COMMUNICATIONS OF THE ACM

CACM.ACM.ORG

03/2020 VOL.63 NO.03

Crowdsourcing Moral Machines

Editing Self-Image

Algorithms to Harvest the Wind

The BBC micro:bit

Spotify Guilds

Four Internets

Pivot Tracing

Association for Computing Machinery

volume 01

number 01

FIRST ISSUE PUBLISHED

Digital Government: Research and Practice is now available in the ACM Digital Library

NUMBER 01 VOLUME 01

acm Association for Computing Machinery

Digital Government

RESEARCH & PRACTICE • 2020

Open Access

dgov.acm.org

Digital Government: Research and Practice (DGOV) is ACM's new open access journal on the potential and impact of technology on governance innovations and its transformation of public institutions. DGOV promotes applied and empirical research from academics, practitioners, designers, and technologists, using political, policy, social, computer, and data sciences methodologies.

acm Association for Computing Machinery

https://dgov.acm.org

East Asia and Oceania

Regional Special Section in April 2020 Issue

***Communications of the ACM*'s regional special sections—designed to spotlight a region of the world with the goal of introducing readers to new voices, innovations, and technological research—will feature emerging research and the latest technical advances from East Asia and Oceania next month.**

This region includes Japan, Korea, Taiwan, South East Asia (Singapore, Malaysia, Indonesia, Brunei, Vietnam, Thailand, Myanmar, Philippines, Laos, Cambodia), and Oceania (Australia, New Zealand, Papua New Guinea, Fiji, Melanesia, Polynesia, Micronesia).

The section includes a dozen articles that explore the technologies from the region drawing the greatest investment, adoption, and future potential.

Some of the topics on tap include:

- **The commercialization of 5G services;**
- **Digitally enabled healthcare ecosystems;**
- **Singapore's quest to achieve a fully smart nation;**
- **Flagship research projects throughout the region;**
- **Advances in cybersecurity, data analytics, and finance technologies;**
- **Technologies for preserving cultural heritage; and,**
- **Tracing significant government investment in artificial intelligence technologies.**

acm

Association for
Computing Machinery

COMMUNICATIONS OF THE ACM

Departments

Last Byte

News

13

Viewpoints

PHOTO BY JUAN ENRIQUE DEL BARRIO

03/2020
VOL. 63 NO. 03

Practice

Articles' development led by acmqueue
queue.acm.org

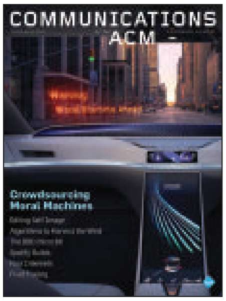
Contributed Articles

Watch the authors discuss this work in the exclusive *Communications* video. https://cacm.acm.org/videos/crowdsourcing-moral-machines

Watch the authors discuss this work in the exclusive *Communications* video. https://cacm.acm.org/videos/spotify-guilds

Review Articles

Research Highlights



About the Cover:
This month's cover story explores how to build intelligent machines into moral machines. Case in point: Design autonomous vehicles that respond to emergencies with intelligent and ethical aptitude. As the authors of "Crowdsourcing Moral Machines" contend, it is a challenge that takes a village. Cover illustration by Kollected Studio.

IMAGE BY VECTOR FUSION ART

Association for Computing Machinery
Advancing Computing as a Science & Profession

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.

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.

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
Editorial Assistant
Danbi Yu

Art Director
Andrij Borys
Associate Art Director
Margaret Gray
Assistant Art Director
Mia Angelica Balaquiot
Production Manager
Bernadette Shade
Intellectual Property Rights Coordinator
Barbara Ryan
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
1601 Broadway, 10th Floor
New York, NY 10019-7434 USA
T (212) 626-0686
F (212) 869-0481

Advertising Sales Account Manager
Ilia Rodriguez
ilia.rodriguez@hq.acm.org

Media Kit acmmediasales@acm.org

Association for Computing Machinery (ACM)
1601 Broadway, 10th Floor
New York, NY 10019-7434 USA
T (212) 869-7440; F (212) 869-0481

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.

COMMUNICATIONS OF THE ACM
(ISSN 0001-0782) is published monthly by ACM Media, 1601 Broadway, 10th Floor New York, NY 10019-7434 USA. Periodicals postage paid at New York, NY 10001, and other mailing offices.

POSTMASTER
Please send address changes to
Communications of the ACM
1601 Broadway, 10th Floor
New York, NY 10019-7434 USA

Printed in the USA.

Association for Computing Machinery

PLEASE RECYCLE THIS MAGAZINE

DOI:10.1145/3381047

Moshe Y. Vardi

Advancing Computing as a Science and Profession—But to What End?

FOUNDED IN 1947, the Association for Computing Machinery (ACM) is the oldest educational and scientific society dedicated to the computing profession. With over 100,000 members around the world it is also the largest. According it its 1947 Certificate of Incorporation, the purpose of the association was to "advance the science, design, development, construction and application of modern machinery and computing techniques, for performing operations in mathematics, logic, statistics, accounting, automatic control, and kindred fields." The narrowness of this purpose was recognized in the ACM Constitution, last changed in 1998, whose Article 2 offers the purpose of "advancing the art, science, engineering, and application of information technology, serving both professional and public interests by fostering the open interchange of information and by promoting the highest professional and ethical standards." ACM's website at *acm.org* offers yet a broader description of ACM's purpose, stating: "Advancing Computing as a Science & Profession—We see a world where computing helps solve tomorrow's problems, where we use our knowledge and skills to advance the profession and make a positive impact."

One can clearly see a growing commitment to the public good between the Certificate of Incorporation, the Constitution, and the descriptive text on ACM's website. While the latter text is nonbinding and could be seen as "marketing," the Preamble of ACM's Code of Ethics states: "Computing professionals' actions change the world. To act responsibly, they should reflect upon the wider impacts of their work, consistently supporting the public good." So ethical computing professionals have a responsibility to support the public good. But what is ACM's responsibility to the public good?

This year, we celebrate the 75th anniversary of "*Science, The Endless Frontier*," a highly influential report submitted in July 1945 to the President of the United States by Vannevar Bush, an American engineer and science administrator, who during World War II headed the U.S. Office of Scientific Research and Development, through which almost all wartime military research and development was carried out. The report, which led to the establishment of the U.S. National Science Foundation, argued that scientific progress is essential to human progress: "Progress in the war against disease depends upon a flow of new scientific knowledge. New products, new industries, and more jobs require continuous additions to knowledge of the laws of nature, and the application of that knowledge to practical purposes. Similarly, our defense against aggression demands new knowledge so that we can develop new and improved weapons." Bush argued, "this essential, new knowledge can be obtained only through basic scientific research" and is "the pacemaker of technological progress." As such, he concluded it is the role of the Federal Government to support the advancement of knowledge. His philosophy can be summarized in one phrase: "Science for the public good."

Bush's 1945 vision was recently revisited in the article "Science Institutions for a Complex, Fast-Paced World,"[a] by Marcia McNutt, president of the National Academy of Sciences, and Michael M. Crow, president of Arizona State University. Writing in *Issues in Science and Technology*, McNutt and Crow point out that "today's understanding of how knowledge, innovation, economic growth, and social change are all intimately interdependent is something of which Bush—and his world—had barely an inkling." Building on that, they note, "In the past 75 years, the challenges—from nuclear proliferation to climate change to wealth concentration to social media's impact on expertise and truth—that have resulted, at least in part, from society's application of scientific advances are now subjects that science itself must directly help to solve."

McNutt and Crow stress the institutions that carried out much of the scientific progress over the past 75 years must re-assess their mission and be committed not only to advancing scientific knowledge but also to addressing the societal problems that technology, driven by scientific knowledge, has created. In other words, the commitment to "science for the public good" should be to pursue the public good via science.

Computing professionals, like their colleagues in the sciences, must also accept the challenges of our era. It is time, in other words, to revisit and update the purpose of ACM. It is not enough to focus on science and profession. ACM's purpose must be "to advance the science and profession of computing for the public good." A vigorous discussion and debate on how best to work toward this purpose must now begin.

Follow me on Facebook and Twitter. C

[a] https://issues.org/science-institutions/

Moshe Y. Vardi (vardi@cs.rice.edu) is the Karen Ostrum George Distinguished Service Professor in Computational Engineering and Director of the Ken Kennedy Institute for Information Technology at Rice University, Houston, TX, USA. He is the former Editor-in-Chief of *Communications.*

DOI:10.1145/3380448

Conferences and Carbon Impact

MOSHE VARDI MAKES an excellent point in his January 2020 column in noting we, as a community, should do more to reduce carbon emissions and suggests ACM conferences do more to support remote participation. While I share his concern about carbon emissions, I have several concerns about his proposals for conferences.

First, time zones often make it difficult to participate in remote events, a problem that is also often faced by members of a distributed development team. At home, I'm nine hours behind Western Europe and about 12.5 behind India, so I would have to join late at night in both cases. That is just not a workable solution for a multiday conference.

Second, my own teaching experience during the past 15 years (plus countless faculty meetings) has repeatedly demonstrated that remote participants are less involved. Maybe they are trying (unsuccessfully) to multitask, but it is simply more difficult for remote attendees to ask questions or join a discussion unless it is a virtual event where everyone is remote and there is a moderator who recognizes participants in turn.

Third, the experience with online courses (Udacity, edX, among others) suggests material should be presented differently to a remote audience than to a local one. Khan Academy has long taught in 10-minute snippets, perhaps in recognition of the shorter attention spans of its audience. Personally, a brief illness last year caused me to deliver a keynote address remotely. Even though I cut my talk down to half of its original length and used slides, there were fewer questions and less discussion than I would have expected.

Fourth, it's important for aspiring and junior faculty to personally meet the senior faculty in their specialty F2F. Not only are they colleagues, but they are often valuable for supporting academic promotions. A connection over LinkedIn, even if accepted, falls well short of a personal connection. Vardi recognizes, and I agree, that there is an important social networking aspect to conferences that cannot be satisfied by remote participation.

Finally, conferences need to build their own community to assure their long-term success, including the leadership of future years of the conference. While it's easy to join a program committee remotely, conference and program chairs, as well as other members of the organizing committees, are more likely to come from repeat attendees who have developed personal relationships with conference organizers.

In summary, I'm trying to do my part (home solar panels, electric car) to reduce my carbon impact, but I think there are some difficult issues with Vardi's proposal. I hope that we can continue the important discussion about our impact on the environment and find some alternative solutions that can address the issues raised here.

Anthony I. Wasserman,
Moffett Field, CA, USA

Author's response

Quoting from my column: "Of course, conferences are more than a paper-publishing system. First and foremost, they are vehicles for information sharing, community building, and networking. But these can be decoupled from research publishing, and other disciplines are able to achieve them with much less travel, usually with one major conference per year. Can we reduce the carbon footprint of computing-research publishing?"

Reducing our carbon footprint is an existential imperative. We cannot blindly cling to the way we have been doing things. For some fresh thinking, see, for example, http://uist.acm.org/uist2019/online/

Moshe Y. Vardi, Houston, TX, USA

Response from the Editor-in-Chief

The idea that the field of computing could reduce its carbon impact by reducing the prominence of conferences and adopting practices from a number of other scientific fields is a good one, and I applaud Vardi's column, Wasserman's response, and other efforts recently highlighted in Communications *(for example, see Pierce et al. on p. 35 of this issue.)*

But if the cause of reducing computing's carbon footprint excites you, recognize that conference travel is a pittance when compared to the negative climate impact of computing's power consumption. Our research collaborators' work of 2019 datacenter global power consumption estimates are nearly double earlier estimates—now 400 TWh! These numbers are a large multiple higher than the best projections based on 2013 data.[3] There has been an important major change. These numbers are shockingly large—and worse—they are growing fast. Recent press about hyperscale cloud reveal growth rates of perhaps 40% per year.[2]

For more, see my broader call to action[1] for computing professionals to address computing's growing and problematic direct *environmental impact. Let's all get moving on this!*

References

1. Chien, A. Owning computing's environmental impact. *Comm. ACM 62*, 3 (Mar. 2019), 5.
2. Kniazhevich, N. and Eckhouse, B. Google tops green-energy buys, BlackRock seen jogging new growth. *Bloomberg Green* (Jan. 28, 2020); https://bloom.bg/31qdPNt.
3. Shehabi, A. et al. United States Data Center Energy Usage Report. LBNL, June 2016.

Andrew A. Chien, Chicago, IL, USA

Reducing Biases in Clinical Prediction Modeling

In "Algorithms, Platforms, and Ethnic Bias" (Nov. 2019), Selena Silva and Martin Kenney visualized a chain of major potential biases. The nine biases, which are not mutually exclusive, indeed must be considered in the design of any data-driven application that may affect individuals, especially if the biases have the potential to negatively affect a person's health condition.

Users may be slightly affected if they are exposed to irrelevant online advertisements or more greatly affected if they are unjustifiably refused a loan at the bank. Even worse would be a poorly designed algorithm that can cause a physician to make a decision that may be harmful to patients. An outdated risk-assessment algorithm can significantly affect many individuals, especially if broadly used. An example of such an algorithm is the Model for End-Stage Liver Disease (MELD) score, a risk-assessment algorithm for the liver that has been in use worldwide since 2002. The score was designed based on data captured from an extremely small group of patients and had only three laboratory covariates, which were manually selected, eliminating other potentially predictive covariates, such as age and other labs, incorporated into the MELD-Plus score in 2017.

Reduction of biases in the design of clinical prediction modeling is crucial. To achieve such a reduction, it is necessary to precisely define the outcome to be predicted; when defining the exact occurrence of a diagnosis or exacerbation of a condition, relying on diagnosis codes alone may result in inaccuracy, as has been widely discussed in the medical literature. The date of exacerbation in heart failure, for example, must be defined by at least two independent data elements that are closely captured in time, such as a diagnosis code date and a diuretic prescription, as opposed to merely capturing an admission associated with the condition with no clear evidence that the primary reason for admission was the patient's worsening heart. To avoid such biases, for example, Khurshid et al.[1] combined multiple data elements to identify the onset of atrial fibrillation.

To reduce biases even further, another approach would be to avoid using subjectively selected elements. For example, there is great variability in how physicians use diagnosis codes to document conditions such as hypertension and type-2 diabetes; such conditions could be defined more precisely based on actual lab values (for example, A1C and blood pressure) rather than relying on diagnosis codes alone. Furthermore, although it is widely known that genetic as well as behavioral variabilities exist across ethnicities and regions of residence, such data elements must be used with caution when incorporated into predictive risk scores because these factors are not objectively measured as labs and may be coincidental relative to a medical outcome and not serve as reliable predictors.

Reference

1. Khurshid, S., Keaney, J., Ellinor, P.T., and Lubitz S.A. A simple and portable algorithm for identifying atrial fibrillation in the electronic medical record. *Am. J. Cardiol.* (2016).

Uri Kartoun, Cambridge, MA, USA

Where Good Software Management Begins

Bertrand Meyer's critique on a project's critical path and Brook's *Mythical Man Month* is so laced with pejorative themes (Blog@CACM, Jan. 2019); his basic thought that heuristics and mathematical models should always be tailored to the situational context is only laboriously revealed. Mocking and ridiculing the work of earlier practitioners negates one's own ideas, as we all build on yesterday's results.

Brook's insight is not a law, but a heuristic based on the simple mathematical formula that calculates the variable possible number of channels (Edges) of communications between a given number of people (nodes): $C = \{ N(N-1) \} / 2$.

Whether ineffective managers blindly throw additional money and resources at a project ('Crashing a project' in project management nomenclature) is not a fault of Brook's insight, but a misapplication of the principle.

The Project Management Institute (PMI) has a well-documented Body of Knowledge (PMBOK) including earned value management (EVM), a suite of simple formulas using a common $cost unit of measure across both time and cost units.

One of the initial guidance principles of PMI and systems engineering is that the rigor and scope of the use of the tools should always be tailored to the particular effort; in other words, you don't need a shotgun when going to an arm-wrestling contest.

Good software engineering management always calls for intelligent application and balance of cost, scope, and time. If you constrain any one side of this triple constraint, the other two will flex. It's not rocket science. And, if anything 40 years old is obsolete, we may as well drop Euclidian geometry, after the advent of Einstein's work and non-Euclidian geometry.

Michael Ayres, San Francisco, CA, USA

Author's response

I am not sure Ayres paid enough attention to what my blog actually says. It is not a "critique" and does not mock anyone. It is the reverse of "pejorative," that is to say, it is actually laudatory: it brings to the attention of the Communications *readership, particularly software project managers, the importance of a key result reported in Steve McConnell's 2006 book, pointing out it deserves to be better known. This is its plain goal, not "that heuristics and mathematical models should always be tailored to the situational context" (which, if I understand this sentence correctly, is probably true but not particularly striking and not what I wrote).*

"Brooks' insight is not a law:" True, that's indeed what my article says, but "Brooks' Law" is what Brooks himself called it when he introduced it in The Mythical Man-Month.

"Anything 40 years old is obsolete:" Of course not, nor did I imply anything like this. Same thing for the blaming of Brooks' Law for ineffective managers; my article makes no such representation.

I guess Ayres's main goal is to highlight the value of the PMBOK, a recommendation that I am happy to endorse.

Bertrand Meyer, Zürich, Switzerland

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:10.1145/3379491 http://cacm.acm.org/blogs/blog-cacm

Coding for Voting

Robin K. Hill explains the ethical responsibility of the computing professional with respect to voting systems.

Robin K. Hill

Voting, Coding, and the Code

http://bit.ly/2t5QQe5
November 27, 2019

Our profession is to be commended for taking steps toward the establishment of computing ethics. They may be baby steps (akin to unstable toddling accompanied by incoherent babble) or perhaps tween steps (akin to headlong running accompanied by giggles, tumbles, and sobs), but steps they are. Let's consider a fundamental process critical to democracy: Voting. The author is inspired by the sesquicentennial, on December 10th, of the passage of the suffrage act in Wyoming, granting women the right to vote and to hold office. Wyoming was a territory at the time, the first known government body to pass general and unconditional (and permanent) female suffrage well before the 19th Amendment granting national suffrage, and entered the Union in 1890 as the first state where women could vote.

What is the responsibility of the computing professional with respect to voting systems? The obvious criteria are accuracy in recording and tallying, reliability in uptime, and security from malicious intervention; all of these are needed for the promotion of trust. Let's probe deeper. This is not about voting laws, or districts, or methods,[2] all rich fields of inquiry in their own right. This is about voting procedures as reflected in the design and implementation of software and hardware. Of special concern is voting with electronic assistance. The scope here is the *election system* as defined by the National Academies report[5] [page 13, footnote 5]—roughly, a technology-based system for collecting, processing, and storing election data. A special issue of this publication[3] in October 2004 carried several articles on this subject still worth reading, including the rejection of the SERVE system[4] that put a stop to the optimistic network-voting plans of the time. This discussion also will refer to sections of the ACM Code of Ethics, as a means of taking the Code out for a spin.[1]

Musing on the peculiarities of voting in the abstract suggests a vote is symbolic, discrete, and devoid of connotation; not an act of communication, but an act of declaration, single-shot, unnegotiated, unilateral. Should it exist as an entity; should a vote be preserved somehow? On paper, it does exist as a tally mark. A poll worker could point to it, and even associate it with other descriptions ("the eleventh one" or "the ballot with the bent corner"). A vote may be open to construal as a first-class artifact (existing on its own, subject to creation, destruction, examination, and modification) that lacks a description or identifier *by design*. First-class objects can be passed as parameters; votes are passed to tallying functions. First-class objects can be compared for equality; that is the salient feature of votes—sameness to or difference from other votes, a stark quality. The voter must give an all-or-nothing choice on each question, no hedging allowed. The hierarchy is flat. All votes count equally, so three votes cast in one polling place should be handled as carefully as thousands from another.

Now to take on the responsibilities of the computing professional, let's outline those at play before coding starts.

First responsibility of the computing professional: To understand why trust in voting is critical. Democracy relies on voting to reveal the collective will of the electorate. In the long view, as in the ethics of care,[7] background matters and situations cannot be assessed in the moment, but must be viewed in a wider scope in time and place. The National Research Council published a report in 2006 remarking, "...although elections do determine in the short run who will be the next political leaders of a nation (or state or county or city), they play an even greater role in the long run in establishing the foundation for the long-term governance of a society. Absent legitimacy, democratic government, which is derived from the will of the people, has no mandate to govern."[6] The report goes on to make the important point that elections must, in particular, satisfy the losers, preserving the trust that allows them to tolerate the policies of the winners. Code 2.1: "Professionals should be cognizant of any serious negative consequences affecting any stakeholder…" Under American standards, loss of faith in democratic government would be a serious negative consequence.

Second responsibility: To know the criteria for an acceptable election system. These criteria include, as examples, that voting should be easy for everyone; that ballots should present all candidates neutrally; that tallying should be computable by the average person; that audits should be possible. Privacy should be secured under all circumstances (Code 1.6: "Respect privacy," and 1.7: "Honor Confidentiality"). The result should be dictated by all and only the exact votes cast. Other sources may give somewhat different criteria, but major standards are accepted universally. Life-support systems demand high reliability. Military systems demand high security. Financial transactions demand high accuracy. Voting demands all of those. Security looms over all of the Code, and is explicitly mentioned in 2.9: "Design and implement systems that are robustly and usably secure." Accuracy, which must also loom over the Code, is not mentioned explicitly. Surely generating wrong answers is the worst transgression of a computing professional. References to quality of work must be intended to cover accuracy or correctness (Code 2.1, 2.2), as well as basic standards of maintainability, efficiency, and so forth, but we might ask whether correctness is a responsibility that transcends these others.

Next responsibility: To interrogate all circumstances, to appreciate the complications, and to acknowledge that unanticipated circumstances will arise. An election system involves many steps of preparation, execution, and resolution, from ballot design and training of poll workers to delivering recounts (and improving procedures for the next election). Complications are rooted in the real-world setting, and the peculiar status of a vote as anonymous but distinct artifact. Code 2.2: "Professional competence starts with technical knowledge and with awareness of the social context in which their work may be deployed." Our county clerk's staff will carry a ballot outside to a car (advance notice requested) for those who cannot easily walk into the polling place. Does that affect the rest of the election system? Code 2.3: "Know and respect existing rules pertaining to professional work." This could mean the entire local voting code and protocols. If one race is over-voted, does that invalidate the whole ballot? How should a write-in be detected? Under what circumstances is a ballot provisional? If the wind blows a ballot out the window onto a piece of charcoal that marks it, or under a car tire that punches it, after its assignment to a voter, how is it replaced? Anecdotes in electoral research describe exceptions to the notions conscientious voters mark ballots unambiguously, and error-free methods tally those votes.[8] An election system must accommodate every non-standard circumstance. Voting is a domain where no data point can be dismissed as "in the noise."

Thus prepared, the computing professional can perform the hardware and software design, coding, and testing. All of the Code applies. Afterward, there are other professional obligations.

Final responsibility of the computing professional: To announce and explain vulnerabilities, errors, quirks, and unknowns, and to suggest solutions. This responsibility is in service to the main one, trust. Demonstrated full disclosure is the best way to instill confidence that, in the face of no disclosure, nothing bad is happening. Code 2.5: "Computing professionals are in a position of trust, and therefore have a special responsibility to provide objective, credible evaluations and testimony to employers, employees, clients, users, and the public." Code 3.7: "Continual monitoring of how society is using a system will allow the organization or group to remain consistent with their ethical obligations outlined in the Code."

As a hypothetical, let's think of a software engineer who notices the tally is incorrect by a small number of votes that exactly offset each other, an error that makes no difference to the tally, nor to the outcomes of any races. Should that flaw be debugged internally? Of course. Should the incident be made public? Yes, because any problem may result in future distortion, which brings this situation under the requirement of Code 1.2: the "obligation to report any signs of system risks that might result in harm." It should be made public as a demonstration that votes are prioritized above tallies. The vote is primary; the tally is derivative. This may have unpleasant repercussions to the programmer, but ethical professionals sacrifice themselves before they sacrifice voters.

These responsibilities apply to all who have a hand in American voting, not just computing professionals. Everyone involved should mind Code 2.9: "In cases where misuse or harm are predictable or unavoidable, the best option may be to not implement the system." The latest National Academies report, among several specific recommendations ranging over many aspects of election systems, recommends the Internet not be used for submitting ballots.[5]

This observer (who claims high interest but shallow expertise) concludes voting turns out to be more complicated than was thought in the early days when electronic procedures were broached. Even though it appears to be counting—the simplest computation of all—voting is a process not amenable to automation except where subordinate to the judgment of election officials. We see the ACM Code of Ethics provides broad but cogent guidance for this computing activity, although we would like to see accuracy incorporated explicitly. C

References

1. ACM Code 2018 Task Force. June 22, 2018. ACM Code of Ethics and Professional Conduct. Association for Computing Machinery, https://www.acm.org/code-of-ethics.
2. Brandt, F., Conitzer, V., Endriss, U., Lang, J., and Procaccia, A., editors. 2016. Handbook of Computational Social Choice, Cambridge University Press.
3. *Commun. ACM, 47*, 10 (Oct. 2004).
4. Jefferson, D., Rubin, A.D., Simon, B., and Wagner, D. 2004. Analyzing Internet Voting Security. *Commun. ACM, 47*, 10 (Oct. 2004)
5. National Academies of Sciences, Engineering, and Medicine and others. 2018. Securing the Vote: Protecting American Democracy. National Academies Press.
6. National Research Council and others. 2006. Asking the right questions about electronic voting. National Academies Press.
7. Sander-Staudt, M. No publication date given; accessed 24 November 2019. Care Ethics. Internet Encyclopedia of Philosophy and its Authors. ISSN 2161-0002.
8. Wikipedia contributors. 2019. Spoilt vote—Wikipedia. Online; accessed 27 November 2019.

Robin K. Hill is a lecturer in the Department of Computer Science and an affiliate of both the Department of Philosophy and Religious Studies and the Wyoming Institute for Humanities Research at the University of Wyoming. She has been a member of ACM since 1978.

