Clark—the Internet's chief protocol architect during most of the 1980s—on network control, where one of the biggest problems is managing transmissions when they threaten to overwhelm the network.**

At the time, the traditional method of congestion control was based on heuristics. It was more of an art than anything. But it was often not very efficient—not very fair to different users—because the Internet is just too big. With my thesis, I tried to connect that art and intuition to the field of control theory, which is a subfield in electrical engineering that is typically used to control plants and manufacturing systems. So you can keep the intuition, but if you infuse into it some of the mathematical models, you can achieve much better results. You can make the network more stable and achieve more efficient systems.

**After you received your Ph.D., you stayed at MIT and began working with information theorist Muriel Médard on network coding, a technique for increasing networks' data capacity that was promising in theory, but had not yet been shown to work on a real network.**

When we began our work, network coding had shown high gains in specific examples, but those examples did not map to the way that networks really operate. For instance, the theory of network [CONTINUED ON P. 119]

# acm Inroads

PAVING THE WAY TOWARD EXCELLENCE IN COMPUTING EDUCATION



Association for  
Computing Machinery

## ACM/SIGCSE Seek Co-EDITORS-IN-CHIEF for *ACM Inroads*



ACM and the Special Interest Group on Computer Science Education (SIGCSE) seek co-editors-in-chief (co-EICs) to lead its quarterly magazine *ACM Inroads*.

The magazine serves computing education professionals globally by fostering dialogue, cooperation, and collaboration between educators worldwide. It achieves this by publishing high-quality content describing, analyzing, and critiquing current issues and practices affecting computer education now and in the future.

The magazine is written by and for educators, with each issue presenting thought-provoking commentaries and articles that examine current research and practices within the computing community.

For more about *Inroads*, see <http://inroads.acm.org/>



### Job Description

The EIC position is a highly visible, hands-on volunteer position responsible for leading, networking, and overseeing all editorial aspects of the magazine's content creation process, including but not limited to: soliciting articles from prospective authors; managing the magazine's editorial board and contributors to meet quarterly publication deadlines; creating new editorial features, special sections, columns; upholding a high bar for the content's quality and diversity; assigning manuscripts to associate editors for review; making final editorial decisions; setting the overall direction and online strategy of the publication. Prior experience leading or managing editorial projects a plus.



### Eligibility Requirements

The co-EiCs search is open to applicants worldwide.

Applications are welcome from both individuals and from pairs wishing to serve as co-EiCs.

**Applicants must be willing and able to make a 3-year commitment to this post.**

To apply, please send your CV along with a 300-word vision statement expressing the reasons for your interest in the position to: [eicsearch@inroads.acm.org](mailto:eicsearch@inroads.acm.org)

**The deadline for submissions is OCTOBER 15, 2018.**

**The editorship will commence on DECEMBER 1, 2018.**







SIGGRAPH  
ASIA 2018  
TOKYO



# CROSSOVER

The 11th ACM SIGGRAPH Conference  
and Exhibition on Computer Graphics  
and Interactive Techniques in Asia

**CONFERENCE** 4 – 7 December 2018  
**EXHIBITION** 5 – 7 December 2018  
Tokyo International Forum, Japan

[SA2018.SIGGRAPH.ORG](http://SA2018.SIGGRAPH.ORG)

Sponsored by



Organized by



we energize your business | since 1924