N news

Science | DOI:10.1145/3379493

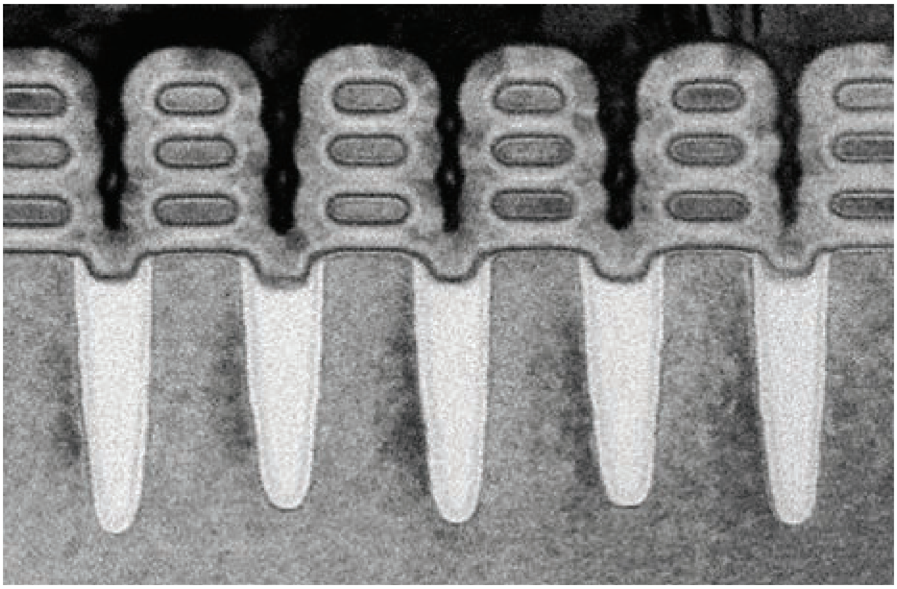
Samuel Greengard

Can Nanosheet Transistors Keep Moore's Law Alive?

The technology promises to advance semiconductors and computing, but also introduces new questions and challenges.

The computing world has always relied on advances in semiconductors. Over the decades, smaller and more efficient transistor designs have produced faster, more powerful, more energy-efficient microchips. This has fueled incredible advances in everything from supercomputing and clouds to smartphones, robotics, virtual reality, augmented reality, additive fabrication, and the Internet of Things (IoT).

The march toward more sophisticated microprocessors has continued unabated for decades. However, Moore's Law, which states the number of transistors in an integrated circuit doubles approximately every one-and-a-half to two years, has begun to slow in recent years. The reason? It has become more difficult to use MOSFET (metal-oxide-semiconductor field-effect transistors) scaling techniques to achieve continued miniaturization. Many chips now contain 20 billion or more switches. Engineers are running into enormous challenges as they reach the physical limits of existing technology.



A 2017 scan of the IBM Research Alliance's 5nm silicon nanosheet transistor containing 30 billion switches.

IMAGE COURTESY OF IBM

However, an emerging technology promises to change the equation. Nanosheet transistors, which also go by the names *gate-all-around*, *multi-bridge channel*, and *nanobeam*, push beyond today's 7-nanometer (nm) node and into more-advanced 5 nm designs with performance boosts of approximately 40% and power consumption cuts of 75%. Samsung announced in May last year it had perfected nanosheet transistors and would be introducing them commercially in the first half of this year. "It's a huge advance in the device structure itself. It will enable significant advances in computing," says Mukesh V. Khare, a vice president at IBM Research.

Miniaturization Matters

Moore's Law has served the semiconductor industry well since Intel cofounder Gordon Moore introduced the idea in 1965. Just over a half-century later, transistor designs appear to finally be approaching their physical limits—at least using current materials and designs. "We are reaching a quantum threshold where the transistors cannot get a lot smaller and we cannot keep on achieving gains at the speed of Moore's Law," explains Peide Ye, Richard J. and Mary Jo Schwartz Professor of Electrical and Computer Engineering at Purdue University.

Current transistors use a time-proven design based on MOSFET technology, which has been in use since 1959. While shapes and materials have advanced and changed over the years, the basic engineering remains the same. The design incorporates a gate stack, channel region, source electrode, and a drain electrode. The structure is designed to transport positive (p-type) or negative (n-type) charges. Together, they produce an integrated circuit (IC) needed for the complementary metal–oxide–semiconductor (CMOS) technology that powers computers and mobile phones.

Today's designs place the gate stack directly above the channel area. The metal gate stack sits atop a dielectric material that conducts an electric field into the transistor channel region to accumulate or block charges that could flow through. In basic terms, this allows current to flow across the transistor and switch on and off as needed. The problem is that as these structures become smaller, it becomes more difficult to block the charge leak across the transistor. The resulting leakage leads to hotter, less power-efficient microchips. Engineers have approached this problem by making the channel region thinner and thinner.

Fin Field Effect Transistor (FinFET) technology is used in virtually all of today's processors. It incorporates stacked sheets and a channel region that is tilted upward (think of it as a wall) to create a wider path for current. The gate and dielectric are placed over the fin so that it is surrounded on three sides instead of just one; this helps reduce current leakage. These three-dimensional (3D) designs, used by major semiconductor manufacturers, have shrunk from about 22 nm in 2011 to between 7 nm and 5 nm today. Unfortunately, they cannot be built at the 3-nm scale and accommodate current switching methods. "The leakage and power drain are simply too much for the technology to be viable at this scale," says Dan Hutcheson, CEO of VLSI Research, Inc., a market research and consulting firm.

> **"Nanosheet transistors are creating a new ecosystem for device structure, modeling, process technology, and various materials."**

For years, researchers and engineers have known they were approaching the end of the road for current transistor designs. Although myriad tweaks, advances, and trade-offs have led to ongoing advances in central processing units (CPUs), graphics processing units (GPUs), and other chips, the need for radically different designs was completely apparent. Nanosheets extend performance by removing material between layers of other material and filling in the gaps with both metal and dielectric.

This leads to a smaller-scale design. What is more, "The gate is wrapping around all four sides of the silicon and the silicon channel thickness scaling is controlled by epitaxial growth, which moves things beyond nanometer control and into atomic level control," Khare explains.

Beyond Silicon

At the heart of nanosheet transistors are new materials and radical design changes. Gary Patton, CTO and head of worldwide research and development at GlobalFoundries, has described them as "a smaller, faster, and more cost-efficient generation of semiconductors." The technology, which IBM began researching in 2006 and which took shape under a public-private industry alliance, essentially creates a device architecture with stacked layers of silicon sheets by retaining the silicon layers from a superlattice structure that consists of alternating crystal layers of silicon and silicon germanium.

The significance of these new materials and designs, such as germanium, should not be minimized. Chipmakers have been forced to reduce clock speeds because of the enormous heat produced by high transistor density. However, by incorporating new materials and designs, it is possible to replace several slower processor cores with a single chip that operates as fast while generating less heat. In some cases, electrons can move more than 10 times faster in these semiconductor designs.

Nanosheet technology represents a remarkable advance in transistors. "These nanosheet layers are patterned lithographically to form gates that wrap around the junction between the source and drain by etching away unwanted material. This is done multiple times to form structures that look something like the center of a layer cake cut in thirds," Hutcheson explains.

It is possible to place upward of 30 billion switches on a fingernail-sized chip. The gate surrounds the channel region in its entirety to deliver greater control than FinFET. This "stacked" structure supports far more advanced semiconductor fabrication processes. "When the industry figured out how to use certain chemistries to lay down substances at a single molecular level and then place others on top of it, the manufacturing process advanced radically," Hutcheson says. "They were no longer painting on a thick surface. They could control the deposited material to a single atomic layer."

The Endura Clover system from Applied Materials, for example, can apply up to 30 layers within a single stack only a few angstroms thick. This ensures an extremely high level of production quality.

To be sure, "Nanosheet transistors are far more than a technology iteration. It is extending Moore's Law for several more years. In fact, the design framework surrounding nanosheet transistors will allow researchers and engineers to develop even more advanced transistors and standard cells than FinFET technology allows, including flexibility in circuit design," Khare says. "The industry is converging around this device structure and it is moving forward with fabs and production. Nanosheet transistors are creating a new ecosystem for device structure, modeling, process technology, and various materials."

Of course, the transition will not happen overnight. The technology will require entirely new fabs and changes in distribution channels. "The cost of a new fab is in the $20-billion range, so it isn't something to take casually. There's an enormous amount of money and planning that must go into their transition," says Purdue's Ye.

While the first nanosheet transistors likely will appear from Samsung some time this year, it may take several more years before production scales up to support widespread adoption. Only Intel, Samsung, and Taiwan Semiconductor Manufacturing Co. (TSMC) have the means to handle this level of miniaturization. It is far more complicated than updating existing fabs. Chipmakers must build entirely new fabs with equipment and systems to handle the specialized nanosheet construction, at a cost that can reach $20 billion, Ye says.

Ultimately, the question is not whether nanosheet technology will impact the market, but rather *when* and *how*. "There's no way to know when we will hit the crossover point and nanosheet transistors will become the dominant technology," Khare says. "There are a lot of technical and economic issues that intersect with it. What's clear is that we will see products emerging within a couple of years and they will impact many aspects of computing, from devices and datacenters to the edge of the network."

Make no mistake, nanosheet transistors will lead to more powerful devices that utilize power far more efficiently—a key consideration in an era where battery life matters, energy costs are exorbitant, and climate change concerns are growing. The technology will introduce new and more advanced capabilities, particularly in the artificial intelligence arena, where advances in computing power can fuel exponential gains. Says Hutcheson, "We will have transistors that can handle heavy-duty artificial intelligence. The impact will ripple out to datacenters, smartphones, self-driving cars, and many other areas."

Designs on the Future

Nanosheet transistors will shape the semiconductor industry for years to come. The technology takes aim at a fundamental problem, Khare says. "Integrated circuits (IC) have been stuck at the same power density for about a decade. It's been impossible to remove more than about 100 watts per square centimeter." Chip designers have focused on keeping heat buildup down, including limiting clock speed to 4 gigahertz or less and using slower multi-core designs that substitute a more-powerful single processor, but generate much less heat.

Nanosheets can break through this barrier with a more efficient transistor design combined with new material, like germanium. It could push the ranges further for power and energy consumption. This addresses a major problem in semiconductors: "As feature sizes shrink, conventional methods of manufacture fail to produce devices that work well electrically," Hutcheson says. "With the tri-gate structure used on current devices, they can fail to switch on or off, because there is not enough surface area contacted by the gate, hence the need to wrap all four sides. There can also be power dissipation problems due to leakage. The reason why new materials are needed is that silicon can't be deposited over a dielectric in a properly oriented crystalline form to form the junctions."

The nanosheet technology also opens up new possibilities and opportunities within the semiconductor field, especially when combined with new materials. These design improvements also create more favorable economics for manufacturing because today's technology is too expensive to produce with some of these materials, Hutcheson says.

In order to bump up clock speeds, Ye and others say it is necessary to produce more powerful and energy-efficient transistors than silicon alone can deliver. Consequently, he and others continue to research different materials and designs that can be used in the channel region. This includes germanium, as well as semiconductors built from indium gallium arsenide (InGaAs). Other researchers are exploring how combinations of germanium, indium arsenide, and gallium antimonide can offer even greater efficiencies in nanosheets and other semiconductor designs.

Researchers have found that electrons can move up to 10 times faster within these more-advanced semiconductors. The end result is chips that not only switch faster, but also operate at much lower voltage levels—thus enabling new types of functionality and features. These designs likely will introduce capabilities that we can't imagine today. For now, chipmakers are sold on the concept. Most have already committed to using nanosheet transistors in their future designs.

Concludes Ye: "The combination of nanosheet transistors and advances in semiconductors will carry us far into the future. The technology will have a significant impact on computing." C

Further Reading

Ye, P., Ernst, T., and Khare, M.V.
The last silicon transistor: Nanosheet devices could be the final evolutionary step for Moore's Law, *IEEE Spectrum*, Volume 56, Issue 8, August 2019. https://ieeexplore.ieee.org/abstract/document/8784120

Moayed, M.M.R., Bielewicz, T., Noei, H., Stierle, A., and Klinke, C.
High-Performance n- and p-Type Field-Effect Transistors Based on Hybridly Surface-Passivated Colloidal PbS Nanosheets. *Advanced Functional Materials.* Volume 28, Issue 19, May 9, 2018. https://onlinelibrary.wiley.com/doi/abs/10.1002/adfm.201706815

Dahiya, A.S., Sporea R.A., Poulin-Vittrant, G., and Alquier, D.
Stability evaluation of ZnO nanosheet based source-gated transistors, *Nature, Scientific Reports*, Article number 2979, February 27, 2019. https://www.nature.com/articles/s41598-019-39833-8

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

Technology | DOI:10.1145/3379497

Don Monroe

Algorithms to Harvest the Wind

Wake steering can help ever-larger turbines work together more efficiently on wind farms.

WIND-GENERATED ELECTRICITY HAS expanded greatly over the past decade. In the U.S., for example, by 2018 wind was generating 6.6% of utility-scale electricity generation, according to the U.S. Energy Information Administration. The criteria for efficient design and reliable operation of the familiar horizontal-axis wind turbines have been well established through decades of experience, leading to ever-larger structures over time, both to intercept more wind and to reach faster winds higher up.

As these gargantuan turbines are assembled into large wind farms, often spread over uneven terrain, complex aerodynamic interactions between them have become increasingly important. To address this issue, researchers have proposed protocols that slightly reorient individual turbines to improve the output of others downwind, and they are working with wind farm operators to assess their real-life performance. Beyond extracting more power from current farms, widespread use of these "wake-steering" techniques could allow denser wind farm designs in the future.

Bigger Is Better

"The tendency is to build higher and higher turbines," said Mireille Bossy, a fluid dynamics expert at Inria, the French national institute for computer science and applied mathematics, located in the Sophia Antipolis technology park near Nice, France. "We are talking in a new project about 300m [about 984 feet] in height." The wake of slower disturbed air typically extends 10 or more times the diameter iof a turbine, robbing downwind turbines of wind. Completely avoiding power loss for downwind turbines would demand several kilometers between turbines, incurring substantial additional costs in real estate and wiring.

PHOTO BY JUAN ENRIQUE DEL BARRIO

These costs and the specific constraints of available sites often lead to less-than-optimal arrangements, however. Predicting the interactions is difficult, especially for farms located in terrain that may create turbulent flows. Bossy described one existing farm where the addition of a single turbine, in what seemed like a good place to minimize wake interactions, would decrease the output of the entire facility. "It's complicated," she stressed. "We cannot just do some wind-tunnel simulation," she stressed. "We need to simulate the site."

Wake Steering

Things get complicated quickly, because each combination of wind direction and speed, as well as other atmospheric details, requires a new simulation. Fortunately, the wake interactions can be reduced by "yawing" the turbines: rotating them slightly around a vertical axis, which deflects their wakes. Although a misalignment slightly reduces the power output of the upwind turbine, for some wind directions this can be more than offset by increased power from downwind turbines (which is proportional to the cube of the windspeed).

Still, the number of combinations of possible yaw angles for each turbine, as well as windspeeds and directions, quickly becomes computationally challenging. For this reason, the U.S. National Renewable Energy Labs (NREL) has developed both a calculation-intensive "large-eddy" computational-fluid-dynamics model, called Simulator fOr Wind Farm Applications (SOWFA), as well as a simpler tool for steady-state calculations called FLOw Redirection and Induction in Steady State (FLORIS).

Paul Fleming, who developed these tools with colleagues at NREL and the Delft University of Technology in The Netherlands, noted that although there is still debate about how to deal with rapidly shifting wind directions, there "seems to be some convergence toward steady-state modeling." Wind farm operators currently prefer to set a yaw angle and hold it for a while, "striking some balance between trying to keep up with the changing wind direction and trying to yaw as little as possible," he said. "Wake steering has to be built on the same structure."

A wind farm's electricity generation over the course of a year, known as annual energy production or AEP, is likely to see only small fractional increases from wake steering, in part because many wind directions would not create substantial losses in any case. For existing facilities, Fleming said, "a reasonable guess for total AEP gain is somewhere between 1% or 2%." The gains could be especially compelling for offshore generation, where winds tend to be steadier, turbines larger, and wakes more persistent, but land-based sites can benefit as well.

This improvement may seem modest, but it could amount to millions of dollars of revenue for very little cost. "It's garnered a lot of interest" from the industry, Fleming said. Indeed, at a September 2019 meeting attended by major wind developers, he said, "There was pretty broad agreement that something like this will be adopted more widely."

Out in the Field

To get this kind of buy-in, "field tests are critical," Fleming said. For this reason, he and his colleagues have worked with manufacturers, including NextEra, which he said is "the largest owner of turbines in the U.S.," to conduct field trials that have validated the simulation predictions. For one unusually close pair of turbines, spaced approximately three times the diameter of the turbines apart, the power from the downwind turbine for the worst wind direction was increased by about 14% when the upwind turbine was yawed to deflect the wake. This deflection produced in an overall 4% increase for the pair.

"Right now, the algorithms we're implementing aren't very complicated; they're essentially a lookup table" of yaw offsets for a particular windspeed and direction, Fleming said. Over time, as the technique proves its value, he expects these algorithms can be refined.

John Dabiri, now at the California Institute of Technology, recently explored one such refinement with colleagues, and followed it up with field experiments. "What we were aiming for was to do site-specific optimization: for a given layout, a given terrain, a given location where the wind conditions are what they are, and to be able to incorporate historical data in a way that informs a physics model."

Other researchers have used such historical data, capturing how much energy each turbine generated under various conditions with no wake steering, to train machine learning models. "The challenge is that we don't typically have enough data," Dabiri said, so models can overfit the existing data but fail to generalize to different locations. He and his team combine the data with a simplified physics model to match each site. The model is efficient enough to optimize the entire set of yaw angles "on a laptop computer in a few seconds."

Dabiri's team, then at Stanford University, worked with wind farm operator TransAlta to test their optimization algorithm on a line of six turbines in Alberta, Canada. "That middle ground, between the two-turbine studies and a full wind farm, is important for us to investigate," he said, to give operators confidence about real-world operation.

"Academic research has largely been focused on numerical simulations, some wind-tunnel studies, and then, even in the field, it's typically maybe a pairwise study," Dabiri said. "We're finding there's still a pretty big leap from standard methods of investigation and what happens in a real wind farm." One concern is "secondary steering," in which a deflected wake is further modified by interactions with the downwind turbine, which is not important for just one pair of turbines.

As the researchers hoped, their algorithm increased electric output by almost 50% for slow winds directed along the line. Wake steering also significantly reduced fluctuations in power generation due to turbulence, another important consideration. However, these wind conditions are rare at this test site, so the improvement is expected to be much smaller when averaged over a year.

In evaluating long-term adoption of wake steering, operators also will need to know how it affects reliability. "Over a 10-year period of operating the turbines in this mode, what could the long-term impacts be on the blade health, et cetera?" he asked. "Those are important questions to consider."

Design for Steering

Although the results from existing farms are promising, "the bigger impact is in how we design future wind farms," Dabiri said. To date, "most wind farms are designed conservatively, such that the turbines are spaced far apart from one another," which is one reason the increases are modest.

Fleming agreed that as operators become comfortable they can mitigate wake losses, it could open "opportunities for densification of wind farms," perhaps significantly. More speculatively, there may even be ways to harness the wake interactions. "When we first modeled wake steering, it was more or less as a horizontal displacement of the wake," and the goal was to "navigate these wakes into the gaps between other turbines" Fleming said. "But when you look at the three-dimensional flow out of CFD (computational fluid dynamics), there's an additive effect to wake steering because of the generation of counterrotating vortices that persist through the flow." These vortices could suck down faster, higher-altitude winds, which he described as "different from just avoiding wake losses."

Dabiri suspects these interactions could be even more important with vertical-axis turbines, although so far such designs are less mature and reliable. "Vertical-axis turbines individually tend to be less efficient," Dabiri acknowledged, but "they perform better when they are in close proximity. We see possibilities of 10X improvement, as opposed to 10% improvement."

Even without such dramatic enhancements, however, the combination of real-time yaw-control algorithms for wake steering and simulations to improve the collective output of entire farms look to help drive the continued growth of wind farms and their implementation at high densities in previously inhospitable terrain. C

Further Reading

Howland, M.F., Lele, S.K., and Dabiri, J.O.
Wind farm power optimization through wake steering, *Proc. Nat. Acad. Sci. 116*, 14495 (2019), http://bit.ly/36FvZx2

Fleming, P., et al
Initial results from a field campaign of wake steering applied at a commercial wind farm – Part 1, *Wind Energ. Sci. 4*, 273 (2019), http://bit.ly/32jZz7J

Renewable & Alternative Energy, U.S. Energy Information Administration, https://www.eia.gov/renewable/data.php#wind

Wind Energy Research, U.S. National Renewable Energy Laboratory, https://www.nrel.gov/wind/

Don Monroe is a science and technology writer based in Boston, MA, USA.

Society | DOI:10.1145/3379495

Keith Kirkpatrick

Across the Language Barrier

Translation devices are getting better at making speech and text understandable in different languages.

"THE GREATEST OBSTACLE to international understanding is the barrier of language," wrote British scholar and author Christopher Dawson in November 1957, believing that relying on live, human translators to accurately capture and reflect a speaker's meaning, inflection, and emotion was too great of a challenge to overcome. More than 60 years later, Dawson's theory may finally be proven outdated, thanks to the development of powerful, portable real-time translation devices.

The convergence of natural language processing technology, machine learning algorithms, and powerful portable chipsets has led to the development of new devices and applications that allow real-time, two-way translation of speech and text. Language translation devices are capable of listening to an audio source in one language, translating what is being said into another language, and then translating a response back into the original language.

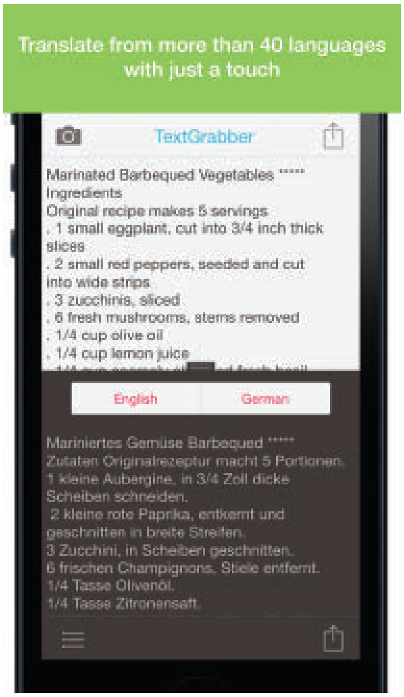
About the size of a small smartphone, most standalone translation devices are equipped with a microphone (or an array of microphones) to capture speakers' voices, a speaker or set of speakers to allow the device to "speak" a translation, and a screen to display text translations. Typically, audio data is captured by the microphones, processed using a natural language processing engine mated to an online language database located either in the cloud or on the device itself, and then the translation is output to the speakers or the screen. Standalone devices, with their dedicated translation engines and small portable form factors, are generally viewed as being more powerful and convenient than accessing a smartphone translation application. Further, many of these devices offer the ability to access translation databases stored locally on the device or access them in the cloud, allowing their use in areas with limited wireless connectivity.

Are there any recommended restaurants nearby?

¿Hay ún restaurante recomendado cerca?

Hay una deliciosa taquería al otro lado de la calle.

There's a delicious taqueria across the street.



Translate from more than 40 languages with just a touch

Marinated Barbequed Vegetables *****
Ingredients
Original recipe makes 5 servings
. 1 small eggplant, cut into 3/4 inch thick slices
. 2 small red peppers, seeded and cut into wide strips
. 3 zucchinis, sliced
. 6 fresh mushrooms, stems removed
. 1/4 cup olive oil
. 1/4 cup lemon juice

Mariniertes Gemüse Barbequed *****
Zutaten Originalrezeptur macht 5 Portionen.
1 kleine Aubergine, in 3/4 Zoll dicke Scheiben schneiden.
2 kleine rote Paprika, entkernt und geschnitten in breite Streifen.
3 Zucchini, in Scheiben geschnitten.
6 frischen Champignons, Stiele entfernt.
1/4 Tasse Olivenöl.
1/4 Tasse Zitronensaft.

IMAGES COURTESY OF POCKETALK, AND ABBYY/EVERNOTE APP CENTER

Instead of trying to translate speech using complex rules based on syntax, grammar, and semantics, these language processing algorithms employ machine learning and statistical modeling. These initial models are trained on huge databases of parallel texts, or documents that are translated into several different languages, such as speeches to the United Nations, famous works of literature, or even multinational marketing and sales materials. The algorithms identify matching phrases across sources and measure how often and where words occur in a given phrase in both languages, which allows translators to account for differences in syntax and structure across languages. This data is then used to construct statistical models that link phrases in one language to phrases in the second, which allows for accurate and fast translation.

In practice, this means devices can translate between languages more quickly than ever before by using such modeling. Incorporating high-powered processors, quality microphones, and speakers into the device, a person can carry on a real-time, two-way conversation with someone who speaks an entirely different language. These devices represent a significant increase in accuracy and functionality above manual, text-based translation applications such as Google Translate.

The advances in technology have not gone unnoticed, as the market for language translation devices is projected to reach $191 million annually by 2024, up from slightly more than $90 million annually in 2018, according to data from Research & Markets. Much of the activity is due to the growth in international travel and tourism, particularly from residents of countries where English language proficiency is relatively low.

For example, countries such as Japan, China, and Brazil feature a strong middle class with the means to travel internationally. Yet, these countries each are ranked "low" on the 2018 Education First English Proficiency Index (EPI), reflecting the challenges many travelers have when leaving their home country.

The ideal solution is for citizens to learn to speak multiple languages, according to Howie Berman, executive director of The American Council on the Teaching of Foreign Languages. "Our position has always been that technology is a complementary piece to the language learning process," Berman says. "I think language really depends a lot, it's not just on what you say, but how you say it. And, I think translation devices really do fail to pick up on a lot of the cultural cues."

However, the casual traveler may not have the time or inclination to become proficient in a new language in preparation for a tourist trip or event, like the 2020 Olympic Games in Japan, or the 2020 FIFA World Cup scheduled to be held in Qatar. For these one-off trips, Berman says, "We certainly don't expect someone going to the Olympics to enroll in multiple classes right before they go; we realize that's not feasible for everyone." Regarding modern translation devices, Berman says, "We think they're valuable tools, but we see them for what they are, as complementary tools to the classroom experience."

Still, the use of machine learning will help translators become better at understanding nuance, regional dialects, and tone. As algorithms are trained on voice data containing these characteristics of everyday speech, the accuracy and intelligence of the models will improve over time, particularly with translations between languages that do not feature similar structures or character sets.

The use of machine learning will help translators become better at understanding nuance, regional dialects, and tone.

One device that addresses these concerns is Pocketalk, a standalone translation device developed and marketed by Japanese software company Sourcenext Corp., which the company says can translate between 74 languages. Pocketalk has shipped globally more than 600,000 units of the $230 device since its debut in 2017, capturing nearly 96% of the global translation device market, according to April 2019 data from analyst firm BCN Retail.

"Pocketalk was created to connect cultures and create experiences for people that do not speak the same language, and can and should be used for both business and leisure," says Joe Miller, general manager and product lead for Pocketalk. Miller says Pocketalk's translation engines can recognize local dialects, dialect nuances, slang, and accents. "The voice translation will use an accent when speaking back the translation, not a robotic voice," Miller says.

However, like other devices designed to support live, multiple-way conversations, Pocketalk relies on a connection to the Internet to access its online language database and translation engine. Devices that feature a limited number of languages often can store these databases on the device, but devices that support dozens of languages generally require a persistent connection to a cloud database. While Pocketalk works on 4G cellular connections, devices such as Birgus' Two Way Language Translator or the ODDO AI pocket translator require the use of a Wi-Fi connection, and will not work using only a cellular connection.

Devices that require a Wi-Fi connection may not be suitable for travelers who spend a lot of time interacting with people outside of formal indoor settings, as they may not be able to access a reliable Wi-Fi signal. That drawback is less of an issue for translating devices designed for the international business user community, who utilize translation devices to conduct real-time business meetings and seminars that require two or more languages to be translated.

"Through our research we found that there was a need for a translator that is optimal for professional uses and can support multiple people easily conversing at the same time," says Andrew Ochoa, founder and CEO of Waverly Labs, creator of the Ambassador, a small over-the-ear translation device that can support up to 20 languages and 42 dialects, but which requires the use of a companion IoS or Android mobile application paired to a smartphone to function. "Whether someone is participating in one-on-one conversation, a multi-person meeting, or larger conference setting, Ambassador allows them to easily listen and communicate with their colleagues and teams."

The Ambassador incorporates a series of microphones, and combines the input with speech recognition neural networks, in order to capture speech clearly. The system also utilizes cloud-based machine translation engines built on translation models that incorporate local accents and dialects, allowing Waverly Labs to use machine learning to tune the accuracy of their devices based on regional parameters.

When traveling, not all communication is verbal. Fujitsu also offers a portable standalone translation device similar to Pocketalk, called Arrows Hello, which also includes a camera that can capture images, such as signs and menus that include foreign characters, and then display the translations of those text-based materials on its screen. Similarly, optical character recognition (OCR) technology company ABBYY offers a consumer-focused mobile app called TextGrabber that can "read" text or QR codes in more than 60 languages, then translate the words or phrases to a different target language while retaining the appropriate syntax and meaning, according to Bruce Orcutt, the company's vice president of product marketing.

"ABBYY's an OCR company, so you can imagine our bias towards converting everything text that's possible," Orcutt says. The TextGrabber app, he says, "uses multiple technologies that have evolved and developed to ultimately identify text, and then we use our OCR technology once we have identified the text." TextGrabber em-

9:41
Ambassador
Converse
Cecilia Ramos
Necesitamos garantizar que el producto cumpla con normas sanitarias.
Ben Fletcher
on't worry I'll ask my team to confirm we have all the pections in place before the container arrives to port.
Spanish
English

IMAGE COURTESY OF WAVERLY LABS

ploys machine learning algorithms to identify text within an image, applies OCR to capture that text, then applies a logic engine to clean up syntax and character misreads, such as being able to discern whether a character is a zero or the letter "O," based on context.

While TextGrabber currently does not include any functionality for capturing voice or video to aid in real-time translation, its OCR translation technology is incorporated into solutions from Microtek, Panasonic, Ricoh, Sharp, and others. Orcutt believes that in the future, devices that can handle any type of media, including audio, moving video, images, and text, will become commonplace.

"If you look at the younger generations, [those] digital first generations, they have no problem navigating these tools, as they're part of their ecosystem," Orcutt says. "And I think with the 2020 Olympics coming up in Japan, there'll be a tremendous amount of innovation in this area to help. I know the Japanese government is interested in making the Japanese market more easily navigated by tourists to make the Olympic experience better."

Clearly, technology developments in machine learning have led to devices that can provide accurate, real-time translations for people attending large, multinational-focused events such as the Olympics. Berman, however, hopes these technical achievements may spur people to take the next step and actually try to learn another language to fully understand its nuances, via a combination of technology and traditional classroom instruction.

"I think it's wonderful that these devices and these tools are elevating the status of language," Berman says. "We think [translation devices] are valuable tools, but we see them as complementary tools to the classroom [learning] experience." C

Further Reading

Brown, Peter F. et al.
"A statistical approach to language translation." COLING (1988). https://www.semanticscholar.org/paper/A-statistical-approach-to-language-translation-Brown-Cocke/2166fa493a8c6e40f7f8562d15712dd3c75f03df

Wenniger, Gideon Maillette de Buy.
"Aligning the foundations of hierarchical statistical machine translation." (2016). https://www.semanticscholar.org/paper/Aligning-the-foundations-of-hierarchical-machine-Wenniger/de12e7ecf32523ac9b480d3dab052ec5b43ebef9

What Buyers need to know about speech translation devices: https://www.youtube.com/watch?v=LUvNcp2xQqM

Keith Kirkpatrick is principal of 4K Research & Consulting, LLC, based in Lynbrook, NY, USA.

ACM Member News

DESIGNING COMMUNITIES WHERE MINDS MEET

Amy Bruckman, a professor and senior associate chair in the School of Interactive Computing at the Georgia Institute of Technology, says her interest in computer science began when she took computer science classes in high school.

Bruckman went on to earn her undergraduate degree in physics from Harvard University, her master of science degree from the Massachusetts Institute of Technology (MIT) Media Lab Interactive Cinema Group, and her Ph.D. from the MIT Media Lab's Epistemology and Learning Group.

After receiving her doctorate, Bruckman joined the faculty at the Georgia Institute of Technology, where she has remained ever since.

Her research interests include social computing, collaboration, social movements, content moderation, conspiratorial ideation, and Internet research ethics.

"I am also writing a book," Bruckman continues. "It is called *Should You Believe Wikipedia? Understanding Knowledge and Community on the Internet*," and "is based on a lot of the things I teach in my Design of Online Communities class, which I have been teaching since 1997. I am hoping to share some of what I have learned in teaching the class in the book."

She anticipates the book will be published next year, by Cambridge University Press.

Bruckman says she has "one fun dream," which is to facilitate communication between people who hold radically different political views, so they can come to understand one another better. "I don't know how to do that, but maybe I will have invented some kind of Internet game or discussion forum where that can happen."

—John Delaney

DOI:10.1145/3379918

Michael Lachney and Aman Yadav

▸ **Mark Guzdial,** Column Editor

Education

Computing and Community in Formal Education

Culturally responsive computing repurposes computer science education by making it meaningful to not only students, but also to their families and communities.

It was a cold morning in late February when Angela (pseudonym), an African American cosmetologist, arrived with a hair mannequin at a middle school in the city where her salon is located. Angela went to the main office and signed the guestbook before making her way to Brenda's classroom. Brenda (pseudonym) is a White technology teacher who has been an educator in the city for more than a decade. She has a strong passion for exposing students to educational technologies, especially those that support engineering and computer science (CS) lessons. This particular morning, she was prepared to implement a two-day programming lesson she developed with Angela and two university researchers.

The lesson used a visual programming application called Cornrow Curves (see Figure 1) that had been created by the Culturally Situated Design Tools research team (see https://csdt.org). Cornrow Curves helps teach block-based programming and transformational geometry by having young people explore an original body of African mathematical knowledge through the history and design of cornrow braids.[1] This grounds it in culturally responsive computing, an area of research and practice that, in part, is intended to confront racial and ethnic underrepresentation in CS. Culturally responsive computing challenges the idea that students' families, interests, heritages, and community contexts are barriers to learning. Alternatively, students' identities are foundational for a quality education.

For culturally responsive computing researchers and practitioners, programming for programming's sake is part of the problem of underrepresentation, as it reinforces the idea of culture-free instruction. This assumption allows for the reproduction of the dominant culture in the classroom, which in the U.S. tends to reflect White middle-class values. To create education contexts that represent more than the White middle-class status quo, culturally responsive computing seeks to "translate" Indigenous knowledges, vernacular practices, civic engagement, hacking, and culturally situated forms of entrepreneurship into CS education.[2]

As a culturally responsive computing application, Cornrow Curves has the anti-racist benefit of highlighting non-European mathematics, which is important for young people of all racial and ethnic backgrounds in demographically homogeneous or heterogeneous classrooms. When it is implemented in a school that serves African American and Black communities (for example, at Brenda's school over 50% of students identify as African American or Black) it has the added benefit of helping to broker school-community relationships. This gives local cultural experts opportunities to shape classroom curricula, which can be especially important for White teachers who are not

IMAGES COURTESY OF CULTURALLY SITUATED DESIGN TOOLS/CSDT.ORG

from and do not live in the communities they serve.

Angela, Brenda, and the two university researchers delivered the Cornrow Curves lesson together, reinforcing the math content across virtual and physical braiding activities. Students moved back and forth between learning to physically braid with Angela (see Figure 2) and program braids on their computers. Each day the classroom was busy with children engaged in culturally and computationally rich activities. Reflecting on the lesson, Brenda explained the importance of collaborating with Angela, "I liked when she interacted with the kids, and she had some of my more difficult young ladies come over and actually be interested in doing Cornrow Curves, where on another occasion they might sit and not participate." What can this vignette tell us about the role of local cultural experts like Angela in broadening the participation of underrepresented communities in computing?

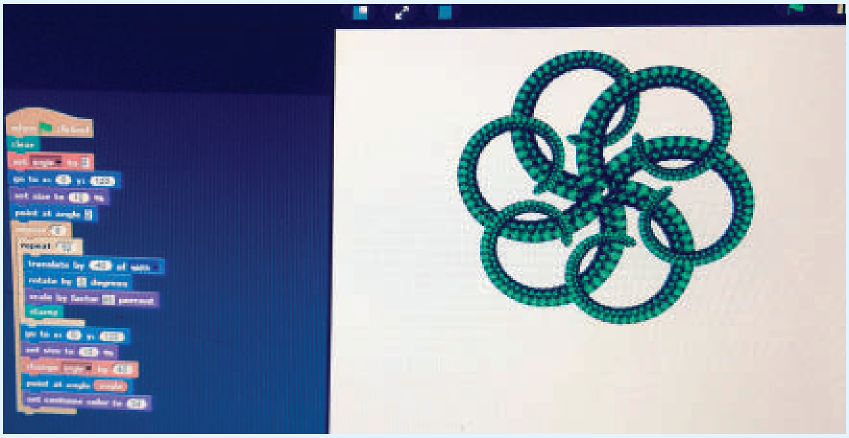
Culturally Responsive Computing in Formal Computer Science Education

While culturally responsive computing in out-of-school or after-school settings provides important insight into the strengths of cultural content for supporting CS education, there has been less attention to its role in formal classrooms. One possible reason for this is the fact that some aspects of culturally responsive computing cannot be easily implemented as a set of predetermined steps. As the name suggests, it aims to be responsive to locally situated contexts. Of course, culturally responsive computing can and should include pre-made curricula and tools. For example, the Exploring Computer Science curriculum (see http://exploringcs.org) includes culturally situated design tools and has rich opportunities for context-specific problem solving. However, if pre-packaging equates to standardization then there is a risk of shallowly representing computing-culture connections.

One way to make deep connections is to foster teachers' relationships with folks from outside the traditional school system who can provide insight into the larger context of students' lives, histories, and knowledge, as seen in the collaboration between Angela and Brenda. In another instance, Sandoval[4] described a CS teacher of European descent who began to develop culturally responsive competencies by working with self-identified Indigenous Xican@s, attending a food justice symposium, and helping students connect classroom content to real-world places, such as community gardens.

Therefore, focusing attention on culturally responsive computing in formal CS education provides opportunities to highlight the importance of out-of-school assets for broadening participation and, potentially, strengthening the relationship schools have with the communities they serve. Indeed, in the second author's CT4EDU (see http://ct4edu.

Figure 1. A student's Cornrow Curves design created during the lesson.



```
when clicked
clear
set angle to [illegible]
go to x: [illegible] y: [illegible]
set size to [illegible] %
point at angle [illegible]
repeat [illegible]
  repeat [illegible]
    translate by [illegible] of width
    rotate by [illegible] degrees
    scale by factor [illegible] percent
    stamp
  go to x: [illegible] y: [illegible]
  set size to [illegible] %
  change angle by [illegible]
  point at angle angle
  set costume color to [illegible]
```

org) project on integrating computing ideas in elementary classrooms, teachers discussed a need to be engaged as partners with various stakeholders if curriculum innovations are to be successful.[5] Specifically, teachers mentioned the need for increasing opportunities to interact with parents out-of-school so as to better understand students' cultures and identities in their classrooms. To build on this idea, we argue that CS teachers need to be supported to engage the communities they serve, developing culturally responsive computing competencies by collaborating with parents, cultural experts, entrepreneurs, technologists of color, and others in the creation of culturally and computationally rich formal CS education.

Figure 2. Angela uses transformational geometry terms to explain cornrow braiding.

Computing in Community

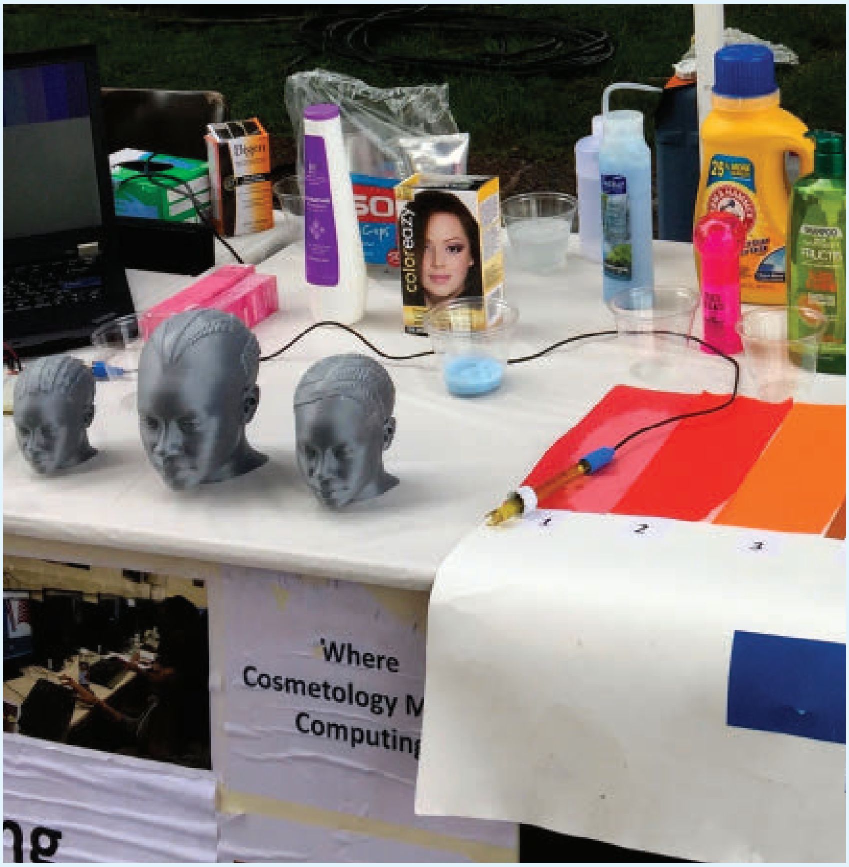
A collaboration like the one between Angela and Brenda provides insights into how culturally responsive computing competencies can be developed not only by in-service teachers but also their community partners. When asked if she learned anything new about computing from creating a design in Cornrow Curves, Angela explained, "Oh yeah. I feel like a scientist now … a scientist and a teacher now, I feel like I can just conquer the world, just kidding [laughs]. No, but I do, I feel like I am a lot more knowledgeable in, you know, computer programming, geometry, hair braiding." Angela's suggestion that she learned new information about her own area of expertise —braiding— may indicate that school-community collaborations can bring CS and cultural knowledge into mutually beneficial relationships. If this is the case, working with teachers as part of formal CS education may provide a way for cultural experts to take what they learn and incorporate it into relevant community locations (such as a hair salon). Formal culturally responsive computing education, then, would become a multidirectional strategy for broadening participation by increasing the chances that students will not only be exposed to CS in the classroom but also while going about their everyday lives.

For example, in the first author's work on the Cos-computing (cosmetology + computing) project,[3] 3D printed cornrow braids, inspired by high school students' Cornrow Curves designs, were displayed at an African American arts and culture festival (see Figure 3) and eventually placed in a cosmetology salon. As a result, one stylist at the salon, who had worked with the high school students and presented at the festival, explained how the 3D prints prompted conversations about CS concepts (for example, algorithms) with her customers, adding computing to the existing repertoire of technical and scientific knowledge (for example, pH, hair follicle anatomy, and so forth) that is part of everyday salon conversations. This creates a type of loop between formalized CS knowledge and the localized cultural knowledge that may be familiar to students. The idea is that culturally responsive computing collaborations can make CS education and local sites of cultural wealth mutually supportive, diffusing knowledge of computing-culture connections across both school and community locations.

Challenges

However, asking teachers who are not from the communities they serve to develop local relationships is a difficult task. Many teachers face school budget cuts, have little control over the types of professional development their schools provide, and are expected to standardize instruction. Therefore, putting additional requirements on teachers to create deep forms of culturally responsive computing alone is not practical. Instead, school districts, unions, and universities should facilitate school-community relationship building by paying local cultural experts to attend or co-design professional development programs and workshops alongside teachers and technologists from industry or the academy. In addition, these stakeholders can seek to leverage the expertise of teachers and

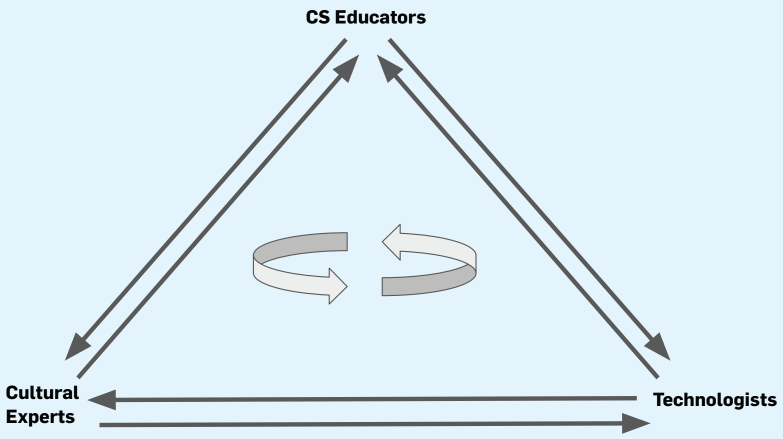
Figure 3. A "Cos-computing" booth, featuring 3D-printed Cornrow Curves designs alongside a pH sensor activity, at an African American arts and culture festival.



school staff who do live in the communities they serve.

Figure 4 represents the different ways that CS educators, cultural experts, and technologists might collaborate in their collective development of culturally responsive computing competencies. While they all bring individual knowledge, we think that the trading and intersecting of expertise at the different vectors will provide opportunities for deeper multi-directional culturally responsive computing engagements, connecting academic pursuits to cultural wealth (for example, hair salons) and technological wealth (for example, computer science departments). The loop in the middle reminds us that in-school and out-of-school contexts can be mutually supportive and reinforcing in broadening participation efforts.

Figure 4. A diagram to think about the depth of culturally responsive computing implementation across expertise.



Conclusion

Computing educators are in a good position to find innovative ways to support broadening the participation of African Americans, Native Americans, Latinxs, and other underrepresented groups in CS. But to develop these competencies requires teachers to connect with the communities where students live and work. This may mean CS educators will need to engage in life beyond the school walls (for example, attending public events, participating in community art projects, spending money at local businesses, and so forth), while also creating opportunities for cultural experts to shape CS curricula. With this in mind, culturally responsive computing aims to repurpose CS education by making it meaningful to not only students, but also to their families and communities. Increasing the buy-in that CS education has with local community members and representing it in culturally meaningful locations may increase the possibility that students will find CS to be a meaningful field where they want to participate and feel like they belong.

References

1. Eglash, R. *African Fractals: Modern Computing and Indigenous Design.* Rutgers University Press, New Brunswick, NJ, 1999.
2. Eglash, R., Gilbert, J.E., and Foster, E. Toward culturally responsive computing education. *Commun. ACM 56*, 7 (July 2013), 33–36.
3. Lachney, M., Babbitt, W., Bennett, A., and Eglash, R. Generative computing: African-American cosmetology as a link between computing education and community wealth. *Interactive Learning Environments* (2019), 1–21.
4. Sandoval, C.D.M. *Ancestral Knowledge Meets Computer Science Education: Environmental Change in Community.* Palgrave Macmillan, New York, 2019.
5. Yadav, A. and Wilson, J. CT4EDU: Building equitable access for CT in elementary classrooms. Poster presented at CSforAll Knowledge Forum, El Paso, TX, (Sept. 2018).

Michael Lachney (lachneym@msu.edu) is an Assistant Professor in the College of Education at Michigan State University, East Lansing, MI, USA.

Aman Yadav (ayadav@msu.edu) is a Professor in the College of Education and Director of the Masters of Arts in Educational Technology program at Michigan State University, East Lansing, MI, USA.

DOI:10.1145/3379920

Peter J. Denning and Dorothy E. Denning

The Profession of IT
Dilemmas of Artificial Intelligence

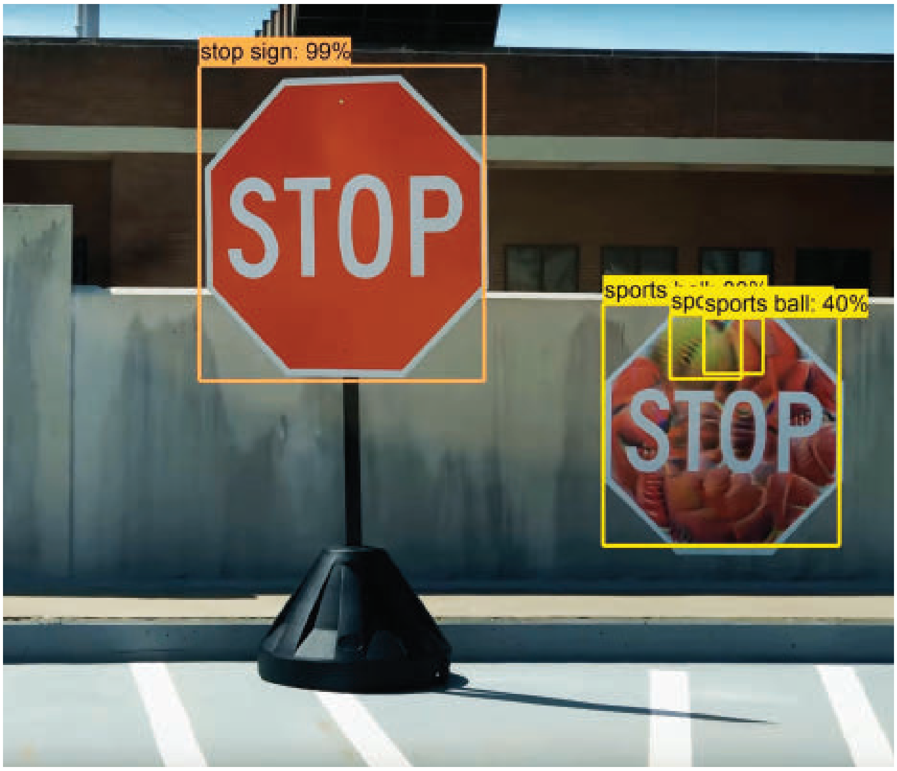
Artificial intelligence has confronted us with a raft of dilemmas that challenge us to decide what values are important in our designs.

MANY SPEAKERS HAVE pointed to various challenging ethical and design dilemmas raised by AI technology—we will describe 10 of the most prominent ones in this column. The first few are mostly technical; they arise from seemingly impenetrable complexity of the new technology. The final few ethical and design dilemmas include strong social dimensions; they arise from the difficulty of resolving emotional value conflicts to everyone's satisfaction.

Explainability

The most common AI technology is the artificial neural network (ANN). An ANN consists of many layers of artificial neurons interconnected via weighted links. ANNs are not programmed in the conventional way by specifying the steps of an algorithm. Instead they are trained by showing them large numbers of examples of input-output pairs and adjusting their internal connection weights so that every input gives a correct output. The matrix of connection weights can amount to several gigabytes of storage. In effect, an ANN encodes the training examples of a function in its connection matrix and extrapolates them to estimate the outputs for data not in the training examples.

What happens if the human operator wants to know why the network generated an unexpected or erroneous output? In a conventional program, the operator would locate the code segment responsible for the output and if necessary repair it. In a neural network, the operator sees no algorithmic steps, just an unintelligible gigabyte size matrix of weights. How the weights relate to the unexpected output is totally opaque. It is a hot research area to find ways to augment neural networks so that their outputs can be explained.



An example of the stop-sign fragility problem: Will a driverless car's road-sign recognizer correctly see a stop sign?

SCREEN CAPTURE OF THE STOP-FRAGILITY PROBLEM COURTESY OF POLO CLUB OF DATA SCIENCE, GEORGIA TECH/YOUTUBE. BASED ON PAPER ARXIV.ORG/ABS/1804.05810

Fragility

Neural networks can be quite sensitive to small changes in their inputs. For example, changing a few pixels of a trained input image can cause the output to change significantly even though the human operator cannot see a difference in the image. This leads to uncertainty in whether to trust a neural network when it is presented with new data on which it was not trained. For example, when shown a new photo of a person's face, will it identify it as that person or someone else? Will a road sign recognizer in a driverless car correctly see a stop sign, and stop?

The sensitivity to small input changes is a vulnerability. A new subfield, "adversarial AI," has sprung up to find defenses against an adversary seeking to cause a neural network to malfunction. In one famous experiment, a road-sign recognizer was confused by an image of a stop sign on which small squares of masking tape were applied at strategic locations; instead of saying "stop sign" the network said "speed limit sign." In the current state of the art, it appears small changes in sensor outputs that feed a neural network can produce significantly wrong outputs. What looks to a human like a small continuous change to the input looks to the network as a discontinuous jump to a new state.

Fragility can also be seen when comparing neural networks. Suppose two neural networks are each trained from a different training set taken as a sample from a larger population. By all standard measures the two training sets are fair representatives of the population. When this is tried in practice, the two networks can respond with different outputs when shown the same input. Statistically minor changes in the training data can result in major changes of the output.

Researchers are looking for improved methods to measure the sensitivity of neural networks to small changes in their inputs, and ways to ensure a small input change results only in a small output change.

Bias

This is an issue that arises with the training data of neural networks. A bias in the training data can skew outputs. Many people are concerned about police use of neural networks trained by faces of predominately white people that give wrong identifications of faces of people of color. The bias of the training data may be invisible to the people running the training algorithms and only becomes visible in the results when the network is presented with untrained inputs.

The bias issue is further complicated by the fact that human beings are inherently biased. Each person has an individual way of interpreting the world that does not always agree with others. What appears as bias to one person may appear as fairness to another. What one person sees as the solution to a bias problem may appear as a new bias to another. This aspect of bias cannot be resolved within the technology by new statistical methods. It demands that humans respect each other's differences and negotiate solutions for conflicts.

It is as hot research area to find ways to augment neural networks so that their outputs can be captured.

Fakes

Tools for editing images, videos, and soundtracks are being combined with AI tools to produce convincing fakes.[1] They cannot be distinguished from real images, videos, or soundtracks without advanced equipment and forensic skills. These digital objects often contain biometric data of specific individuals, used for identification. How can we trust digital identifications when digitized forms of traditional identifications cannot be distinguished from fakes?

High Cost of Reliable Training Data

Neural networks require large training sets. Getting properly labeled data is time consuming and expensive. Consider the labor costs of a training scenario. Trained physicians must review colon images to identify suspicious polyps and label the images with their diagnoses. Suppose training a suspicious-polyp recognizer needs a million labeled images and a physician can diagnose and label an image in six minutes. Then 100,000 physician hours are needed to complete the labeling. If physicians were paid $50 an hour for this job, the training set would cost $50 million.

Training is also energy-intensive: a training that takes several days is as computationally intensive as bitcoin mining.

This means good quality training sets are hard to come by.

To keep the costs down there is a lot of interest in open source training sets. Users of these training sets are right to be concerned over the quality of the data because the persons contributing might be low-wage amateurs rather than well-paid professionals. There are reports of exactly this happening in open datasets that are then used to train medical diagnosis networks.

So even if developers are determined to avoid bias by getting large datasets, they will be expensive and right now it is difficult to determine their quality.

The big tech companies have a lot of reliable raw data about their users but are not sharing.

Military Uses of AI

Project Maven is a U.S. Pentagon project to use AI to give drones the power to distinguish between people and objects. Google was a partner and outsourced image differentiation to a company that used captchas to distinguish people from other objects. The gig workers looking at the captchas did not know they were teaching an AI system for a military purpose. When 3,000 Google employees formally protested, saying Google should not be developing technologies of war, Google executives decided not to renew the Maven contract.

Aversion to research for the military has been a difficult issue in universities since the days of the U.S. Vietnam war. Most universities divested themselves of laboratories that researched such technologies. Most DOD contracts are with private companies that are not involved with universities. With the large influx of new graduates into the big tech companies, the same aversion is now showing up among employees of private companies. The dilemma is in how to balance the need for national defense with the desire of many employees to avoid contributing to war.

Weapons and Control

The military's interest in AI to distinguish potential targets for drone attacks introduces another dilemma: Should a drone be allowed to deploy its weapon without an explicit command from a human operator? If AI is used in any weapons system, should a human have the final say in whether a weapon is launched?

Looking to the future, AI may also facilitate the creation of inexpensive weapons of mass destruction. Stuart Russell, a computer science professor at UC Berkeley and an AI pioneer issued a dire warning about AI controlled drones being used as WMD.[2] He produced a video, "Slaughterbots," which presented a near-future scenario where swarms of cheap drones with on-board facial recognition and a deadly payload assassinate political opponents and perform other atrocities. A swarm of 25,000 drones could be as destructive as a small nuclear bomb at a tiny fraction of the price.

Russell worries not only about the destructive potential of current AI technology, but about even more destructive potential of advanced AI. He says the creation of a super-intelligent computer would be the most significant event in human history—and might well be its last.

Issac Asimov postulated the famous Three Laws of Robotics in 1950 but no one has found a way to enforce them in the design of robots. The dilemma is: Should we continue to work on developing general AI when we do not know if we can control it?

Employment and Jobs

There is widespread fear that AI-powered machines will automate many familiar office tasks and displace many jobs. This fear is not unique to AI technology. For hundreds of years, new technologies have stirred social unrest when workers felt threatened by loss of their jobs and livelihoods. The fear is heightened in the modern age by the accelerated pace of AI automation. A century ago, a technology change was a slow process that took a generation to be fully adopted. Today, a technology change can appear as an avalanche, sweeping away jobs, identities, and professions in just a few years. Although the historical record says the new technology is likely to produce more jobs in the long run than it displaces, the new jobs require new skill sets the displaced workers do not have. The appearance of new jobs does not help the displaced.

> **Should we work on developing general AI when we do not know if we can control it?**

One solution to this is regional training centers that help displaced workers move into the new professions. Unfortunately, the investment in such centers is currently limited.

Another proposed solution is the Universal Base Income (UBI), which would give every adult a monthly stipend to make up for income lost to automation. This proposal is very controversial.

Surveillance Capitalism

Surveillance capitalism is a term coined by Shoshana Zubhoff to describe a new phenomenon arising in the commercial space of the Internet.[3] The issue is that most online services capture voluminous data about user actions, which the service provider then sells to advertisers. The advertisers then use AI to target ads and tempt individuals into purchases they find difficult to resist. They also use AI to selectively customize information to individuals to manipulate their behavior such as their thinking about political candidates or causes.

The phenomenon is spreading to app developers as well. Their apps are Internet connected and provide data from mobile device sensors. A growing number are opting for "X as a service," meaning function X is no longer provided as installable software, but is instead a subscription service. In addition to a steady stream of monetizable personal data, this strategy provides a steady stream of income from subscribers.

Many of these services and apps are so attractive and convenient the tide to adopt them will not soon reverse. The dilemma for app developers is to find a way that provides the service without compromising individual user control over their data. The dilemma for citizens is how to effectively resist the trend to monetize their personal data and manipulate their behavior.

Decision Making

Dilemmas arise around machines that make decisions in lieu of humans. Consider the self-driving car when the sensors indicate "pedestrian ahead." How does the car decide between applying the brakes abruptly and potentially harming the occupant, or applying the brakes moderately and potentially hitting the pedestrian? Or, should the car swerve into the car alongside or drive off a cliff? Or do we hand control to the human and let that person choose an alternative? More generally, do we want machines to only make recommendations or machines that make and act on decisions autonomously? Is it even possible for machines to "act ethically"? Or is that something only humans can do?

Conclusion

None of these dilemmas is easily resolved. Many can be couched as ethical dilemmas that no professional code of ethics has been able to answer. Some of these dilemmas make obeying Asimov's First Law impossible: no matter what action is taken (or not taken), a human will get hurt. Software developers face major challenges in finding designs that resolve them.

References

1. Farid, H. *Fake Photos.* MIT Press Essential Knowledge Series, 2019.
2. Russell, S.R. *Human Compatible: AI and the Problem of Control.* Allen Lane of Penguin Books, Random House, UK, 2019.
3. Zuboff, S. *The Age of Surveillance Capitalism.* Public Affairs Books, 2019.

For more on autonomous vehicles, see Awad et al. on p. 48.

Peter J. Denning (pjd@nps.edu) is Distinguished Professor of Computer Science and Director of the Cebrowski Institute for information innovation at he Naval Postgraduate School in Monterey, CA, USA, is Editor of ACM *Ubiquity*, and is a past president of ACM. The author's views expressed here are not necessarily those of his employer or the U.S. federal government.

Dorothy E. Denning (dedennin@nps.edu) is Distinguished Professor Emeritus of Defense Analysis at the Naval Postgraduate School in Monterey, CA, USA.

DOI:10.1145/3378543

Omer Reingold

Viewpoint
Through the Lens of a Passionate Theoretician

Considering the far-reaching and fundamental implications of computing beyond digital computers.

WHEN I WAS taking my very first CS class—almost three decades ago—the lecturer recommended David Harel's book *Algorithmics: The Spirit of Computing*:[1] "If you want your friends and family to know what you are doing here" he told us, "let them read this book." A wonderful piece of advice, still applicable today, which enabled me to share my budding love for CS with others, but also helped me further my own understanding of "what we are doing here."

Many years later, our understanding of the Theory of Computation (ToC) has dramatically grown and the field is thriving. I find myself yet again with the opportunity to share my love of the riveting notion of computation. Since Avi Wigderson's *Mathematics and Computation: A Theory Revolutionizing Technology and Science*,[2] was published recently, I have been recommending it to anyone who showed an interest in ToC. But I also recommend it to my fellow theoreticians, who study ToC, because *Mathematics and Computation* is a way for us too to better understand what it *really* is that "we are doing here."

My personal-perspective Viewpoint on *Mathematics and Computation*, should not be read as a comprehensive summary of the book, but rather as an invitation to read and investigate it for yourself. Throughout the chapters (and especially in the self-contained Chapters 13 and 20), Wigderson lays out a body of evidence demonstrating the intellectual reach of ToC. Despite its technical breadth, the book takes a conceptual perspective and aims to reach a wide audience. While not a popular science book, there is something in the book for many audiences: advanced students, researchers in variety of area,[a] educators, and motivated non-academics.

Mathematics and Computation focuses on an important branch of ToC known as Complexity Theory. Wigderson leads the reader on a fascinating expedition through notions, results (many of which were discovered after Harel's book[1] appeared) and fundamental open problems. The book discusses sub-areas like randomized computations and derandomization, cryptography, learning theory, quantum computing as well as basic topics like the P vs. NP problem, the PCP Theorem, and Zero-Knowledge proofs. Wigderson tells the story of computation and leads the reader on a fascinating expedition through these topics and many more. But he also tells us the story of the rather small research community that studies ToC and its tremendous impact.

I will devote the remainder of this Viewpoint to the book's epilogue (Chapter 20), which may very well be its most widely appealing part. The

[a] As the name suggests, a focus of *Mathematics and Computation*, is the deep interactions between ToC and mathematics. In particular, Chapter 13 explores the impact of the computational lens on mathematics.

IMAGE BY PORING STUDIO

How can computation be so far-reaching and fundamental?

epilogue takes a much more complete view of ToC and thoroughly examines the power of the so-called computational lens: Increasingly, natural and social phenomenon are viewed by scientists as *being, or performing, computations*. It is, therefore, natural for computer scientists to study computations, whether they take place in our smartphones or in our cells, in the structure of social networks as well as in the evolution of species. Indeed, in recent times questions traditionally viewed as part of biology, physics, economics, sociology, arts and more, have been approached through this "computational lens."

How can computation be so far-reaching and fundamental? To understand, we first need to see computation in its full generality as "the evolution process of some environment, by a sequence of *simple, local steps*."[2] This takes us far beyond digital computers, to all the places where computation can happen.

For starters, it would not be controversial to argue that the human mind computes (after all, identifying the letters, words, and meanings you are reading just now requires impressive computing power). But humans are not the only animals that compute, in the words of Cole Porter, "birds do it, bees do it, even educated fleas do it." Indeed, small or large tasks of small or large animals require evolved computations. If individual animals compute, the behavior of communities of animals, composed of countless parallel actions taken by individuals in these communities, can also be naturally viewed as elaborate algorithms. The colony of ants searching an environment for food by laying down and picking up pheromone trails perform an effective algorithm for finding short paths to available food (interestingly, these algorithms inspired optimization algorithms that are executed on digital computers). Another example is the marvelous maneuver of birds flocking and fish schooling, which is made up of individuals following their programming to give space but also align with others. But if communities of other animals compute, why not communities of humans? The behavior of individuals in economic markets (computing prices), or in social networks (computing influence), are very complex but still composed of many local (and comparatively simple) steps. Not only can we observe the social behavior of communities and species through the computational lens, but we can also view the process that created these species in the first place as algorithmic in nature. The evolution of species from single cells to the splendid organisms in existence follows local, incremental steps where genetic material transforms and survives as a probabilistic function of its fitness to the environment. While, so far, we zoomed out to examine increasingly larger instances of computations, we might as well zoom in and consider computations in tiny spaces: cellular processes, such as the folding of proteins, and intercellular interactions are some of the first phenomena to be studied algorithmically with basic operations being the chemical reactions between the molecules that comprise the cell. Finally, if the cellular or molecular level is not small enough, we can consider the atomic level, where the interactions of subatomic particles is the basis of the disruptive field of quantum computation.

The real question of course is not how many phenomena can be viewed as computations, but rather *what can be gained* from such a perspective. The secret here lies in the *efficiency* and the *complexity* of computation: When we study computation in the world around us, it is imperative that we understand not only what is *possible* but also what is *feasible* given existing resources. The role of the computer scientist is to investigate the rules that separate what is efficient from what is too complex; this way, computational insights offer a unique perspective on old questions. A model of evolution should justify not only that such evolution is possible in principle, but also that the mechanism—the algorithm—governing evolution is likely to have produced such an organism within the resources that were at the disposal of evolution (the number of generations, the number of organisms, the number of major environmental changes). If we want to predict the behavior of a market, it is not enough to prove an equilibrium exists but also that such an equilibrium could be efficiently computed by the market. Similar considerations refine the study of any natural and social process that could be understood as a computation and makes the tools of ToC particularly powerful. Assuming the various participants in a given interaction have limited resources (for example, time or space) revolutionizes the way we understand basic notions such as knowledge, randomness, entropy, learning, secrecy, fairness, and many more.

Studying computation is not new. After all, both algebra and geometry are algorithmic since birth (it is not a coincidence that both the names algebra and algorithms originate in the name and work of the great Persian mathematician and scientist Muhammad ibn Musa al-Khwarizmi). But the birth of ToC as a modern field of study can be pinpointed to the seminal work of Turing in 1936. Turing gave a definition of computation (anticipating the invention of digital computers by a decade) which is both simple and powerful, through what is now called a Turing Machine. Since then ToC has grown in sophistication and depth, but it preserves much of the magic and the values of Turing's work. The contemporary picture portrayed by Wigderson's book is that of a deep and insightful core of ToC, surrounded by application areas within ToC (learning, algorithmic game theory, verification, pseudorandomness, property testing, distributed computing, communication complexity, quantum computing, cryptography, and more), which interact with diverse fields including computer science, mathematics, statistics, social science, biology, physics, economics.

The journey to which Wigderson invites us goes very far and very deep: it immerses the reader into the magnificent world surrounding the notion of computation. Along with the wealth of accessible knowledge and understanding offered by this book, there is passion pouring from every page, a passion that is inspiring and difficult to resist: an admiring contemplation of the scientific riddles surrounding computation, together with a convincing, book-length argument that they are as essential to unraveling the mysteries of the universe as any other pursuit of knowledge. I therefore feel it is most appropriate to conclude by letting *Mathematics and Computation* speak for itself, in a passage that beautifully captures the heart of the story it tells.

"The theory of computation, since its inception by Turing in 1936, is as revolutionary, fundamental, and beautiful as the great theories of mathematics, physics, biology, economics ... that are regularly hailed as such. Its impact has been similarly staggering. The mysteries still baffling ToC are as challenging as those left open in other fields. Moreover, the ubiquity of computation makes its theory central to all other disciplines. In creating the theoretical foundations of computing systems, ToC has already played, and continues to play, a major part in one of the greatest scientific and technological revolutions in human history. But the intrinsic study of computation transcends human-made artifacts and underlies natural and artificial processes of all types. Its expanding connections and interactions with all sciences, integrating computational modeling, algorithms, and complexity into theories of nature and society, is at the heart of a new scientific revolution!"[2]

References

1. Harel, D. *Algorithmics: The Spirit of Computing.* Addison-Wesley, Reading, MA, 1st edition, 1987; 2nd edition, 1992.
2. Wigderson, A. *Mathematics and Computation: A Theory Revolutionizing Technology and Science.* Princeton University Press, Princeton, NJ, 2019.

Omer Reingold (omer.reingold@gmail.com) is a faculty member of the Computer Science Department at Stanford University, Stanford University, Stanford, CA, USA. He received the 2005 ACM Grace Murray Hopper Award for his work in finding a deterministic logarithmic-space algorithm for ST-connectivity in undirected graphs.

Calendar of Events

Mar. 3–4
HotMobile '20: The 21st International Workshop on Mobil Computing Systems and Applications,
Austin, TX, USA,
Sponsored: ACM/SIG,
Contact: Padmanabhan Pillai,
Email: pillai@umich.edu

Mar. 3
SOSR '20: Symposium on SDN Research,
San Jose, CA, USA
Sponsored: ACM/SIG,
Contact: Anduo Wang,
Email: Wadaries@gmail.com

Mar. 9–13
Date '20: Design, Automation and Test in Europe,
Grenoble, France,
Contact: Giorgio Di Natale,
Email: giorgio.di-natale@univ-grenoble-alpes.fr

Mar. 11–14
SIGCSE '20: The 51st ACM Technical Symposium on Computer Science Education,
Portland, OR, USA,
Sponsored: ACM/SIG,
Contact: Mark Sherriff,
Email: sherriff@virginia.edu

Mar. 14–18
CHIIR '20: Conference on Human Information Interaction and Retrieval,
Vancouver, BC, Canada,
Sponsored: ACM/SIG,
Contact: Heather O'Brien,
Email: h.obrien@ubc.ca

Mar. 16–20
ASPLOS '20: Architectural Support for Programming Languages and Operating Systems,
Lausanne, Switzerland,
Sponsored: ACM/SIG,
Contact: James Larus,
Email: larus@larusstone.org

Mar. 16–18
CODASPY '20: 10th ACM Conference on Data and Application Security and Privacy,
New Orleans, LA, USA,
Sponsored: ACM/SIG,
Contact: Vassil Roussev,
Email: acm@roussev.net

DOI:10.1145/3341722

Kieron O'Hara and Wendy Hall

Viewpoint
Four Internets

Considering the merits of several models and approaches to Internet governance.

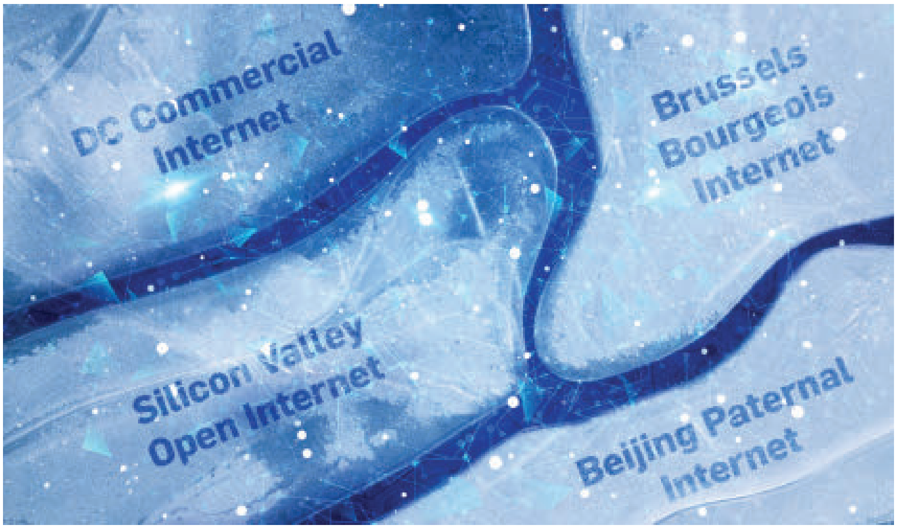

DC Commercial Internet
Brussels Bourgeois Internet
Silicon Valley Open Internet
Beijing Paternal Internet

IMAGE BY ANDRIJ BORYS ASSOCIATES, USING SHUTTERSTOCK

The vision of an open Internet is characteristic of Silicon Valley's tech pioneers. The free and efficient flow of packets of bits requires decentralization to prevent bottlenecks occurring at the central points as the system scales, open standards to allow interoperability, and IP addresses to identify the correct destination. We take this system for granted, but one does not need a very long memory to recall a time when IT was dominated by proprietary protocols like AppleTalk or DECnet, and when one could not easily send an email message from AOL to Prodigy. Yet the Internet has not simply improved—it has evolved into an open system as a result of philosophical and political decisions, as well as technical ones.[2,5]

In a recent *Communications* "Cerf's Up" column, Vinton Cerf argued there is a fundamental division between the IP layer and the application layers of the Internet, which together function to keep the open Internet flowing, and what he called the "virtual political layer," higher in the stack where the content is consumed and judged. At the lower levels, protocols such as TCP, SMTP, and HTTP ignore content, using only metadata such as payload types, timestamps, and email formats. Cerf worries that constraints imposed on information at the upper levels will have effects further down the stack.[3]

We concur with Cerf's assessment, but we must beware of concluding that values are only relevant to the upper levels where we worry about the social effects of processing information, while down below the Internet's plumbing just gets collections of bits to the right place in the right order as efficiently as possible.[5] Aiming for seamless interoperability, for example, is certainly important, but that should not be equated with being value neutral.[12]

For those of a libertarian cast of mind, politics and engineering complement each other to create the **Silicon Valley Open Internet**. The free flow of information through the network supports and is supported by free speech and unrestricted association.[1] However, if liberty is unrestricted, individually rational behavior may damage public goods. Efficient transfer of information is wonderful, unless the information is hate speech or a virus or sensitive personal data; it is already value-laden to suggest that we can meaningfully evaluate the efficiency of information flow independently of its content.

Different nations and organizations regulate and constrain where they can. In our recent paper for the Centre for International Governance Innovation, *Four Internets: The Geopolitics of Internet Governance*, we argue that some key geopolitical actors are projecting models of Internet governance, and consequently creating their own realities—alternative Internets to Silicon Valley's.[10]

This does not mean the Internet is (necessarily) fragmenting; we agree with Milton Mueller that 'fragment' is "the wrong word with which to approach this problem."[8] However, we are not as sanguine as he that the network effects and economic benefits of a seamlessly connected Internet "will continue to defeat … systematic deterioration of the global technical compatibility that the public Internet created."

We also dissent from Mueller's narrow focus on sovereignty; the actors we describe push back against the logic of the Open Internet with ideologically informed aspirations intended to provide models for the Internet as a whole, projecting ideals and foreign policy, not merely defending national sovereignty. Furthermore, on the multistakeholder governance model of the Internet, governments are not the only actors of importance;[5] others include engineers and hackers, civil society, lawyers, business, and private individuals with political agendas.

The different models we describe in the Viewpoint all recognize the advantage of connection to the network. They can—indeed, do—co-exist in uneasy armistice, relying on those lower protocol levels to keep them connected, like a dysfunctional family sharing the family home. But they compete for influence to shape the Internet's development, often at the relatively high level of institutions and regulation, but also at the lower levels. As a specific example of how technical issues influence and are influenced by the higher levels, the decision not to make IPv6 backward compatible with IPv4 has opened up new avenues of development and freedom for innovation that have removed constraints to many alternative approaches to Internet governance.

What are these alternative Internets?

The birth of the Internet within the U.S. military-industrial complex brought libertarians together in coalition with more hard-headed types. But this coalition is coming apart, and we are seeing a distinct and also largely American vision emerge in tension with Silicon Valley's Open Internet, which we call the **DC Commercial Internet**. If we think of data and Internet resources as property, then on this view the walled gardens of the tech giants are legitimate creations of their owners, to exploit commercially as they think fit. Users find these gardens easy, useful, and attractive. There is an oligopoly of giant companies, but, as Schumpeter argued, near-monopolies should be tolerated if they produce innovation—which the tech behemoths certainly have.

The distinction is most clearly seen in the interminable arguments over Net neutrality.[9] The First Amendment prevents the government from abridging free speech—but does that mean the government must therefore use its powers to promote free speech by preventing interference with the free flow of information by private actors? The Silicon Valley answer is 'yes', and Net neutrality follows. The response of the Supreme Court (which has remained consistent for some decades, as presidents have come and gone—hence our location of this ideology in Washington, DC), is 'no', and that, if a service provider wishes to censor the speech (that is, slow down the packets) of its users, the government cannot prevent it without abridging the *provider's* free speech. On the DC Commercial Internet vision, Net neutrality should be determined by the contract between provider and user.

What are these alternate Internets?

Not everyone wants market solutions, however. A third vision imagines a more or less open Internet, on which good behavior is the norm. Trolling, privacy invasion and fake news should be marginalized or regulated away by a strong civil society whose members are trustworthy and trusting. This vision is particularly popular in the EU, as a means of protecting "fundamental European values and principle."[6] The well-ordered, self-regulating, responsible **Brussels Bourgeois Internet**, long an ideal, has been given teeth by the Court of Justice of the European Union, in a series of aggressive interpretations of data protection and competition law. GDPR is perhaps its most powerful weapon, and some European data protection regulators are using it to project European values (and regulations) internationally, with some success—for instance, the Brazilian data protection law, the GDPL, is pretty similar,[a] while Tim Cook and Mark Zuckerberg have each canvassed the possibility of harmonizing global laws around GDPR. Its influence reaches down the protocol stack—for instance, its championing encryption as best data protection practice will incentivize encryption in the application layer, where so many security breaches occur.

GDPR is not the only influence, though. A controversial new EU Directive on Copyright for the Digital Single Market[b] is expected to impact popular content sites such as YouTube or Twitter, while the UK has recently released a white paper intended to regulate harmful content on global tech platforms.[c]

Regulation is not the only response to openness and markets. A stronger view is that the Internet—the medium for so much human interaction—could be the means of creating social harmony, not disruption, by ensuring it allows 'good' things to happen, and 'bad' things to be prevented. The Internet, in other words, can be used for social control, by authorities who define and judge 'social good'. This kind of paternalism comes in many tempting flavors, from the mild 'nudge' philosophy through to outright authoritarianism (active intolerance of dissent). It can be disastrous; a Ugandan tax on Internet connections, intended to discourage gossip, recently resulted in a massive decline in Internet subscriptions.[d] The leading light in this area is China, which has placed digital technologies at the heart of propaganda, public opinion, and social control, and so we dub this fourth vision the **Beijing Paternal Internet**, although all governments find it attractive to some extent.

China's ambitions with respect to the Internet were made clear in a series of articles in this magazine in 2018.[15] Its own tech giants, Baidu, Alibaba, and Tencent, have commercial freedom to develop innovative services, but work closely with government on a tacit national project both to create a cyber superpower and to manage data, search, commerce, and other types of Internet access. A national data-driven 'social credit' system may well grow out of a series of pilots to use crowdsourced data to score the trustworthiness of citizens, penalize those who have failed to pay debts or fines, and reward those who make social contributions, such as by donating blood.[4]

Other ideals exist, but they lack powerful geopolitical backing. One fi-

[a] See http://bit.ly/2ReDxzL

[b] See http://bit.ly/370QmER

[c] See http://bit.ly/2NnUWF6

[d] See http://bit.ly/30iTidg

nal model deserves a mention: the hacking ethic, the use of the Internet against itself, despite itself, to create a world in which truth is in the eye of the beholder and in which anyone's motives can be made to appear impure. This outlaw view has long been pursued by individuals (as U.S. President Trump memorably suggested, by "somebody sitting on their bed that weighs 400 pounds"), but it has been weaponized by some nations impatient of the international order and the rule of law, most notably Russia, whose President Vladimir Putin has long espoused the paranoid nihilism of the mystical nationalist philosopher Ivan Ilyin (1883–1954).[13,e] This model, the **Moscow Spoiler**, is not a fifth vision for the Internet, because it does not push for a new Internet; it asks only an Internet upon which to be parasitic. It will not even trust that; Russia is reported to have successfully tested its cyberdefense capability by temporarily disconnecting itself entirely,[f] seeing security in separation.[11]

We have associated these various models with geopolitical actors that proselytize them, or have given them their most distinctive twists. However, the actual policies of any government cannot be reduced to a single ideological viewpoint. To take one example, Russia's foreign policy uses the Internet aggressively, but it also wants to promote business and social stability, which require different ideas, while its policing of the opposition has spawned an impressive surveillance capability.[12] Conversely, many nations indulge in misinformation and hacking, not just the Russians—the CIA are hardly amateurs in the game. So the Moscow spoiler model is neither equivalent to Russian Internet policy, nor unique to Russia. We say only that some arms of the Russian government, together with nationalist actors in a shady private sector, have refined the spoiler model to the *ne plus ultra* of disinformation, and so they get the credit reflected in the name. The four positive visions can be combined creatively. For instance, Tim Berners-Lee's Solid project to re-decentralize the Web uses Silicon Valley openness as a means to "restore balance—by giving every one of us complete control over data, personal or not, in a revolutionary way," but the end is recognizably Brussels bourgeois, to stop the Web being "an engine of inequity and division."[g] Sadly, they more often vie with each other for supremacy. This matters. For instance, the future of AI will be to a large extent determined by the regulation of data. China may be well-placed in the future to centralize data as its people are enthusiastic users of e-commerce and social media within an authoritarian context.[7]

Each of the visions has its merits.

Furthermore, approximately 50% of the world has yet to be connected to the Internet. The potential for growth is in Africa, India, and China itself. Which visions make themselves attractive to countries coming online will influence how the Internet will develop over the next decade. India's electronic ID system Aadhaar, for instance, is an incredible effort to give usable identities to the currently unvoiced, but what an instrument of potential social control is also being created. At least 20 governments[h] are interested in an Aadhaar of their own, with the World Bank helping export it.

Each of the visions, unlike the Moscow spoiler model, has its merits. Openness is key to the efficient and effective flow of information. The DC vision has produced incredible innovation, genuinely valuable and free services, and networks of undreamt-of complexity and density. Meanwhile, both the Beijing and Brussels visions emphasize defending public goods against disruption.

It is not possible to force agreement between differing geopolitical forces and ideological positions. However, in a world where international relations are increasingly seen as a zero-sum game, we need to focus on the mutual advantages of Internet unity, even if it is divided into a series of *de facto* satrapies governed on different principles. This means we need to work out methods and principles for Internet governance that simultaneously accept the range of views about its role in society, preserve the open standards that have made it such a revolutionary and successful technology, and ensure human dignity and privacy are respected. This is not a trivial task, and the future of the Internet may depend in particular on how data about individuals and groups is treated, and whether the current level of exploitation of data can be maintained without diminishing trust in the technology that provides it.

References

1. Benkler, Y. *The Wealth of Networks: How Social Production Transforms Markets and Freedom.* Yale University Press, New Haven, CT, 2006.
2. Cerf, V.G. Ownership vs. stewardship, *Commun. ACM 62*, 3 (Mar. 2019), 6; https://doi.org/10.1145/3310251.
3. Cerf, V.G. The upper layers of the Internet. *Commun. ACM 61*, 11 (Nov. 2018), 5, https://doi.org/10.1145/3281164.
4. Creemers, R. Cyber China: Upgrading propaganda, public opinion work and social management for the twenty-first century, *Journal of Contemporary China 26*, 103, 85–100, https://doi.org/10.1080/10670564.2016.1206281.
5. DeNardis, L. *The Global War for Internet Governance.* Yale University Press, New Haven, CT, 2014.
6. EDPS Ethics Advisory Group. *Towards a Digital Ethics*, European Data Protection Supervisor, 2018, https://edps.europa.eu/sites/edp/files/publication/18-01-25_eag_report_en.pdf
7. Lee, K.-F. *AI Superpowers: China, Silicon Valley and the New World Order.* Houghton Mifflin Harcourt, New York, 2018.
8. Mueller, M. *Will the Internet Fragment*? Polity Press, Cambridge, MA, 2017.
9. Nunziato, D.C. *Virtual Freedom: Net Neutrality and Free Speech in the Internet Age.* Stanford University Press, Stanford, CA, 2009.
10. O'Hara, K. and Hall, W. *Four Internets: The Geopolitics of Internet Governance*, Centre for International Governance Innovation paper no.206, 2018; https://www.cigionline.org/publications/four-internets-geopolitics-digital-governance.
11. Ristolainen, M. Should 'RuNet 2020' be taken seriously? Contradictory views about cyber security between Russia and the West. *Journal of Information Warfare 16*, (2017), 113–131.
12. Shilton, K. Engaging values despite neutrality: Challenges and approaches to values reflection during the design of Internet infrastructure. *Science, Technology, and Human Values*, 43, 2 (2018), 247–269; https://doi.org/10.1177%2F0162243917714869.
13. Snyder, T. *The Road to Unfreedom.* Bodley Head, London, 2018.
14. Soldatov, A. and Borogan, I. *The Red Web: The Kremlin's War on the Internet.* Public Affairs, New York, 2015.
15. Zaagman, E. China's computing ambitions. *Commun. ACM 61*, 11 (Nov. 2018), 40–41, https://doi.org/10.1145/3239534.

Kieron O'Hara (kmoh@soton.ac.uk) is an associate professor in the department of Electronics and Computer Science at the University of Southampton, Southampton, England, U.K.

Wendy Hall (wh@ecs.soton.ac.uk) is Regius Professor of Computer Science at the University of Southampton, Southampton, England, U.K.

[e] See https://nyti.ms/2NmATqI
[f] See https://bbc.in/2tKPFkD
[g] See http://bit.ly/2Tl5Qzu
[h] See http://bit.ly/35RaTtX

DOI:10.1145/3342102

Marc Canellas and Rachel Haga

Viewpoint
Unsafe At Any Level

The U.S. NHTSA's levels of automation are a liability for automated vehicles.

IMAGE BY FLYSTOCK/SHUTTERSTOCK.COM

WALTER HUANG, A 38-year-old Apple Inc. engineer, died on March 23, 2018, after his Tesla Model X crashed into a highway barrier in Mountain View, CA.[a] Tesla disavowed responsibility for the accident. "The fundamental premise of both moral and legal liability is a broken promise, and there was none here: [Mr. Huang] was well aware that the Autopilot was not perfect [and the] only way for this accident to have occurred is if Mr. Huang was not paying attention to the road, despite the car providing multiple warnings to do so."[b]

This is the standard response from Tesla and Uber, the manufacturers of the automated vehicles involved in the six fatal accidents to date: the automated vehicle is not perfect, the driver knew it was not perfect, and if only the driver had been paying attention and heeded the vehicle's warnings, the accident would never have occurred.[c] However, as researchers focused on human-automation interaction in aviation and military operations, we cannot help but wonder if there really are no broken promises and no legal liabilities.

These automated vehicle accidents are predicted by the science of human-automation interaction and the major aviation accidents caused, in large part, by naïve implementation of automation in the cockpit and airspace. Aviation has historically been plagued by designers ignoring defects until they have caused fatal accidents. We even have a term for this attitude: tombstone design. Acknowledging tragedies and the need to better understand their causes led aviation to become the canonical domain for understanding human-automation interaction in complex, safety-critical operations. Today, aviation is an incredibly safe mode of transportation, but we are constantly reminded of why we must

[a] See https://bloom.bg/2QSXGMY

[b] See http://bit.ly/382qmsH

[c] After fatal accidents in China and Florida in 2016, Tesla responded that "every time the Autopilot is engaged, the car reminds the driver to 'Always keep your hands on the wheel. Be prepared to take over at any time'" (http://bit.ly/2QWavpX). After a fatal accident in Arizona in March, 2018, Uber responded by installing new driver monitoring systems for detecting "inattentive behavior" (http://bit.ly/2RhoVzS). After the fourth fatal Tesla accident in Delray Beach, Florida, in 2019, Tesla responded that "when used properly by an attentive driver who is prepared to take control at all times, drivers supported by Autopilot are safer than those operating without assistance" (http://bit.ly/2uG3xMT).

respect the realities of human-automation interaction. A recent tragic example is Boeing 737 MAX 8's MCAS automation that contributed to two crashes and the deaths of 346 people before the human-automation aspect interaction failure was publicly acknowledged.

Science, like human-automation interaction, has a critical role in determining legal liability, and courts appropriately rely on scientists and engineers to determine whether an accident, or harm, was foreseeable. Specifically, a designer could be found liable if, at the time of the accident, scientists knew there was a systematic relationship between the accident and the designer's untaken precaution.[8]

The scientific evidence is undeniable. There is a systematic relationship between the design of automated vehicles and the types of accidents that are occurring now and will inevitably continue to occur in the future. These accidents were not unforeseeable and the drivers were not exclusively to blame. In fact, the vehicle designs and fatalities are both symptoms of a larger failed system: the five levels of automation (LOA) for automated vehicles.

The LOA framework is defined in the SAE International J3016 Standard (SAE J3016)[10] and adopted as the U.S. National Highway Transportation Safety Administration's (NHTSA) standard automated vehicle categories.[1] The LOA framework is premised on the idea that automation is collaborating at various levels of interaction as part of a team with a human operator. The typical LOA is a one-dimensional spectrum of interaction ranging from fully manual to fully automated, exemplified by NHTSA's Level 0 and Level 5. For their part, SAE states that their LOA "provides a logical taxonomy for [classification] ... in order to facilitate clear communications" and caveats that their LOA "is not a specification and imposes no requirements."[10]

The central flaw of LOA is right there in its name. Levels of automation focus on a singular, static definition of the automation's capabilities, ignoring the deeper ideas of teamwork, collaboration, and interdependency necessary for mission success—in this case operating a vehicle. Just reading the names of NHTSA's levels, you can see that the focus is solely on what the automation can do: 0, No Driving Automation; 1, Driver Assistance; 2, Partial Driving Automation; 3, Conditional Driving Automation; 4, High Driving Automation; 5, Full Driving Automation.

There is a systematic relationship between the design of automated vehicles and the types of accidents that are occurring now and will inevitably occur in the future.

This automation-centric perspective is counter to the idea of teamwork and explains why, despite their former prevalence in the academic literature, LOA is now acknowledged to be limited, problematic, and, to some, worth discarding altogether.[4,6] Even Tom Sheridan, who originated the idea of LOA in 1978,[17] explained recently that LOA was never intended to be "a prescription for designing automation" and that the NHTSA's categories for automated vehicles is a key example of "LOA that are not appropriate to [their] given context," not only in design but also in taxonomy and communication.[16,d]

The scientific literature shows that today's automated vehicles and corresponding LOA are characterized by the same serious design and communication flaws that human-automation interaction engineers have been fighting for nearly 70 years: automating as much as possible without concern for the human operator's capabilities or needs; relying on hidden, interdependent and coupled tasks for safety; and requiring the operator to immediately take over

[d] For a thorough discussion of the problems of current single-dimensional LOA and how they can be modified to account for human capabilities and operational needs, see the special issue on Advancing Models of Human-Automation Interaction in the *Journal of Cognitive Engineering and Decision Making* (http://bit.ly/3a9rnRG).

control in emergency situations without explicit support.

To make some of these reasons more salient, imagine you are part of a two-person team required to complete an assignment. Imagine that only your teammate was given the instructions for what was needed to complete the assignment. Conversely, you were only told that at some point your teammate may be unable to complete the assignment and, without prior notice, you will need to immediately finish it. You were also told that if your team fails to complete the assignment, it is entirely your fault.

Is this a recipe for good teamwork and success? Would you feel the need to constantly monitor your teammate? Would you feel like you have all the responsibility for the outcome but limited or no ability to affect it? At what point would it be easier to just do the work on your own?

With this example in mind, consider the definition of NHTSA's Level 2 Partial Driving Automation. This is currently the highest level of automation allowed without formal regulation in many U.S. states and the level for each of the five fatal Tesla accidents: "SAE J3016 Level 2 Partial Driving Automation: The driving automation system (while engaged) performs part of the dynamic driving task by executing both the lateral and the longitudinal vehicle motion control subtasks, and disengages immediately upon driver request; The human driver (at all times) performs the remainder of the [dynamic driving task] not performed by the driving automation system; supervises the driving automation system and intervenes as necessary to maintain safe operation of the vehicle; determines whether/when engagement and disengagement of the driving automation system is appropriate; immediately performs the entire [dynamic driving task] whenever required or desired."

Level 2 is the first point where the automation assumes full control of the foundational "lateral and longitudinal vehicle motion control subtasks" typically performed by human drivers such as lane centering, parking assist, and adaptive cruise control. The first stated role of the human driver in Level 2 is to "(at all times) [perform] the remainder of the [dynamic driving task] not performed by the driving automation system." These remaining tasks include supervising the automation and intervening as necessary based on object and event detection.

This is where LOA begins to show itself to be inappropriate for design, taxonomy, or communication of the safety-critical aspects of human-automation interaction in driving contexts as alluded to by Sheridan.[16] These remaining tasks are the textbook definition of leftover allocation: automate as many tasks as technology will permit and assume the human will pick up whichever tasks are left over.[2] Leftover allocation often results in incoherent sets of tasks and situations where humans are being required to monitor automation or the environment for conditions beyond which the automation can operate[18]—situations in which humans are ineffective.[13]

Level 2 is oversimplifying and obscuring the interdependence of the human driver and the automated driving system, assuming that the human driver's leftover tasks are complete, coherent, and capable of being performed. By focusing on "who does what," instead of emphasizing "how to work together," the LOA is giving "the illusion that we can successfully deploy automation by simply assigning functions to automation that were once performed by people ... [Neglecting] the fact that such assignments do not simply substitute automation for people but create new functions for the people who are left to manage the automation."[11]

Level 2's distribution of tasks is particularly troubling because engineers have known since the 1950s that monitoring is not a task humans can maintain for extended periods of time.[7] When a driver's interactions are limited to monitoring, they will lose real-time situation awareness, which can result in surprises. Workload will spike during off-nominal situations and be excessively low during normal operations between spikes, ultimately leading to humans who are notionally "in-the-loop" becoming, practically, "out-of-the-loop."[2] These spikes and lulls in workload can lead to the well-recognized problem of automation bias where humans will tend to disregard or not search for contradictory information in light of an automated judgment or decision that is accepted as correct.[15] Beyond automation bias, the lack of system interaction over a prolonged period prevents the human from acquiring expertise in the first place and can lead to long-term knowledge and skill degradation.[6] Combining this degradation with an incoherent set of leftover tasks will make it all but impossible for a driver to make an informed decision in an emergency situation.

Drivers are sold the fantasy of being a passenger at times, but to the manufacturer they never stopped being the fully liable driver.

The Level 2 Partial Automation Vehicle standard concludes with a final, fatal flaw: requiring the human operator to determine "whether/when engagement and disengagement of the driving automation system is appropriate," and if disengagement is necessary, "immediately [perform] the entire [dynamic driving task]."

In complex work environments such as automated vehicles where many tasks are interdependent and hidden, the driver is unlikely to know when disengagement is "appropriate"—especially given the ambiguity built into the SAE standard.[e] Studies have shown that these hidden interdependencies can result in insufficient coordination and exacerbate workload lulls and spikes.[6] This makes for a prototypically brittle human-automated system because there is no discussion of how the human operator should be supported during disengagement or takeover in emergency situations.[14]

[e] Two notable stipulations in the SAE standard expand the number of vehicles states that the driver would be required to monitor. By definition, "Levels are assigned, rather than measured, and reflect the design *intent* for the driving automation system feature as defined by its manufacturer" (8.2, emphasis added). Even further, the standard states that a system levels are not fixed and can deliver multiple features at different levels under varying conditions (8.4).

With this extensive history of human-automation interaction science, we can now perform the foreseeability analysis the law requires: Is there existing scientific evidence for a relationship between the accidents like the one that killed Mr. Huang and the design of Level 2 Partial Automation Vehicles?

In short, yes. Nearly 70 years of research argues against depending on human supervision of automation in complex, safety-critical environments without express consideration of the interdependent capabilities and needs of both the automation *and* the human. It is insufficient, inappropriate, and dangerous to automate everything you can and leave the rest to the human. It is insufficient, inappropriate, and dangerous for NHTSA to allow automated vehicles to be designed this way.

Beyond the research, consider the paradoxical expectations for drivers who purchase and operate these automated vehicles. Drivers are sold the fantasy of being a passenger at times,[f] but to the manufacturer they never stopped being the fully liable driver.

NHTSA seems to have acknowledged the surface of these issues by providing human factors design guidance for Levels 2 and 3 because "safe and efficient operation … requires [vehicles] be designed in a manner consistent with driver limitations, capabilities, and expectations."[5]

However, this NHTSA guidance does not address the fundamental crisis of confidence in the LOA framework: Can LOA appropriately regulate operations in complex work environments like automated vehicles?[9,11] Does NHTSA's LOA simply need to be implemented better? Or does NHTSA need to completely reimagine their framework beyond LOA's who-does-what perspective?

To answer this question, NHTSA should follow its own advice that "lessons learned through the aviation industry's experience with the introduction of automated systems may be instructive and inform the development of thoughtful, balanced approaches."[1]

Designers of automated vehicles face the same decisions today that aircraft designers have faced for decades.

In 1989, in response to high-profile fatal accidents, the Air Transport Association of America (ATA) established a task force to examine the impact of automation on aviation safety. The task force's prescient conclusion remains true today:[3] "During the 1970s and early 1980s … the concept of automating as much as possible was considered appropriate. The expected benefits were a reduction in pilot workload and increased safety … Although many of these benefits have been realized, serious questions have arisen and incidents/accidents have occurred which question the underlying assumption that maximum available automation is always appropriate or that we understand how to design automated systems so that they are fully compatible with the capabilities and limitations of the humans in the system."

Designers of automated vehicles face the same decisions today that aircraft designers have faced for decades. Automation has the potential to bring all the benefits of safety, reliability, economy, and comfort to our roads that have been brought to our airspace. But vehicle designers like Tesla and regulators like the NHTSA cannot abdicate their responsibility to stop foreseeable and preventable accidents by blaming the driver any more than aircraft designers can blame pilots.

Aviation has already learned that tragedy should not be the only time regulations and designs are reconsidered. As automated vehicles begin driving in public spaces, entrusted with the lives of drivers, passengers, and pedestrians, these vehicle designers and regulators must learn from aviation's tragic history of tombstone design, rather than repeating it.

[f] A survey of 1,212 owners of automated vehicles revealed that the "prevalence of drivers' willingness to engage in other activities, look away from the roadway or rely on the technology to the exclusion of ordinary safe driving practices … may indicate lack of understanding or appreciation of the fact that these technologies are designed to assist the driver, and that the driver is still required to be attentive and in control of the vehicle at all times to ensure safety."[12]

References

1. Automated Vehicles 3.0: Preparing for the Future of Transportation. Federal Policy Framework. National Highway Transportation Safety Administration, U.S. Department of Transportation, 2018.
2. Bainbridge, L. Ironies of automation. *Automatica 19*, 6 (June 1983), 775–779.
3. Billings, C.E. *Aviation Automation: The Search for a Human-Centered Approach.* Mahway, NJ, 1997.
4. Bradshaw, J.M. et al. The seven deadly myths of "autonomous systems." *IEEE Intelligent Systems 13* (2013), 2–9.
5. Campbell, J.L. et al. *Human Factors Design Guidance for Level 2 and Level 3 Automated Driving Concepts.* Report No. DOT HS 812 555. National Highway Transportation Safety Administration, 2018.
6. Feigh, K.M. and Amy R Pritchett, A.R. Requirements for effective function allocation: A critical review. *Journal of Cognitive Engineering and Decision Making 8*, 1 (Jan. 2014), 23–32.
7. Fitts, P.M. *Human Engineering for an Effective Air-Navigation and Traffic-Control System.* Technical Report. Division of National Research Council, 1951.
8. Grady, M.F. Proximate cause decoded. *UCLA Law Review 50* (2002), 293–335.
9. Jamieson, G.A. and Skraaning, Jr.. G. Levels of automation in human factors models for automation design: Why we might consider throwing the baby out with the bathwater. *Journal of Cognitive Engineering and Decision Making 12*, 1 (2018), 42–49.
10. J3016: Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles [JUN 2018]. Surface Vehicle Recommended Practice. SAE International, 2018.
11. Lee, J.D. Perspectives on Automotive Automation and Autonomy. *Journal of Cognitive Engineering and Decision Making 12*, 1 (Jan. 2018), 53–57.
12. McDonald, A., Carney, C., and McGehee, D. *Vehicle Owners' Experiences with and Reactions to Advanced Driver Assistance Systems.* AAA Foundation for Traffic Safety, 2018.
13. Molloy, R. and Parasuraman, R. Monitoring an automated system for a single failure: Vigilance and task complexity effects. *Human Factors 38* (1996), 311–322.
14. Norman, D.A. The 'problem' with automation: Inappropriate feedback and interaction, not 'over-automation'. Philosophical *Transactions of the Royal Society B: Biological Sciences 327*, 1241 (1990), 585–593.
15. Parasuraman, R. and Riley, V. Humans and automation: Use, misuse, disuse, abuse. *Human Factors: The Journal of the Human Factors and Ergonomics Society 39*, 2 (Feb. 1997), 230–253.
16. Sheridan, T.B. Comments on "Issues in Human–Automation Interaction Modeling: Presumptive Aspects of Frameworks of Types and Levels of Automation" by David B. Kaber. *Journal of Cognitive Engineering and Decision Making 12*, 1 (Jan. 2018), 25–28.
17. Sheridan, T.B. and Verplank, W.L. *Human and Computer Control of Undersea Teleoperators.* MIT Man-Machine Systems Laboratory, Cambridge, MA, 1978.
18. Wiener, E.L. and Curry, R.E. Flight-deck automation: Promises and problems. *Ergonomics 23* (1980), 995–1011.

For more on this topic, see Awad et al. on p. 48.

Marc Canellas (marc.c.canellas@gmail.com) is the Vice-Chair of the IEEE-USA Artificial Intelligence and Autonomous Systems Policy Committee, and a Cybersecurity Service Scholar and Jacobson Business Scholar at the New York University School of Law in New York, NY, USA.

Rachel Haga (rachel.haga@gmail.com) is a member of the IEEE-USA Artificial Intelligence and Autonomous Systems Policy Committee, and a Data Scientist at Elicit Insights in New York, NY, USA.

The contents of this Viewpoint are solely the responsibility of the authors.

DOI:10.1145/3380445

Benjamin C. Pierce, Michael Hicks, Crista Lopes, and Jens Palsberg

Viewpoint
Conferences in an Era of Expensive Carbon

Balancing sustainability and science.

A BROAD SCIENTIFIC CONSENSUS warns that human emissions of greenhouse gases are warming the earth. This is a present-day emergency: the UN's Intergovernmental Panel on Climate Change (IPCC) says a 40% decrease in emissions is needed by 2030 to avoid irreversible damage.[10] Reductions on this scale require urgent and sustained commitment at all levels of society—not only national, state, and city governments, but also universities, companies, and scientific societies.

Indeed, scientific societies have an especially important role to play, since, for many members, travel to conferences represents a substantial or even dominant part of their individual contribution to climate change. A single round-trip flight from Philadelphia, PA to Paris, France typically emits the equivalent of approximately 1.8 tons of carbon dioxide (CO_2e, or informally "carbon") per passenger.[14] This is a significant fraction of the total yearly emissions for an average resident of the U.S. (16.5 tons) or Europe (7 tons).[5] Moreover, these emissions have no near-term technological fix, as jet fuel is difficult to replace with renewable energy sources.[15]

How should ACM respond to these facts?[a]

In 2016, ACM's Special Interest Group on Programming Languages (SIGPLAN) convened an ad hoc Climate Committee to consider this question.[2] After investigating many options,[7] we are putting forward two concrete proposals. First, all ACM conferences should publicly account for the CO_2e emitted as a result of putting them on—in particular, from travel to the conference. Second, ACM should put a price on carbon in conference budgets, creating a steady pressure on organizers to reduce their footprints.

Mandate Public Accountability

Our first proposal is modest: **Every ACM-sponsored conference should publicly report its carbon footprint.** These reports should be collected in a central place, in a uniform format. Most conferences' footprints will be dominated by participants' air travel, but the data gathered should go beyond this to include ground transportation, travel to in-person program committee meetings, and estimated emissions from hotels and food.

ACM should develop tools to gather and publicize this data. For example,

[a] See http://bit.ly/2suhQUg for information pertaining to the ACM Carbon Offset Program, including a link to a carbon offset calculator.

IMAGE BY RAWPIXEL.COM

SIGPLAN recently built an air-travel-focused carbon calculator for conferences.[13] Users can upload conference registration data, and the calculator will estimate the CO_2e cost of air travel.

There is some reputational risk to ACM in taking the step of publicizing its carbon footprint: the numbers are likely to be high, and they may be used to criticize both ACM and the broader academic community. But making this information available is a crucial first step: we cannot manage what we have not measured.

Easy trimming. One effect of public accounting will be to nudge conference organizers and attendees to change their behavior. By analogy, chain restaurants in the U.S. are now required by law to post calorie counts of food items on menu boards; studies show that enlightened customers order, on average, up to 50 fewer calories a day.[6]

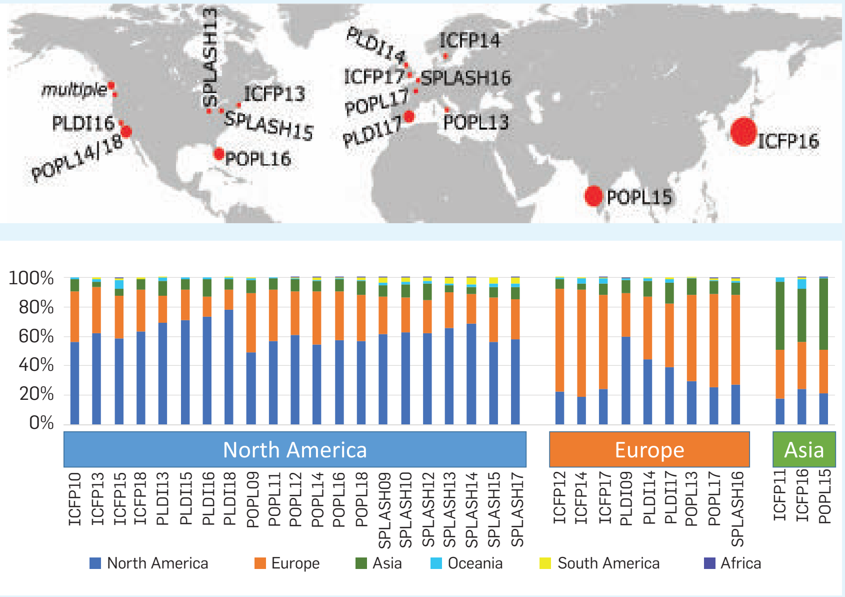
Similarly, SIGPLAN has been considering how to reduce emissions, informed by an accounting of its own carbon footprint. This discussion has led the organizers of two flagship conferences (POPL and ICFP) to switch from in-person to online program committee meetings, joining a trend among other SIGs, and has prompted several conferences to increase investments in livestreaming and video recording to support remote participation.

Difficult choices. However, while public accounting of emissions will encourage easy reductions, it is not likely, by itself, to induce major shifts in behavior. Science is a fundamentally social process, and the conference system accelerates scientific research through high-bandwidth interaction, direct dissemination of results, network building, and serendipitous cross-fertilization. Organizers and attendees will naturally be reluctant to consider changes that might threaten these benefits.

To illustrate the challenges, consider the problem of choosing a conference location that minimizes emissions from participants' travel. Using recent registration data from four SIGPLAN conferences, the accompanying figure shows two ways of looking at the relation between locations and emissions. The top diagram shows an estimated per-participant CO_2e footprint for each instance of each conference over the past 10 years (excluding a few for which we had difficulty getting data), with larger dots representing higher emissions. Eyeballing this diagram, it might seem that carbon-conscious organizers should hold all of these conferences in either the northeast U.S. or western Europe every year. But the bottom visualization tells a different story. The horizontal colored bar at the bottom represents the continent on which each conference was held, and each vertical bar gives a breakdown of the participants in that conference, colored according to the continent of their work address. A glance at the colors makes clear that—though a minority come from far away—the majority of participants in each conference are local to the region where the conference is being held. This suggests that always locating conferences in the same one or two places would significantly impact the diversity of the research community by discouraging participation from other parts of the world. Indeed, one might conclude that, from the point of view of strengthening the research community, conferences should move around as much as possible! These disparate perspectives suggest that significantly reducing conference emissions may require genuinely painful compromises. The impulse to ignore the issue is entirely understandable.

Carbon footprint per participant for travel to recent SIGPLAN conferences.

The smallest dot (ICFP 14, in Gothenburg, Sweden) represents 0.9 tons of CO_2e per participant; the largest (ICFP 16, in Nara, Japan) represents 1.94 tons per participant. Bottom: Breakdown of continent-of-origin for participants in each conference. Colored bars represent percentages of participants whose home city is in each continent: blue for North America, orange for Europe, green for Asia.



However, the present trajectory of world emissions is unsustainable: difficult choices will have to be made, and soon, if ACM is to play its part by reducing its own emissions. How do we motivate organizers to face these choices?

Put a Price on Carbon

This dilemma is a microcosm of one faced by all of society. To address it, many policy experts advocate using some form of carbon pricing to impose a concrete, immediate cost on emissions.[8] Doing so makes manifest the hidden environmental cost of emissions, incentivizing CO_2e-reducing changes without mandating exactly which ones, and thus allowing for creative and efficient responses. Continuing the junk-food analogy, some municipalities including Berkeley and Philadelphia have imposed a per-calorie tax on soft drinks; studies found that doing so significantly reduced consumption.[11]

Thus, our second proposal is that **ACM should impose a surcharge on conferences based on their carbon footprint**. The charge should start low and increase steadily and predictably, year on year. Conference organizers can then choose how best to balance their budgets—whether by decreasing per-participant emissions, decreasing (physical) participation, increasing registration fees, soliciting corporate sponsorship, or other means. In this process, a primary concern should be to find ways of reducing the financial burden of such a surcharge on those disproportionately affected by it—students without grant support, participants from developing areas, and so forth. Well-funded participants should subsidize the carbon surcharges of less-wealthy ones.

Ideally, at some point, governments will impose carbon pricing uniformly, and all carbon-intensive activities will have to pay it. But ACM can send a strong message about the importance of this issue—and get ahead of the coming changes—by acting now.

Precipitating change. What should ACM do with the funds collected from this surcharge? One obvious possibility is purchasing carbon offsets.[9] A carbon offset is sold by a vendor, who uses the funds to finance an activity that permanently removes or avoids emitting some amount of greenhouse gases. The veracity and permanence of this activity is certified by a watchdog organization. (For example, planting trees is often considered not to be a certifiable activity, since it is difficult to guarantee they will not be cut down; reductions from installing methane capture devices on landfill sites or buying fuel-efficient stoves to replace open-fire cooking in poor communities are easier to predict.) Many organizations, including companies such as Google, Dell, Microsoft, General Motors, Delta Airlines, Lyft, and Expedia, as well as universities, academic societies, and even energy companies such as Exxon, now use carbon offsets to reduce their net footprint. ACM conferences should consider doing the same, and the purchases should be included in the public accounting we are proposing (see ACM's Carbon Offset Program http://bit.ly/2suhQUg).

Beyond buying offsets, one can imagine many good uses for the funds generated from a carbon surcharge: defraying the costs of virtualizing conferences (livestreaming, and so forth), and supporting "green" computing research.[1] As an example of the last, ACM could help fund a cross-cutting research initiative specifically aimed at understanding how to best replace or approximate the socializing and networking aspects of conferences in a virtual setting.

At some point, governments will impose carbon pricing uniformly, but ACM can send a strong message about the importance of this issue by acting now.

Ultimately, however, carbon offsets and other "good works" cannot substitute for real reductions in emissions:[2,3] they are, at best, a short-term expedient that buys time to agree on more difficult cuts. Indeed, the main goal of carbon pricing should be to stimulate creative rethinking of the conference model itself—for example, seriously considering alternatives such as rapid-turnaround journal-only publishing models, yearly mega-conferences, and entirely virtual conferences.[4] A potential sticking point is that some of these will significantly reduce conference revenues, in turn impacting the income stream of ACM itself; this could make emissions reduction politically problematic unless ACM's conference-focused business model is also adjusted.

What should the price be? Another key issue in implementing this second proposal will be how to set a price that reflects the true social cost of carbon, without unduly harming the scientific community. Initial data gathering will play a role, and a steady and predictable annual increase is a necessary component, but setting both the initial price and the slope of the ramp will likely be difficult political decisions. One measurable target, in line with the latest evidence from climate science, would be to tune the parameters with a goal of halving actual emissions every decade from now on, following a recently proposed "Carbon Law"[12]

Conclusion

The climate crisis is too urgent to leave to world leaders to address at their own pace: Organizations at every scale, including ACM, must confront their own contributions, raise awareness and foster discussion among their membership,[1] and establish new ways of doing business in the lower-carbon future that is now upon us. We in ACM should do our part by mandating public accounting of conference carbon footprints and by putting a concrete price on the carbon we use. C

References

1. acm-climate mailing list; http://bit.ly/3OnoYhO
2. Anderson, K. The inconvenient truth of carbon offsets. *Nature News 484*, 7392 (2012), 7.
3. Carbon offsets are not our get-out-of-jail free card. United Nations Environment Programme; http://bit.ly/2NnrsHu
4. CO_2 emissions (metric tons per capita). A Nearly Carbon-Neutral Conference Model: White Paper/Practical Guide; http://bit.ly/2TlLbLo
5. CO_2 emissions (metric tons per capita). The World Bank Data Bank, using data from the Carbon Dioxide Information Analysis Center, Environmental Sciences Division, Oak Ridge National Laboratory, Tennessee, USA; http://bit.ly/2RfCnUJ
6. Galewitz, P. Obamacare's calorie count rules go into effect. CNN, May 2018; https://cnn.it/3891Dmi
7. Hicks, M.W., Lopes, C., and Pierce, B.C. Engaging with climate change: Some possible steps for SIGPLAN (preliminary report of the SIGPLAN Climate Committee, version 1.2), June 2018; http://bit.ly/2NoB9pi
8. Jenkins, J. Why carbon pricing falls short. Kleinman Center for Energy Policy, April 2019; http://bit.ly/3aaEtxR
9. Kim, R. and Pierce, B.C. Carbon offsets: An overview for scientific societies. June 2018: http://bit.ly/2FTqsGM
10. Masson-Delmotte, V. et al., Eds. Global warming of 1:5_C: An IPCC Special Report on the impacts of global warming of 1:5_C above pre-industrial levels and related global greenhouse gas emission pathways, in the context of strengthening the global response to the threat of climate change, sustainable development, and efforts to eradicate poverty. 2018.
11. Miller, G. The global soda tax experiment. *Knowable Magazine* (Oct. 2019); http://bit.ly/35PJ2u5
12. Rockström, J. et al. A roadmap for rapid decarbonization. *Science* 355, 6331:1269–1271 (2017).
13. Rolnick, D. et al. Tackling climate change with machine learning. arXiv preprint arXiv:1906.05433, 2019.
14. SIGPLAN and Climate Change; http://bit.ly/3OnoYhO
15. Viswanathan, V., Sripad, S., and Fredericks, W.L. Why aren't there electric airplanes yet? *Universal-Sci.* (Nov. 2018); http://bit.ly/2ToCYpL

Benjamin C. Pierce (bcpierce@cis.upenn.edu) is a Professor of Computer and Information Science at the University of Pennsylvania, in Philadelphia, PA, USA.

Michael Hicks (mwh@cs.umd.edu) is a Professor of Computer Science at the University of Maryland, College Park, MD, USA.

Crista Lopes (lopes@uci.edu) is a Professor of Software Engineering at the University of California, Irvine, CA, USA.

Jens Palsberg (palsberg@cs.ucla.edu) is a Professor of Computer Science at the University of California, Los Angeles, CA, USA.

DOI:10.1145/3379512

Article development led by acmqueue
queue.acm.org

The hardware root of trust.

BY JESSIE FRAZELLE

Securing the Boot Process

THE BOOT SEQUENCE for a machine typically starts with the BMC (baseboard management controller) or PCH (platform controller hub). In the case of an Intel CPU, the Intel Management Engine runs in the PCH and starts before the CPU. After configuring the machine's hardware, the BMC (or PCH, depending on the system) allows the CPU to come out of reset. The CPU then loads the boot (unified extensible firmware interface, UEFI) firmware from the SPI (serial peripheral interface) flash. The boot firmware then accesses the boot sector on the machine's persistent storage and loads the bootloader into the system memory. The boot firmware then passes execution control to the bootloader, which loads the initial OS image from storage into system memory and passes execution control to the operating system. For example, in popular Linux distros, GRUB (derived from Grand Unified Bootloader) acts as the bootloader and loads the operating system image for the machine.

This is much like a relay race where one team member passes a baton to another to win the race. In a relay race, you hopefully know the members of your team and trust them to do their part for the team to get to the finish line. With machines, this chain of trust is a bit more complex. How can we verify that each step in the boot sequence is running software we know is secure? If our hardware or software has been compromised at any point in the boot sequence then the attacker has the most privilege on our system and likely can do anything they want.

The goal of a hardware root of trust is to verify that the software installed in every component of the hardware is the software that was intended. This way you can verify and know without a doubt whether a machine's hardware or software has been hacked or overwritten by an adversary. In a world of modchips,[16] supply chain attacks, evil maid attacks,[7] cloud provider vulnerabilities in hardware components,[2] and other attack vectors it has become more and more necessary to ensure hardware and software integrity. This is an introduction to a complicated topic; some sections just touch the surface, but the intention is to provide a full picture of the world of secure booting mechanisms.

Trusted platform module (TPM). A TPM is a standard for a dedicated microchip designed to secure hardware through integrated cryptographic keys. TPM was standardized by the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC) in 2009 as ISO/IEC 11889.[9] The TPM is typically installed on the motherboard of a computer, and it communicates with the remainder of the system by using a hardware bus.

IMAGE BY VECTOR FUSION ART

A TPM has the following features:[18]

- A random number generator;
- A way to generate cryptographic keys;
- Integrity measurement;
- Attestation;
- Wrapping/binding keys; and,
- Sealing/unsealing keys.

Integrity measurement. Measurement is the process by which information about the software, hardware, and configuration of a system is collected and digested. At load-time, the TPM uses a hash function to fingerprint an executable and its configuration. These hash values are used in attestation to reliably establish code identity to remote or local verifiers. The hash values can also be used in conjunction with the sealed storage feature. A secret can be sealed along with a list of hash values of programs that are allowed to unseal the secret. This allows the creation of data files that can only be opened by specific applications.

Attestation reports the state of the hardware and software configuration. The integrity measurement software in charge of creating the hash key used for the configuration data determines the extent of the summary. The goal of attestation is to prove to a third party that your operating system and application softwa e are intact and trustworthy. The verifier trusts that attestation data is accurate because it is signed by a TPM whose key is certified by the certificate authority (CA). TPMs are manufactured with a public/private key pair built into the hardware, known as the endorsement key. The endorsement key is unique to a specific TPM and is signed by a trusted CA. The trust for attestation data is dependent on the trust for the CA that originally signed the endorsement key.

Attestation can reliably tell a verifier what applications are running on a client machine, but the verifier must still make the judgment about whether each given piece of software is trustworthy.

Wrapping/binding a key. Machines that use a TPM can create cryptographic keys and encrypt them so that they can only be decrypted by the TPM. This process, known as *wrapping* or *binding* a key, can help protect the key from disclosure. Each TPM has a master wrapping key, also known as the storage root key, which is stored within the TPM itself. The private portion of a storage root key, or endorsement key, that is created in a TPM is never exposed to any other device, process, application, software, or user.

Sealing/unsealing a key. Machines that use a TPM can also create a key that has not only been wrapped but is also tied to certain platform measurements. This type of key can be unwrapped only when those platform measurements have the same values that they had when the key was created. This process is known as *sealing* the key to the TPM. Decrypting the key is called *unsealing*. The TPM can also seal and unseal data that is generated outside the TPM. With this sealed key and software, you can lock data until specific hardware or software conditions are met.

Custom silicon. It is important to note the limitations of TPMs and some solutions to those. TPMs can attest the firmware running on a machine is the firmware we want to run, but there is no mechanism in a TPM for verifying that the code is secure. It is up to the user to verify the security of the firmware and to ensure it does not contain any backdoors, which is impossible if the code is proprietary.

When booting a machine securely, you want the first instruction run on that machine to be the one you would expect to run. A TPM is insufficient for verifying the actual bits of code to be executed are secure, so a few companies created their own silicon for expanding on the security of TPMs.

Google's Titan

For Google's infrastructure as well as Chromebooks, Google expanded on the security of the TPM with their own chip Titan. Google open sourced[5] a version of Titan[14] (with both specs and code), which is under active development, in October of 2019. In creating Titan, Google added two new features that did not exist with TPMs: first-instruction integrity and remediation.

First-instruction integrity allows verification of the earliest code that runs on each machine's startup cycle. Titan observes every byte of boot firmware by interposing itself between the boot firmware flash (BIOS) of the BMC (or PCH) and the main CPU via the SPI bus. Therefore, the boot sequence for a machine with a Titan chip is different from a normal boot sequence.

The boot sequence with Titan is as follows:

- Titan holds the machine in reset.
- Titan's application processor executes code from its embedded read-only memory (boot ROM).
- Titan runs a memory built-in self-test to ensure all memory (including ROM) has not been tampered with.
- Titan verifies its own firmware using public key cryptography and mixes the identity of this verified code into Titan's key hierarchy.
- Titan loads the verified firmware.
- Titan verifies the host's boot firmware flash (BIOS/UEFI).
- Titan signals readiness to release the rest of the machine from reset.
- The CPU loads the basic firmware (BIOS/UEFI) from the boot firmware flash, which performs further hardware/software configuration.
- The rest of the standard boot sequence continues.

Holding the machine in reset while Titan cryptographically verifies the boot firmware, Titan enables the verification of the first instruction. Titan knows what boot firmware and OS booted on our machine from the very first instruction. Titan even knows which microcode patches may have been fetched before the boot firmware's first instruction.

Remediation. What happens when we need to patch bugs in Titan's firmware? This is where remediation comes into play. In the event of patching bugs in the Titan firmware, trust can be reestablished through remediation. Remediation is based on a strong cryptographic identity. To provide a strong identity, the Titan chip manufacturing process generates unique keying material for each chip. The Titan-based identity system not only verifies the provenance of the chips creating the certificate signing requests (CSRs), but also verifies the firmware running on the chips, as the code identity of the firmware is hashed into the on-chip key hierarchy. This property allows Google to fix bugs in Titan firmware and issue certificates that can only be wielded by patched Titan chips.

The Titan-based identity system enables back-end systems to securely provision secrets and keys to individual Titan-enabled machines, or jobs running on those machines. Titan is also able to chain and sign critical audit logs, making those logs tamper evident. This ensures audit logs cannot be altered or deleted without detection, even by insiders with root access to the relevant machine.

Microsoft's Cerberus

Microsoft open sourced[11] the specs for their chip, Cerberus. (At the time of writing this article, only the specs have been open sourced). Like Titan, Cerberus interposes on the SPI bus where firmware is stored for the CPU. This allows Cerberus to continuously measure and attest these accesses to ensure firmware integrity and thereby protect against unauthorized access and malicious updates.

Apple's T2

Apple is a poster child for secure booting devices. Most people remember when the FBI wanted a backdoor into iPhones and Tim Cook refused.[10] Between Macs, iPhones, and Chromebooks, an industry standard for products includes security by default.

For Apple machines, secure boot is done with their T2 chip,[1] Ivan Krstić of Apple gave a talk at Black Hat[12] detailing the boot process for a Mac with Apple's T2 chip. Unlike Titan and Cerberus which interpose on the SPI flash, T2 provides the firmware and boots the CPU over an eSPI (Enhanced Serial Peripheral Interface) bus.

Apple's requirements for T2 were the following:

- Signature verification of complete boot chain.
- System Software Authorization (server-side downgrade protection).
- Authorization "personalized" for the requesting device (not portable).
- User authentication required to downgrade secure boot policy.
- Secure boot policy protected against physical tamper.
- System can always be restored to known-good state.

The boot sequence for a machine using a T2 chip is as follows:

- The machine is powered on.
- T2 ROM is loaded and executed.
- T2 ROM passes off to iBoot, the bootloader.
- The bootloader executes the bridgeOS kernel, the kernel for the T2 chip.
- The bridgeOS kernel passes off to the UEFI firmware for the T2 chip.
- The T2 chip then allows the CPU out of reset and loads the UEFI firmware for the CPU.

- The UEFI firmware for the CPU then loads macOS booter, the bootloader.
- The macOS booter then executes the macOS kernel.

One important design element of the T2 chip is how Apple verifies the version of MacOS running on a computer. T2 verifies the hash of MacOS against a list of approved hashes for running. Apple is in a unique position to have this level of verification since they own the entire stack and prevent users from running any other OS on their devices. If you would like to go deeper on the internals of the T2 chip, I would suggest reading the slides for Ivan Krstić's Black Hat talk.[12]

Platform firmware resiliency. Chip vendors are investing in platform firmware resiliency (PFR) based on National Institute of Standards and Technology (NIST) guidelines.[15] These guidelines focus on ensuring the firmware remains in a state of integrity, detecting when it has been corrupted, and recovering the pieces of firmware back to a state of integrity.

PFR addresses the vulnerability of enterprise servers that contain multiple processing components, each having its own firmware. This firmware can be attacked by hackers who may surreptitiously install malicious code in a component's flash memory that hides from standard system-level detection methods and leaves the system permanently compromised.

The PFR specification is based on the following principles:

- *Protection:* Ensures firmware code and critical data remain in a state of integrity and are protected from corruption, such as the process for ensuring the authenticity and integrity of firmware updates.
- *Detection:* Detect when firmware code and critical data have been corrupted.
- *Recovery:* Restore firmware code and critical data to a state of integrity in the event that any such firmware code or critical data are detected to have been corrupted, or when forced to recover through an authorized mechanism.

Vendors have been building features around the NIST guidelines for PFR. Intel[8] and Lattice Semiconductors[13] each have a product.

UEFI Secure Boot[21] is designed to ensure that EFI binaries that are executed during boot are verified, either through a checksum or a valid signature, backed by a locally trusted certificate. When a machine using UEFI Secure Boot powers on, the UEFI firmware validates each EFI binary either has a valid signature or the binary's checksum is present on an allowed list. Counter to the allow list is a deny list that is also checked to ensure no binary's checksum or signature exists on it. Users can configure the list of trusted certificates and checksums as EFI variables. These variables get stored in non-volatile memory used by the UEFI firmware environment to store settings and configuration data.

Attestation can reliably tell a verifier what applications are running on a client machine, but the verifier must still make the judgment about whether each given piece of software is trustworthy.

The UEFI kernel is extremely complex and has millions of lines of code. It consists of boot services and runtime services. The specification[19] is quite verbose and complex. The UEFI kernel is a common vector for many vulnerabilities since it has some of the same proprietary code used on many different platforms. The UEFI kernel is shared on multiple platforms, making it a great target for attackers. Additionally, since only UEFI can rewrite itself, exploits can be made persistent. This is because UEFI lives in the processor's firmware, typically stored in the SPI flash. Even if a user were to wipe the entire operating system or install a new hard drive, an attack would persist in the SPI flash.

Intel's Boot Guard. Boot Guard is Intel's solution to verify the firmware signatures for the processor. Boot Guard works by flashing the public key of the BIOS signature into the field programmable fuses (FPFs), a one-time programmable memory inside Intel Management Engine (ME), during the manufacturing process. The machine then has the public key of the BIOS and it can verify the correct signature during every subsequent boot. However, once Boot Guard is enabled by the manufacturer, it cannot be disabled.

The problem with Boot Guard is that only Intel or the manufacturer has the keys for signing firmware packages. This makes it impossible to use coreboot, LinuxBoot, or any other equivalents as firmware on those processors. If you tried, the firmware would not be signed with the correct key, and the failed attempt to boot would brick the board.

Matthew Garrett wrote a great post about Boot Guard that highlights the importance of user freedom when it comes to firmware.[4] The owner of the hardware has a right to own the firmware as well. Boot Guard prevents this. In the security keynote at the 2018 Open Source Firmware Conference,[6] Trammel Hudson described how he found a vulnerability to bypass Boot Guard, CVE-2018-12169.[3] The bug[20] allows an attacker to use unsigned firmware and boot normally, completely negating the purpose of Boot Guard. Because Boot Guard is tied to the CPU, it does not have the control that a custom silicon hardware root of trust has when it comes to other firmware for components in the system.

System transparency. Mullvad wrote up a paper on what they call system transparency (ST),[17] which is aimed at facilitating trust for the components of a system by giving every server a unique identity, limiting the attack surface and mutable state in the firmware and allowing both owners and users to verify all software running on a platform starting from the first instruction executed after power on.

ST accomplishes these goals by following seven principles:

1. A key ceremony of each server to bind the server's unique identity with a difficult-to-forge physical artifact like a video.

2. *Physical write-protection of the firmware.* Writable code sections are a mutable state, so ST limits the possible changes to this critical piece of code. Read-only code also serves as a root of trust for all other software-enforced security mechanisms.

3. *Tamper detection.* Attackers cannot be stopped from changing the content of the firmware flash by replacing the actual chip. So, violations of the physical integrity of the server hardware need to be detectable.

4. *Measured boot.* ST has the goal to give all parties insight into what code was run as part of the system boot. A measured boot in combination with remote attestation allows third parties to acquire a cryptographic log of the boot.

5. *Reproducible builds.* Ensures that if a binary artifact is built once, it can be built again and again and produce the same artifact. This establishes a verifiable link between the human-readable code and the binary that was attested using the measured boot mechanism.

6. *Immutable infrastructure.* System transparency only works when changes to the operating system are limited. Allowing somebody to log into the system and make arbitrary changes invalidates all guarantees of a measured boot.

7. *Binary transparency log.* All firmware and OS images that can be booted on a system are signed by the system's owner and are inserted into a public, append-only log. Users of the system can monitor this log for new entries and catch malicious system owners booting backdoored firmware on new servers.

The Importance of Open Source Firmware

It is clear that securing the boot process with a hardware root of trust has various implementations throughout the industry. Without open source firmware, the proprietary bits of the boot process are still lacking the visibility and audibility to ensure our software is secure. Even if we can verify through a hardware root of trust that the hash of proprietary firmware is the hash we know to be true, we need visibility to the source code for the firmware for assurance it does not contain any backdoors. Through this visibility we can also gain ease of use in debugging and fixing problems without relying on a vendor.

Firmware is scattered throughout motherboards of machines and their components; it is in the CPU (central processing unit), NIC (network interface controller), SSD (solid-state drive), HDD (hard-disk drive), GPU (graphics processing unit), fans, and more. To ensure the integrity of a machine, all these components must be verified. In the future, these custom silicon chips will interpose not only on the SPI flash but also on every other device communicating with the BMC.

If you would like to help with the open source firmware movement, push back on your vendors and platforms you are using to make their firmware open source.

Acknowledgments

Thank you to Ivan Krstić, Matthew Garrett, Kai Michaelis, Fredrik Strömberg, and Trammell Hudson for their research and work in this area, which helped me to write this article.

Related articles on queue.acm.org

Security for the Modern Age
Jessie Frazelle
https://queue.acm.org/detail.cfm?id=3301253

Simulators: Virtual Machines of the Past (and Future)
Bob Supnik
https://queue.acm.org/detail.cfm?id=1017002

Automating Software Failure Reporting
Brendan Murphy
https://queue.acm.org/detail.cfm?id=1036498

References

1. Apple. Apple T2 Security Chip,2018; https://www.apple.com/mac/docs/Apple_T2_Security_Chip_Overview.pdf
2. Cimpanu, C. Hackers can hijack bare-metal cloud servers by corrupting their BMC firmware; https://zd.net/2MyXFLI
3. Common Vulnerabilities and Exposures, 2018; https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-12169
4. Garrett, M. Intel Boot Guard, 2015; Coreboot and user freedom; https://mjg59.dreamwidth.org/33981.html
5. Google Open Source Blog. OpenTitan—Open sourcing transparent, trustworthy, and secure silicon; https://opensource.googleblog.com/2019/11/opentitan-open-sourcing-transparent.html.
6. Hudson, T. Open Source Firmware Conference's Security Keynote; https://trmm.net/OSFC_2018_Security_keynote#Boot_Guard
7. Hudson, T. Thunderstrike EFI bootkit FAQ; https://trmm.net/Thunderstrike_FAQ#Does_anyone_actually_use_evil-maid_attacks.3F
8. Intel. Intel Data Center Block with Firmware Resilience, 2017; https://intel.ly/2POBjXj
9. ISO/IEC 11889-1:2009. Information technology—Trusted platform module; https://www.iso.org/standard/50970.html.
10. Kahney, L. The FBI wanted a back door to the iPhone. Tim Cook said no. *Wired* (Apr. 16, 2019); https://www.wired.com/story/the-time-tim-cook-stood-his-ground-against-fbi/.
11. Kelly, B. Open Compute Project—Project Cerberus Security Architecture Overview Specification, 2017; http://bit.ly/2sts9aO.
12. Krstic, I. Behind the scenes of iOS and Mac security, 2019; https://ubm.io/34rrmnY
13. Lattice Semiconductors. Universal Platform Firmware Resiliency (PFR)—Servers; http://www.latticesemi.com/Solutions/Solutions/SolutionsDetails02/PFR.
14. OpenTitan. Introduction to OpenTitan, 2019; https://docs.opentitan.org/.
15. Regenscheid, A. Platform firmware resiliency guidelines. NIST Special Publication 800-193, 2018; https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-193.pdf.
16. Robertson, J., Riley, M. The Big Hack: How China Used a Tiny Chip to Infiltrate U.S. Companies, 2018; https://bloom.bg/2PY108V
17. Strömberg, F. System transparency, 2019; https://mullvad.net/media/system-transparency-rev5.pdf.
18. Trusted Computing Group. TPM main, part 1, design principles, 2011; http://bit.ly/36GDQcK.
19. UEFI; https://uefi.org/specifications
20. Wang, J. Bug 1614 (CVE-2019-11098) – BootGuard TOCTOU vulnerability; https://bugzilla.tianocore.org/show_bug.cgi?id=1614
21. Wilkins, R. 2013. UEFI secure boot in modern computer security solutions; http://bit.ly/362a4Q2

Jessie Frazelle is the co-founder and Chief Product Officer of the Oxide Computer Company. Previously, she worked on various parts of Linux, including containers and also the Go programming language.

DOI:10.1145/3379510

Article development led by acmqueue
queue.acm.org

The resilience of Internet-facing systems relies on what is above the line of representation.

BY RICHARD I. COOK

Above the Line, Below the Line

PEOPLE WORKING ABOVE the line of representation continuously build and refresh their models of what lies below the line. That activity is critical to the resilience of Internet-facing systems and the principal source of adaptive capacity.

Imagine all the people involved in keeping your Web-based enterprise up and running suddenly stopped working. How long would that system continue to function as intended? Almost everyone recognizes the "care and feeding" of enterprise software systems requires more or less constant attention. Problems that require intervention crop up regularly—several times a week for many enterprises; for others, several times a day.

Publicly, companies usually describe these events as sporadic and minor—systemically equivalent to a cold or flu that is easily treated at home or with a doctor's office visit. Even a cursory look inside, however, shows a situation more like an intensive care unit: continuous monitoring, elaborate struggles to manage related resources, and many interventions by teams of around-the-clock experts working in shifts. Far from being hale and hearty, these are brittle and often quite fragile assemblies that totter along only because they are surrounded by people who understand how they work, how they fail, what can happen, and what to do about it.

What's Going On?

The intimate, ongoing relationship between tech software/hardware components and the people who make, modify, and repair them is at once remarkable and frustrating. The exceptional reach and capacity of Internet-based enterprises results from indissolubly linking humans and machines into a continuously changing, nondeterministic, fully distributed system.

Their general and specific knowledge of how and why those bits are assembled as they are gives these humans the capacity to build, maintain, and extend enterprise technology. Those bits continuously change, creating an absolute requirement to adjust and refresh knowledge, expectations, and plans. Keeping pace with this change is a daunting task, but it is possible—just—for several reasons:

The intimate, ongoing relationship between tech software/hardware components and the people who make, modify, and repair them is at once remarkable and frustrating.

1. *The people who intervene in failure are often the same people who built the stuff in the first place.* The diagnosticians and repairers are frequently the same people who designed, wrote, debugged, and installed the very software and hardware that are now failing. They participated in the intricacies, dependencies, and assumptions that produced and arranged these artifacts. Even when they did not, they often have worked and interacted with others and have learned along the way who contributed and who is expert in those areas. This sets the community apart from other operator communities (for example, pilots, nurses).

2. *The people who intervene have unprecedented access to the internals of the assemblies.* The fixers can look at source code, interrogate processes, and view statistical summaries of activities in near realtime. In no other domain is there so much detail available for troubleshooting problems. Admittedly, the huge volume of accessible material is a challenge as well: It can be difficult to find a meaningful thread of cause and effect and to trace that thread through to its sources. Here again the collective is often the critical resource—someone knows where and how things are connected and dependent so that the work of addressing an incident in progress often includes the work of figuring out what is germane and whose expertise matches the pattern.

3. *The continuing failures constantly redirect attention to the places where their understandings are incomplete or inaccurate.* There is an ongoing stream of anomalies that demand attention. The resulting engagement produces insight into the fragility, limitations, and perversities that matter at the moment. Anomalies are pointers to those areas where problems manifest, what Beth Long calls "the explody bits." These are also areas where further exploration is likely to be rewarding. This is valuable information, especially because continuous change moves the locus of failure. This is also why longitudinal collections of incidents so rarely prove useful: Past performance is no guarantee of future returns.

4. *There is a distinct community of practice with its own ethos.* The people who do this work form what Jean Lave and Etienne Wenger call a community of practice.[3] This is a tangled network characterized by communications and processes that simultaneously share knowledge, distribute responsibility, and provoke actions. The network has some remarkable features. New people are joining all the time, and their induction into the community leads its members to revisit old ground. Because so much learning takes place on the job and in real rather than simulated settings, it has qualities of a guild. Because the people involved change jobs frequently, the guild extends over time and across corporate boundaries. This produces diffusion of expertise across the industry and simultaneously creates a relationship mesh that bridges corporate boundaries.

The barriers to entry into this network are low. There is not yet a formal process of training nor certification of authority found in other domains (for example, medicine). This has promoted rapid growth of the community while also creating uncertainty that manifests in hiring practices (for example, code-writing exercises).

This community of practice appears to have a distinct ethos that puts great emphasis on keeping the system working and defending it against failures, damage, or disruption. The community values both technical expertise and the capacity to function under stress; membership in the community depends on having successfully weathered difficult and demanding situations. Similarly, the collective nature of work during threatening events encourages both cooperation and support. As Lave and Wenger observed for other communities of practice, mastery here is gained via "legitimate peripheral participation."

5. *The work is demanding and has remained challenging over time.* Keeping the enterprise going and growing requires the expertise and dedication that this group provides. Although technology enthusiasts have predicted

a diminishing role for people in the system, there is no sign of this happening. The intervals between breakdowns are so short and the measures required to remedy faults so varied that only a concerted and energetic effort to replenish the network and refresh its knowledge has any chance of success.

The Line of Representation

All these features are simultaneously products of the environment and enablers of it. They have emerged in large part because the technical artifacts are evolving quickly, but moreso because the artifacts cannot be observed or manipulated directly. Computing is detectable only via representations synthesized to show its passing. Similarly, it can be manipulated only via representations.

The accompanying figure shows an Internet-facing system. The horizontal line comprises all the representations available to people working above that line, including all the displays, screens, and other output devices, and keyboards and other input devices. Below this line lie the technical artifacts: code libraries, IDEs, test suites, compilers, CI/CD (continuous integration/continuous delivery) pipeline components, and the computational capacity itself including technology stacks and services. Above the line of representation are the people, organizations, and processes that shape, direct, and restore the technical artifacts that lie below that line.

People who work above the line routinely describe what is below the line using concrete, realistic language. Yet, remarkably, *nothing* below the line can be seen or acted upon directly. The displays, keyboards, and mice that constitute the line of representation are the *only* tangible evidence that anything at all lies below the line.

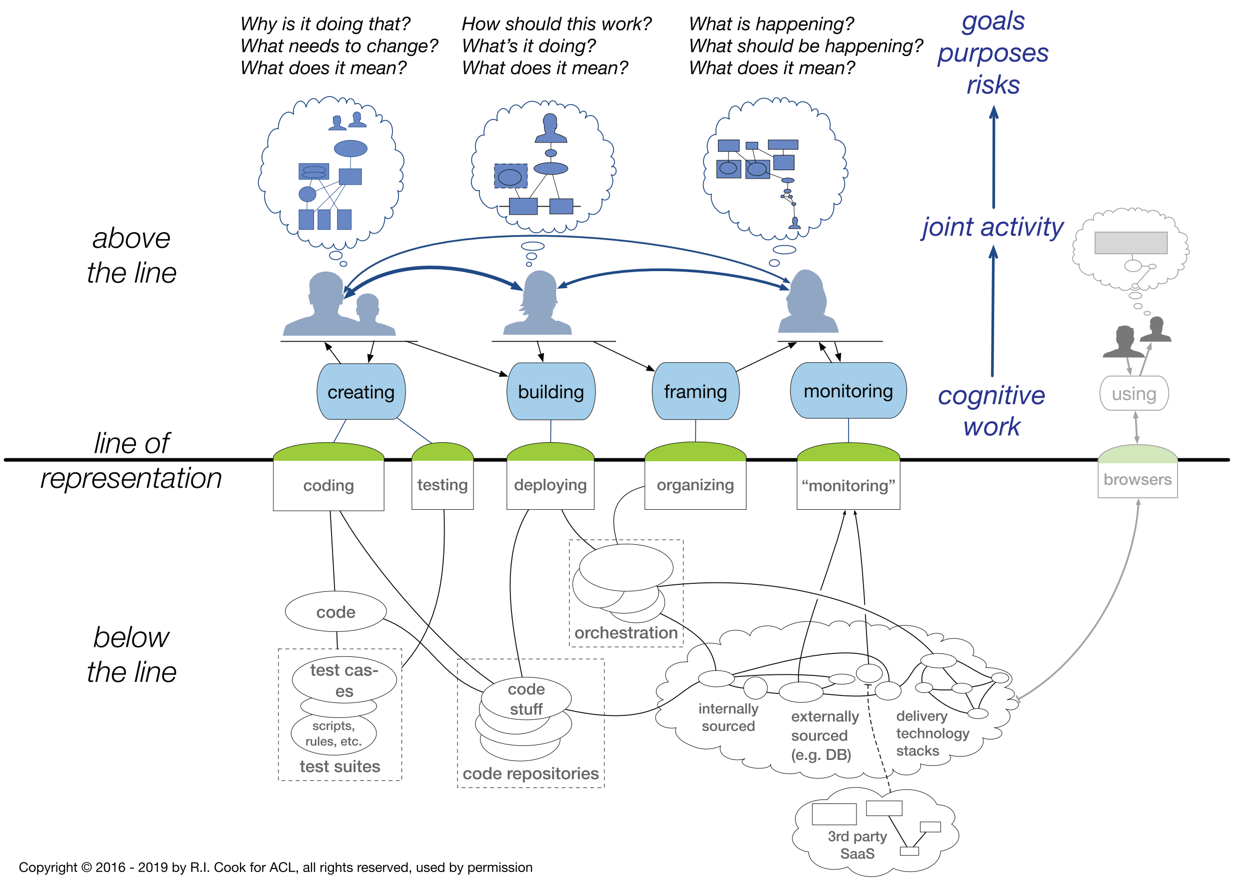
All understandings of what lies below the line are *constructed* in the sense proposed by Bruno Latour and Steve Woolgar.[2] What we "know"—what we *can* know—about what lies below the line depends on inferences made from representations that appear on the screens and displays. These inferences draw on our mental models—those that have been developed and refined over years, then modified, updated, refined, and focused by recent events. Our understandings of how things work, what will happen, what *can* happen, what avenues are open, and where hazards lie are contained in these models.

Implications

It will be immediately apparent that no individual mental model can ever be comprehensive. The scope and rate of change assure that any complete model will be stale and that any fresh model will be incomplete. David Woods said this clearly in what is known as Woods' theorem:[4]

As the complexity of a system increases, the accuracy of any single agent's own

An Internet-facing system.



Copyright © 2016 - 2019 by R.I. Cook for ACL, all rights reserved, used by permission *model of that system decreases rapidly.*

1. *This is a complex system; it is always changing.* The composition and arrangement of the components are such that the system's behavior is nondeterministic. Continuous and often substantial change is going on both above and below the line. There is no way to capture its state nor to reproduce a given state. All models of the system are approximations. It is impossible to anticipate all the ways that it might break down or defend against all eventualities.

The level of complexity below and above the line is similar. As the complexity below the line has increased, so too has the complexity above the line.

2. *Collaboration is necessary; collaboration is routine.* Many episodic activities—especially troubleshooting and repair beyond handling of minor anomalies—cannot be accomplished by a single person and require collaboration. Although occasionally the demands of an event may be well matched to the knowledge and capacity of the first person who encounters it, work on most incidents is likely to require joint action by several (or many) people. These events test the capacity to combine, test, and revise mental models. This can be seen to play out in the incident dialog and the after-incident review.

3. *Coordinating collaborative efforts is challenging.* The job of bringing expertise to bear and coordinating the application of that expertise is nontrivial and often undertaken under severe time and consequence pressure. A burgeoning field of interest is the application of various methods to identify and engage people in problem solving, to generate productive, parallel threads of action, to bring these threads back together, and to evaluate and make decisions. Many organizations are developing support tools, managerial processes, and training to address this need. In particular, controlling the costs of coordination of these parallel and joint activities is a continuing challenge.

For some events troubleshooting and repair are highly localized below and above the line. When there is a one-to-one mapping from a below-the-line component to an above-the-line individual or team, the work of coordination can be small. For other events the troubleshooting and repair can be arduous because the manifestations of the anomaly are far from its sources—so far, in fact, it is unclear whose knowledge could be useful.

These events are often quite different from those in domains where roles and functions are relatively well defined and task assignment is a primary concern. Coordinating collaborative problem solving in critical digital services is the subject of intense investigation and the target of many methods and tools, yet it remains a knotty problem.

4. *Similar faults and failures can occur above and below the line.* The reverberation across the line of representation tends to shape the structure below the line (particularly the functional boundaries) to be like that above the line, and vice versa. Because structure and function above the line parallel structure and function below, parallels can be expected in the forms of dysfunction that can occur. Both are distributed systems. This suggests that specific below the line failure forms (for example, susceptibility to partition, CAP [consistency, availability, partition tolerance], or even the potential for saturation or cascading failure) will also be found in some form above the line.

5. *It's one system, not two.* The line of representation appears to be a convenient boundary separating two "systems," a technical one below the line and a human one above it. Reciprocal cause and effect above and below make that view untenable. People are constantly interacting with technologies below the line; they build, modify, direct, and respond to them. But these technologies affect those people in myriad ways, and experience with the technologies produces changes above the line. These interactions weld what is above the line to what is below it. There are not two systems separated by a representational barrier; there is only one system.

A similar argument developed around human-computer interaction in the 1970s. Efforts to treat the computer and the human operator as separate and independent entities broke down and were replaced by a description of human *and* computer as a "system." Large-scale distributed computing and the similarly distributed approaches to programming and operations are replicating this experience on a larger scale.

Incidents Occur *Above* the Line

Incidents are a "set of activities, bounded in time, that are related to an undesirable system behavior."[1] The decision to describe some set of activities as an incident is a judgment made by people above the line. Thus, an incident begins when someone says that it has begun and ends when someone says it has ended. Like the understanding of what lies below the line, incidents are *constructed*.

Conclusion

Knowledge and understanding of below-the-line structure and function are continuously in flux. Near-constant effort is required to calibrate and refresh the understanding of the workings, dependencies, limitations, and capabilities of what is present there. In this dynamic situation no individual or group can ever know the system state. Instead, individuals and groups must be content with partial, fragmented mental models that require more or less constant updating and adjustment if they are to be useful. C

Related articles on queue.acm.org

Continuous Delivery Sounds Great, but Will It Work Here?
Jez Humble
https://queue.acm.org/detail.cfm?id=3190610

A Decade of OS Access-control Extensibility
Robert N.M. Watson
https://queue.acm.org/detail.cfm?id=2430732

The Network's NEW Role
Taf Anthias and Krishna Sankar
https://queue.acm.org/detail.cfm?id=1142069

References

1. Allspaw, J., Cook, R.I. SRE cognitive work. *Seeking SRE: Conversations About Running Production Systems at Scale.* D. Blank-Edelman, ed. O'Reilly Media, 2018, 441–465.
2. Latour, B., Woolgar, S. *Laboratory Life: The Construction of Scientific Facts.* Sage Publications, Beverly Hills, CA, 1979.
3. Lave, J., Wenger, E. *Situated Learning: Legitimate Peripheral Participation.* Cambridge University Press, Cambridge,U.K., 1991.
4. Woods, D.D. *Stella: Report from the SNAFUcatchers Workshop on Coping with Complexity.* The Ohio State University, 2017; https://snafucatchers.github.io/.

Richard I. Cook, M.D., is an expert in safety, accidents, and resilience in complex systems. His 30 years of experience includes work in medicine, transportation, manufacturing, and information technology. He is the author of the oft-cited "How Complex Systems Fail" and "Going Solid: A Model of System Dynamics and Consequences for Patient Safety."

DOI:10.1145/3339904

A platform for creating a crowdsourced picture of human opinions on how machines should handle moral dilemmas.

BY EDMOND AWAD, SOHAN DSOUZA, JEAN-FRANÇOIS BONNEFON, AZIM SHARIFF, AND IYAD RAHWAN

Crowdsourcing Moral Machines

ROBOTS AND OTHER artificial intelligence (AI) systems are transitioning from performing well-defined tasks in closed environments to becoming significant physical actors in the real world. No longer confined within the walls of factories, robots will permeate the urban environment, moving people and goods around, and performing tasks alongside humans. Perhaps the most striking example of this transition is the imminent rise of automated vehicles (AVs). AVs promise numerous social and economic advantages. They are expected to increase the efficiency of transportation, and free up millions of person-hours of productivity. Even more importantly, they promise to drastically reduce the number of deaths and injuries from traffic accidents.[12,30] Indeed, AVs are arguably

» key insights

- Machines are assuming new roles in which they will make autonomous decisions that influence our lives. In order to avoid societal pushback that would slow the adoption of beneficial technologies, we must sort out the ethics of these decisions.
- Behavioral surveys and experiments can play an important role in identifying citizens' expectations about the ethics of machines, but they raise numerous concerns that we illustrate with the ethics of driverless cars and the Moral Machine experiment.
- Data collected shows discrepancies between the preferences of the public, the experts, and citizens of different countries—calling for an interdisciplinary framework for the regulation of moral machines.

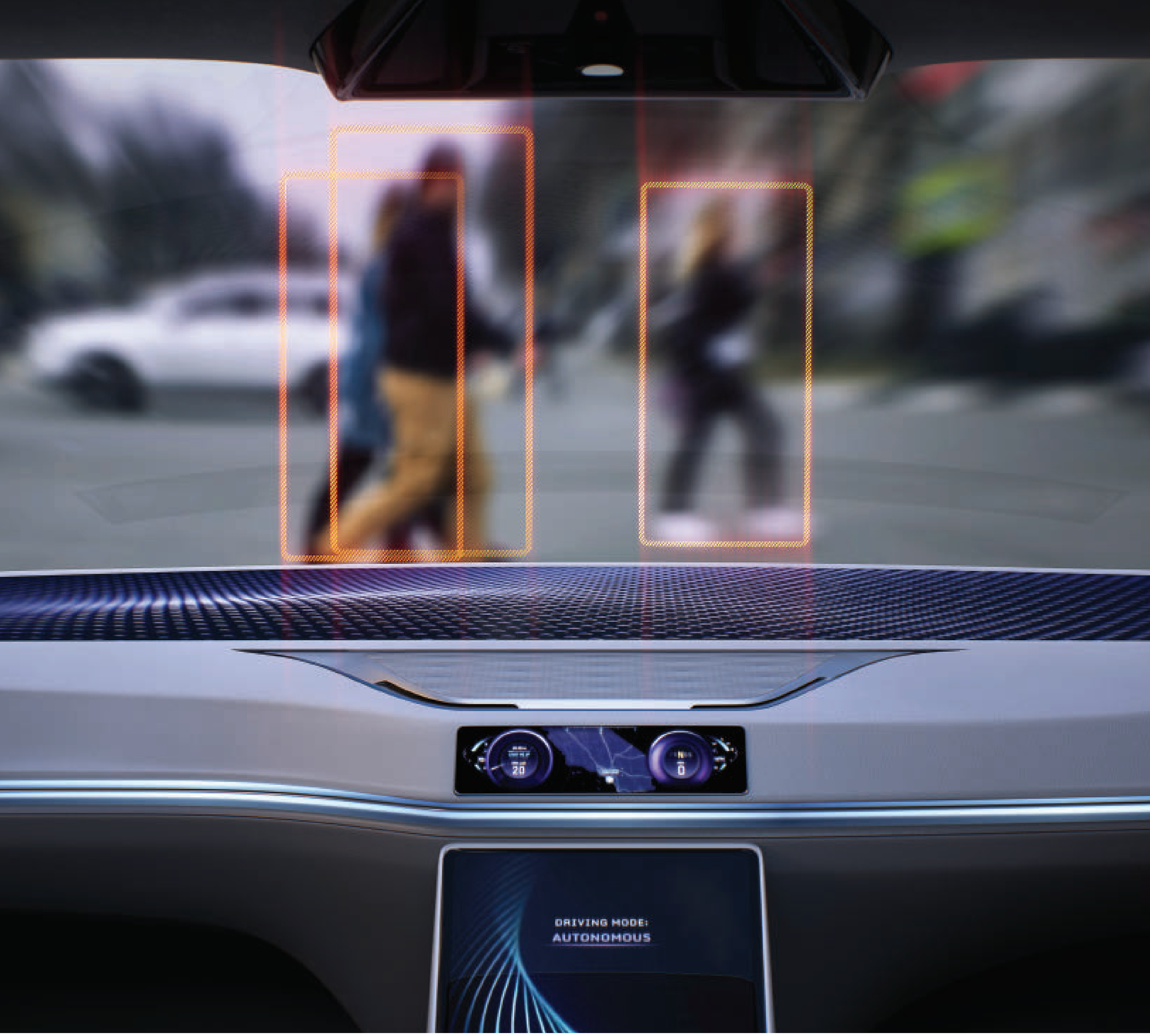


ILLUSTRATION BY KOLLECTED STUDIO

the first human-made artifact to make autonomous decisions with potential life-and-death consequences on a broad scale. This marks a qualitative shift in the consequences of design choices made by engineers.

The decisions of AVs will generate indirect negative consequences, such as consequences affecting the physical integrity of third parties not involved in their adoption—for example, AVs may prioritize the safety of their passengers over that of pedestrians. Such negative consequences can have a large impact on overall well-being and economic growth. While indirect negative consequences are typically curbed by centralized regulations and policies, this strategy will be challenging in the case of intelligent machines.

First, intelligent machines are often black boxes:[24] it can be unclear how exactly they process their input to arrive at a decision, even to those who actually programmed them in the first place.

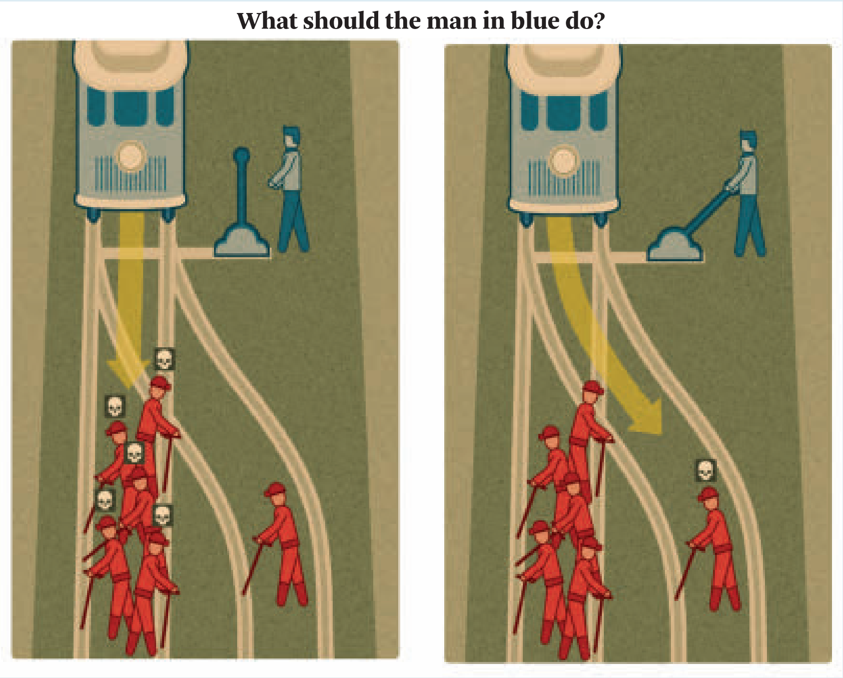
Second, intelligent machines may be constantly learning and changing their perceptual capabilities or decision processes, outpacing human efforts at defining and regulating their negative externalities. Third, even when an intelligent machine is shown to have made biased decisions,[27] it can be unclear whether the bias is due to its decision process or learned from the human behavior it has been trained on or interacted with.

All these factors make it especially challenging to regulate the negative externalities created by intelligent machines, and to turn them into moral machines. And if the ethics of machine behavior are not sorted out soon, it is likely that societal push-back will drastically slow down the adoption of intelligent machines—even when, like in the case of AVs, these machines promise widespread benefits.

Sorting out the ethics of intelligent machines will require a joint effort of engineers, who build the machines, and humanities scholars, who theorize about human values. The problem, though, is that these two communities

Figure 1. A visual depiction of the classic Trolley Problem as displayed in the Moral Machine interface.

A man in blue is standing by the railroad tracks when he notices an empty trolley rolling out of control. It is moving so fast that anyone it hits will die. Ahead on the main track are five people. There is one person standing on a side track that does not rejoin the main track. If the man in blue does nothing, the trolley will hit the five people on the main track, but not the one person on the side track. If the man in blue flips a switch next to him, it will divert the trolley to the side track where it will hit the one person, and not hit the five people on the main track. What should the man in blue do?



Common criticisms and responses regarding the crowdsourcing of AV ethics using the Trolley Problem method.

Too Naïve	Laypersons' responses to public polls can be biased or ill-informed. Ethical trade-offs must be solved by policy experts, not majority voting.	Policymakers must know about the values most important to the public, so they can either accommodate these values, or anticipate frictions that need be explained.
Too Simple	Real accidents do not involve only two possible actions, and these actions do not have deterministic outcomes.	Highly complex scenarios would only allow for highly specific conclusions. Simplified scenarios zero in on the general principles that guide citizens' ethical intuitions.
Too Improbable	AV-Trolleys are based on very implausible sets of assumptions, and their actual probability of occurrence is too small to deserve attention.	Edge cases can have a massive impact on public opinion, and AV-Trolleys are the discrete form of a very real statistical problem.
Too Early	AV-Trolleys regulations should be avoided at this early technological stage, because their consequences are hard to predict.	Even though it may be too early to regulate about AV-Trolleys, it is the right time to start crowdsourcing citizen preferences.
Too Disconnected	Stated preferences are too disconnected from real actions	The behavior of human drivers is irrelevant to the proposed crowdsourcing task.
Too Distracting	Car makers should focus on making AVs safer, instead of wasting time and resources on crowdsourcing ethical dilemmas.	True, and this is why we need computational social scientist to handle that task.
Too Scary	Overexposing people to AV-Trolleys may scare them away, and be detrimental for their trust in the technology.	This is an empirical question, and our surveys did not find any evidence for such an adverse effect.

are not used to talking to each other. Ethicists, legal scholars, and moral philosophers are well trained in diagnosing moral hazards and identifying violations of laws and norms, but they are typically not trained to frame their recommendations in a programmable way. In parallel, engineers are not always capable of communicating the expected behaviors of their systems in a language that ethicists and legal theorists use and understand. Another example is that while many ethicists may focus more on the normative aspect of moral decisions (that is, what we should do), most companies and their engineers may care more about the actual consumer behavior (what we actually do). These contrasting skills and priorities of the two communities make it difficult to establish a moral code for machines.

We believe that social scientists, and computational social scientists have a pivotal role to play as intermediaries between engineers and humanities scholars, in order to help them articulate the ethical principles and priorities that society wishes to embed into intelligent machines. This enterprise will require elicitation of social expectations and preferences with respect to machine-made decisions in high-stakes domains; to articulate these expectations and preferences in an operationalizable language; and to characterize quantitative methods that can help to communicate the ethical behavior of machines in an understandable way, in order for citizens—or regulatory agencies acting on their behalf—to examine this behavior against their ethical preferences. This process, which we call 'Society in The Loop' (SITL),[25] will have to be iterative, and it may be painfully slow, but it will be necessary for reaching a dynamic consensus on the ethics of intelligent machines as their scope of usage and capabilities expands.

This article aims to provide a compelling case to the computer science (CS) community to pay more attention to the ethics of AVs, an interdisciplinary topic that includes the use of CS tools (crowdsourcing) to approach a societal issue that relates to CS (AVs). In so doing, we discuss the role of psychological experiments in informing the engineering and regulation of AVs,[4,21] and we respond to major objections to both the Trolley Problem and crowdsourcing ethical opinions about that dilemma. We also describe our experience in building a public engagement tool called the Moral Machine, which asks people to make decisions about how an AV should behave in dramatic situations. This tool promoted public discussion about the moral values expected of AVs and allowed us to collect some 40 million decisions that provided a snapshot of current preferences about these values over the entire world.[1]

The Problem with the Trolley Problem

Today, more than ever, computer scientists and engineers find themselves in a position where their work is having major societal consequences.[10,23] As a result, there is increasing pressure on computer scientists to be familiar with the humanities and social sciences in order to realize the potential consequences of their work on various stakeholders, to get training in ethics,[16] and to provide normative statements on how their machines should resolve moral trade-offs. These are new missions for computer scientists, for which they did not always receive relevant training, and this pressure can sometimes result in frustration, instead of leading to the intended ideal outcomes.

The Trolley Problem provides a striking example of the contrast between what computer scientists are trained to do and what they are suddenly expected to do. Scientists working on AVs are constantly asked about their solution to the Trolley Problem, an infamous philosophical dilemma[a] illustrated in Figure 1. At first glance, the Trolley Problem seems completely irrelevant to CS. Its 21st-century version, however, goes like this: An AV with a brake failure is about to run over five pedestrians crossing the street. The only way out is to swerve to one side, crashing into a barrier and killing its sole passenger. What should the AV do?[13,18] What if there are three passengers in the car? What if two of these passengers are children?

We believe that social scientists and computational social scientists have a pivotal role to play as intermediaries between engineers and humanities scholars in order to help them articulate the ethical principles and priorities that society wishes to embed into intelligent machines.

The AV version of the Trolley Problem (AV-Trolley, henceforth) has become so popular that computer scientists, engineers, and roboticists are endlessly asked about it, even when their work has nothing to do with it. It has become the poster child in debates about the ethics of AI, among AV enthusiasts, technologists, moral psychologists, philosophers, and policymakers.[3,18,22] Whether or not this prominence is deserved, the AV-Trolley is everywhere, and it is worth looking in detail at the arguments that have been made for (but mainly against) its relevance for the field of AVs, and for the importance of polling citizens about the solutions they might find acceptable (see the accompany table for a summary).

The citizens are too naïve. First, one may question the usefulness of seeking input from lay citizens when dealing with such complex issues as AV ethics. Certainly, using a simple thought experiment such as the AV-Trolley makes it possible to poll citizens about their preferences. But what are we to do with their responses? Is it not dangerous, or even irresponsible, to seek the opinions of naïve citizens whose responses may be biased or ill-informed? We very much agree that regulations of ethical trade-offs should be left to policy experts, rather than resolved by referendum. But we also believe that policy experts will best serve the public interest when they are well informed about citizens' preferences, regardless of whether they ultimately decide to accommodate these preferences.[2] Sometimes, when policy experts cannot reach a consensus, they may use citizens' preferences as a tie-breaker. Other times, when policy experts find citizens' preferences problematic, and decide not to follow them, they must be prepared for the friction their policies will create and think carefully about how they will justify their choices in the public eye. Whether policy experts decide to take a step toward the preferences of citizens, or to explain why they took a step away, they need to know about the preferences of citizens in the first place.

[a] The Trolley Problem, together with all its variants,[11,28,29] is ubiquitous in studies of law and ethics. It was traditionally used to test ethical principles against moral intuitions. More recently, the Trolley Problem has been used extensively in moral psychology and neuroscience to explore not how humans should make ethical decisions, but how they actually do so. This literature delivered deep insights into moral cognition, as well as about the contextual factors that influence moral judgment.[8,9,15]

The scenarios are too simple. Is the AV-Trolley too simplistic to be valuable? Real accidents do not involve only two possible actions, and these actions do not have deterministic outcomes. AVs will have many options beyond staying or swerving, and it is not clear they will be able to precalculate the consequences of all these actions with enough certainty. Many factors that would be relevant for real accidents are simply absent in an AV-Trolley scenario. Note, however, that AV-Trolleys are meant to be abstract and simplified, in order to cleanly capture basic preferences. Using realistic crash scenarios would make it difficult to tease out the effect of multiple contributing factors and make it difficult to draw general conclusions beyond the highly specific set of circumstances that they feature. The AV-Trolley can be used to conduct simplified controlled experiments, in which respondents are randomly assigned to different conditions (accident scenarios), in which the scenarios are simpler than what they would be in the real world, and in which everything is kept constant but for the variables of interest.

AVs will have many options beyond staying or swerving, and it is not clear they will be able to precalculate the consequences of all these actions with enough certainty.

The scenarios are too improbable. AV-Trolleys are based on a series of assumptions that are extremely improbable. For example, respondents must accept the very unlikely premises that the AV is driving at an unsafe speed in view of a pedestrian crossing, that its brakes are failing, that there is no other way for it to stop, and that the pedestrians just stay there paralyzed. This combination of unlikely assumptions means the probability of an AV-Trolley actually happening is perhaps too small to deserve so much attention. Or is it? Philosopher Patrick Lin has laid down forceful arguments for the relevance of the AV-Trolley, despite its tiny probability of occurrence.[19] Even if we accept that AV-Trolley scenarios are extremely rare, their consequences may be extremely powerful. The few AV crashes that took place so far received massive coverage in the media, way beyond the coverage of all crashes happening the same year, and way beyond the positive coverage of progress in the performance of AVs. Similarly, a single occurrence of a real AV-Trolley crash could have massive impact on the public trust in AVs. Such a low-probability, high-risk event is known as an edge case, and handling edge cases is important for the design of any product. Finally, even if AV-Trolley crashes are very rare, they can help to think about their statistical extension, the *statistical trolley problem*.[5,14,19] In its discrete version, the AV-Trolley asks about a black-and-white, all-or-none situation where people choose who should live and who should certainly die. The statistical trolley problem ultimately involves the same trade-offs, but ones that occur only when billions of decisions about how minor risks should be allocated are aggregated over millions of miles driven. Imagine an AV driving in a middle lane between a truck and a cyclist. Depending on how much of a berth the AV gives either the truck or the cyclist, its behavior results in a shift of risk between itself, the truck, and the cyclist. This creates the problem of deciding which risk transfers are fair or acceptable. Suppose that conventional cars kill 100 people (80 passengers and 20 cyclists). Program A kills only 20 people (15 passengers and five cyclists), and so does Program B (one passenger, 19 cyclists). What would be the morally preferable program? Should 15 passengers die for five cyclists, or should one passenger die for 19 cyclists? This statistical trolley problem is very real, but much more complex than its discrete version. Data collected with the discrete version of the AV-Trolley do not solve its statistical version but provide a useful starting point for experimental investigations of this statistical version.

Stated preferences are too disconnected from real actions. The idea of "crowdsourcing preferences" assumes that stated preferences provide useful evidence about what respondents would actually do when faced with a physical situation with real life-or-death consequences. But previous work has showed that people's stated preferences and their actual actions diverge in many contexts. In this case, studies that put subjects in simulators and prompt them to react, would provide a better measure of the actual preferences of respondents. While we agree with this assessment, we note that the behavior of human drivers is irrelevant to the proposed crowdsourcing task.

The goal of the crowdsourcing task here is not to capture the actual actions, but to capture what humans would believe (from the comfort of an armchair) to be the best course of action. We can certainly do better with AVs than just imitating the reflexes of a stressed human driver in a split-second crash. Since cars can be programmed and humans cannot, cars can be programmed to do what humans would like to do, rather than what humans would actually decide, on impulse, in a split-second car crash.

It is too early to regulate. Even if AV-Trolley crashes may have major consequences for public trust, they still belong to a rather distant future. They involve highly automated, fully autonomous cars that may not be available for a while, whose behavior on the road is still unknown, and whose technology has not matured. For all these reasons, it may be too early to design regulations for AV-Trolleys. This point relates to the "Collingridge dilemma,"[7] which states that with every new technology, there are two competing concerns. On one hand, regulations are difficult to develop at an early technological stage because their consequences are difficult to predict. On the other hand, if regulations are postponed until the technology is widely used, then the recommendations come too late. In the case of AV-Trolleys, it would seem the ethical debate started well before the technology would be actually available, which means it might be premature to regulate just now. However, it is not too early to inform future regulators about the preferences of citizens. Perhaps right now is not the time to establish rules—but it is the right time to start crowdsourcing preferences, especially when this crowdsourcing effort might take several years.

The debate is too distracting. Car makers are in the business of making safe cars, not in the business of solving age-old ethical dilemmas. By burdening them with the AV-Trolley, the criticism goes, we distract them from their real mission, which is to maximize the safety of AVs, and bring them to the public as soon as possible. This will be better achieved by directing their resources to safety engineering, than to philosophical musing or moral psychology. This is absolutely true, and this is why we believe that computational scientists have a critical role to play in crowdsourcing machine ethics, and in translating their results in a way that is useful to ethicists, policymakers, and the car industry. The burden must be shared, and computational social scientists are best equipped to handle this crowdsourcing of ethics. Not to mention it is highly implausible the car industry would ever compromise car safety in order to invest in philosophy.

The crowdsourcing is too scary. One main objective of crowdsourcing the ethics of AVs is to find the best possible alignment between regulations and citizen preferences—and a major reason for doing so is to improve trust and social acceptance of AV technology. But crowdsourcing AV ethics using AV-Trolleys could be counterproductive in that respect, since it focuses the attention of the public on scary, improbable edge cases. This is a serious concern, but also an empirical question: Is it true that exposure to AV-Trolleys adversely affects public trust, excitement, or general attitude toward AVs? Our team tested this possibility with both a correlational approach (measuring the link between prior exposure to AV-Trolleys and attitude toward AVs) and a causal approach (measuring the effect of a very first exposure to AV-Trolleys) and found no statistical evidence for any adverse effect of the exposure to AV-Trolleys.[6] People may not like some specific solutions to AV-Trolleys, but they do not react negatively to the problem itself.

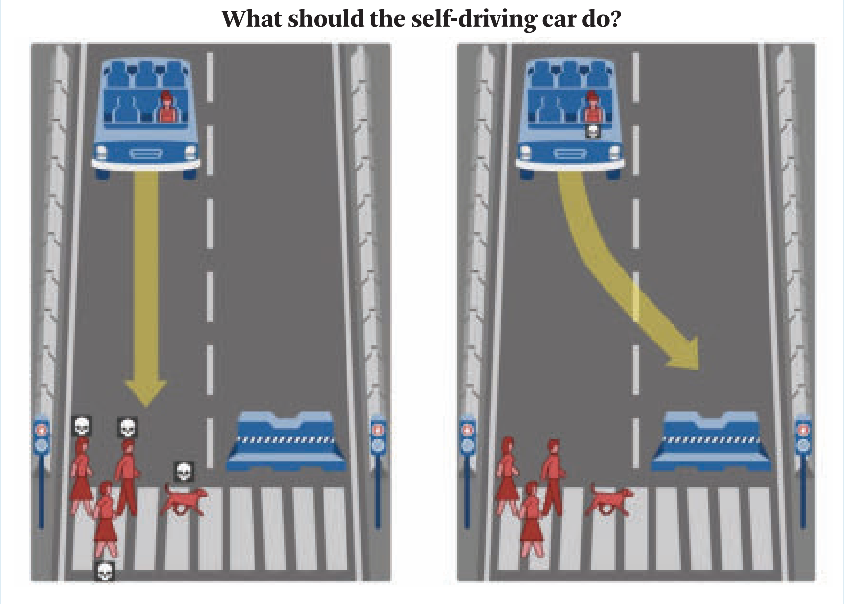
Moral Machine

Having made it clear our support of the use of AV-Trolleys for crowdsourcing the ethics of automated vehicles, the reason for this support, and the limitations of this crowdsourcing exercise, we now describe the platform we created for this purpose, and the data it allowed us to collect. In June 2016, we deployed Moral Machine (MM), a platform for gathering data on human perception of the moral acceptability of decisions made by AVs faced with choosing which humans to harm and which to save. MM fits the specifications of a massive online experimentation tool, given its scalability, accessibility to the online community, and the random assignment of users to conditions. Another purpose to the platform is the facilitation of public feedback, discussion of scenarios and acceptable outcomes, and especially public discussion of the moral questions relevant to self-driving vehicles, which was previously scarce.

The central data-gathering feature is the Judge mode, illustrated in Figure 2.

Figure 2. Moral Machine-Judge interface.

A pictorial representation of a dilemma faced by an AV. If the AV continues ahead it will hit and kill a group of pedestrians, including three adults and a dog, crossing on a red light. If the AV swerves, it will hit a barrier and result in the death of its sole passenger, a female athlete.



In this mode, users are presented with a series of 13 moral dilemma scenarios, each with two possible outcomes. The MM restricted scenarios to just two outcomes and did not, for example, offer the solution to drive more slowly and stop safely. This was done on purpose, to ensure participants would have to face difficult ethical decisions, without being able to select a completely satisfying resolution. While this methodological choice was justified in the specific context of the MM project, safe driving and appropriate speed do constitute critically important issues for the broader debate about the ethics of AVs.

The scenarios are generated using randomization under constraints, chosen so that each scenario tests specifically for a response along one of six dimensions (age, gender, fitness, social status, number, and species). Each user is presented with two randomly sampled scenarios of each of the six dimensions, in addition to one completely random scenario (that can have any number of characters on each side, and in any combination of characters). These together make the 13 scenarios per session. The order of the 13 scenarios is also counterbalanced over sessions. In addition to the six dimensions, three other dimensions (interventionism, relation to AV, and legality) are randomly sampled in conjunction with every scenario of the six dimensions. Each of the 13 scenarios features combinations of characters from a list of 20 different characters.

Figure 3. The ranking of countries according to the average preference to spare the lawful (pedestrians crossing at the "walk" signal, instead of the "wait" signal).

All countries show preference for sparing the lawful at the cost of the unlawful. The top five countries in terms of readiness index[17] are highlighted in red, and they fall on different sides of the world average for sparing the lawful.

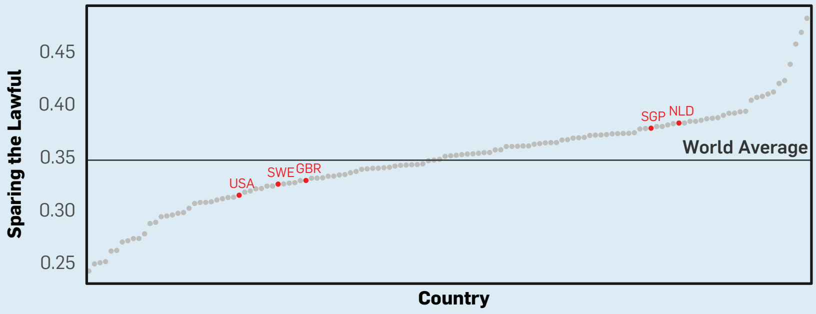
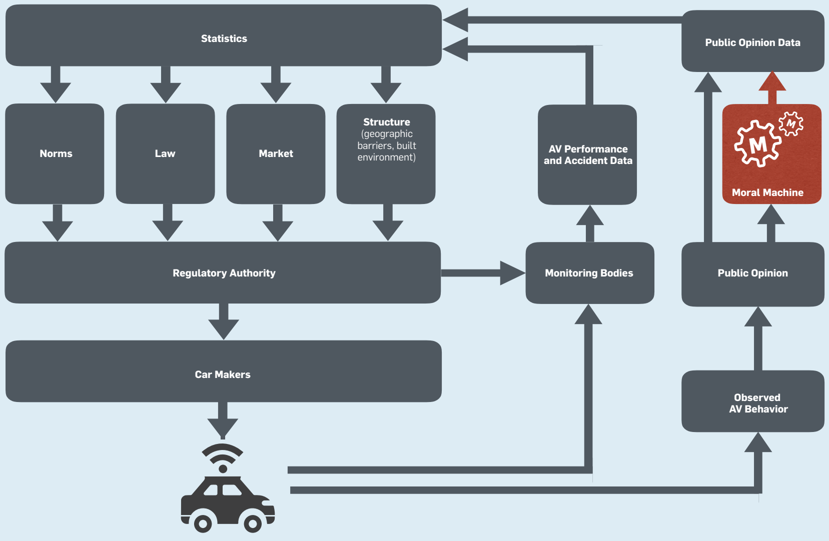


Figure 4. A society-in-the-loop framework for AV regulation.

The model does not represent an actual regulatory system, but it clarifies how a crowdsourcing platform like the Moral Machine fits into the broader regulatory system by providing data on societal norms.



Upon deployment in 2016, the MM website got covered in various media outlets and went viral beyond all expectations. Accordingly, the website's publicity has allowed us to collect the largest dataset on AI ethics ever (40 million decisions by millions of visitors from 233 countries and territories to date).

The results drawn from the data collected through MM were published two years ago.[1] The study reports two main findings: First, among the nine tested attributes, three attributes received considerably higher approval rate than the rest. These are the preference to spare humans over pets, the preference to spare more characters over fewer characters, and the preference to spare the younger humans over the older humans.

Second, while responses from most countries agree on the directions of the preferences, the magnitude of these preferences are considerably different. And countries' aggregate responses broadly cluster into three main clusters: Western (including a majority of English-speaking, Catholic, Orthodox, and Protestant countries), Eastern (including a majority of Islamic, Confucian, and South Asian countries, and Southern (comprising Latin America and former French colonies). The findings also presented predictive factors of country-level differences. One example is the strength of rule of law in a country being correlated with a stronger preference to spare the lawful.

Providing a full discussion about the policy implications of these findings is beyond the scope of this article. However, we note here a summary of the implications. In 2016, Germany became the first country to draft regulations for AVs. The country formed a committee of experts to draft ethical guidelines for automated vehicles.[20] Comparing the preferences we collected via MM to the German commission report, we notice that while there is some overlap between the opinions of the public and the experts (for example, both agree on sacrificing animals in order to spare human life), there are also key points of disagreement (for example, while the public largely approves of sparing children at the cost of the elderly, the experts prohibit any discrimination based on age). While the experts are not required to cater to the public's preferences when making ethical decisions, they may be interested in knowing the views of the public, especially in cases where the right decision is difficult to discern, and where it may be important to gauge and anticipate public reaction to important decisions.

Clearly, this was the case for Germany. What would be the case for other countries? To date, Germany remains the only country with any guidelines for AVs. Once other countries form their own guidelines, they may end up being similar or different. This leads to our second main finding: Programming ethical decisions in AVs using the same rules is likely to get different levels of push-back in different countries. For example, if AVs are programmed in a way that disadvantages jaywalkers, such AVs may be judged more acceptable in some countries (where the rule of law is stronger) than in others.

The possibility of seeing this happening might manifest itself sooner than we expect. A recent article by KPMG reported on the top countries in terms of readiness for AVs.[17] According to the report, the readiest five countries are the Netherlands, Singapore, the U.S., Sweden, and the U.K. Figure 3 shows that even these top five countries have some disagreement over the magnitude of preference for sparing the lawful. This could mean that a rule such as programming AVs to increase safety for law-abiding citizens at the cost of jaywalkers, while expected to gain high acceptability in the Netherlands and Singapore, may stir anger in the U.S., Sweden, and the U.K.

A Regulatory Framework

As we argued at the beginning of this article, we believe bringing about accountable intelligent machines that embody human ethics requires an interdisciplinary approach. First, engineers build and refine intelligent machines, and tell us how they are capable of operating. Second, scholars from the humanities—philosophers, lawyers, social theorists—propose how machines ought to behave, and identify hidden moral hazards in the system. Third, behavioral scientists, armed with tools for public engagement and data collection like the MM, provide a quantitative picture of the public's trust in intelligent machines, and of their expectations of how they should behave.[b] Finally, regulators monitor and quantify the performance of machines in the real world, making this data available to engineers and citizens, while using their enforcement tools to adjust the incentives of engineers and corporations building the machines.

We summarize this regulatory architecture in Figure 4, clarifying where crowdsourcing tools can be useful. The Moral Machine project serves as an example of a tool that empowers the public engagement component of our approach to putting 'society in the loop.' It exemplifies interdisciplinary collaboration that combines tools from philosophy, psychology, humanities, computer science and statistics, to inform our quest for a world teeming with increasingly intelligent and autonomous machines that nevertheless behave in line with human values.

[b] We note here that in order to keep the project tractable, the MM experiment had to constrain the possible responses that participants can provide. This is precisely why we do not believe the MM responses are sufficient, on their own, to inform the programming of automated vehicles, which should take into account a variety of perspectives, and the real-world complexity of actual dilemmas of risk distribution. Recent work by Sütfeld et al.[26] suggests the MM results do generalize to different ways of presenting the stimulus, but more work remains to be done on this problem to test the external validity of the findings.

References

1. Awad, E., Dsouza, S., Kim, R., Schulz, J., Henrich, J., Shariff, A., Bonnefon, J., and Rahwan, I. The Moral Machine experiment. *Nature 563*, 7729 (2018), 59.
2. Awad, E. and Levine, S. We Should Crowdsource Ethics. In press.
3. Bogost, I. *Enough with the Trolley Problem.* The Atlantic (2018).
4. Bonnefon, J., Shariff, A., and Rahwan, I. The social dilemma of autonomous vehicles. *Science 352*, 6293 (2016), 1573–1576; http://bit.ly/2NyQyUa
5. Bonnefon, J., Shariff, A., and Rahwan, I. The trolley, the bull bar, and why engineers should care about the ethics of autonomous cars. In *Proceedings of IEEE 107*, 3 (2019), 502–504.
6. Bonnefon, J., Shariff, A., and Rahwan, I. The moral psychology of AI and the ethical opt-out problem. *The Ethics of Artificial Intelligence.* S.M. Liao, ed. Oxford University Press, Oxford, U.K., in press.
7. Collingridge, D. The social control of technology. (1982).
8. Cushman, F. and Young, L. Patterns of moral judgment derive from nonmoral psychological representations. *Cognitive Science 35*, 6 (2011), 1052–1075.
9. Edmonds, D. *Would You Kill the Fat Man?: The Trolley Problem and What Your Answer Tells Us About Right and Wrong.* Princeton University Press, 2013.
10. Eubanks, V. *Automating Inequality: How High-Tech Tools Profile, Police, and Punish the Poor.* St. Martin's Press, 2018.
11. Foot, P. The problem of abortion and the doctrine of double effect. (1967).
12. Gao, P., Hensley, R., and Zielke, A. A road map to the future for the auto industry. *McKinsey Quarterly*, (Oct. 2014).
13. Goodall, N. Ethical decision making during automated vehicle crashes. *Transportation Research Record: J. Transportation Research Board* 2424 (2014), 58–65.
14. Goodall, N.J. Away from trolley problems and toward risk management. *Applied Artificial Intelligence 30*, 8 (2016), 810–821.
15. Greene, J.D., Sommerville, R.B., Nystrom, L.E., Darley, J.M., and Cohen, J.M. An fMRI investigation of emotional engagement in moral judgment. *Science 293*, 5537 (2001), 2105–2108.
16. Huff, C. and Furchert, A. Toward a pedagogy of ethical practice. *Commun. ACM 57*, 7 (July 2014), 25–27.
17. KPMG International. Autonomous Vehicles Readiness Index: Assessing countries openness and preparedness for autonomous vehicles; https://assets.kpmg/content/dam/kpmg/xx/pdf/2018/01/avri.pdf
18. Lin, P. The ethics of autonomous cars. *The Atlantic* (2013).
19. Lin, P. Robot cars and fake ethical dilemmas. *Forbes* (2017).
20. Luetge, C. The German ethics code for automated and connected driving. *Philosophy & Technology 30*, 4 (2017), 547–558.
21. Malle, B.F., Scheutz, M., Arnold, T., Voiklis, J., and Cusimano, C. Sacrifice one for the good of many? People apply different moral norms to human and robot agents. In *Proceedings of the 10th annual ACM/IEEE Iintern. Conf. Human-Robot Interaction.* ACM, 2015, 117–124.
22. Marshall, A. Lawyers, not ethicists, will solve the robocar 'Trolley Problem.' *WIRED* (May 28, 2017).
23. O'Neil, C. *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy.* Broadway Books, 2016.
24. Pasquale, F. *The Black Box Society: The Secret Algorithms that Control Money and Information.* Harvard University Press, 2015
25. Rahwan, I. Society-in-the-loop: Programming the algorithmic social contract. *Ethics and Information Technology 20*, 1 (2018), 5–14.
26. Sütfeld, L.R., Ehinger, B.V., König, P., and Pipa, G. How does the method change what we measure? Comparing virtual reality and text-based surveys for the assessment of moral decisions in traffic dilemmas. (2019).
27. Sweeney, L. Discrimination in online ad delivery. *Queue 11*, 3 (2013), 10.
28. Thomson, J.J. Killing, letting die, and the trolley problem. *The Monist 59*, 2 (1976), 204–217.
29. Thomson, J.J. The trolley problem. *The Yale Law J. 94*, 6 (1985), 1395–1415.
30. Van Arem, B., Driel, C., and Visser, R. The impact of cooperative adaptive cruise control on traffic-flow characteristics. *IEEE Trans. Intelligent Transportation Systems 7*, 4 (2006), 429–436

Edmond Awad (e.awad@exeter.ac.uk) is a lecturer in the Department of Economics at the University of Exeter Business School, Exeter, U.K.

Sohan Dsouza (dsouza@mit.edu) is a research assistant at MIT Media Lab, Cambridge, MA, USA.

Jean-François Bonnefon (jfbonnefon@gmail.com) is a research director at the Toulouse School of Economics (TSM-R), CNRS, Université Toulouse Capitole, Toulouse, France.

Azim Shariff (afshariff@gmail.com) is an associate professor at the University of British Columbia, Vancouver, Canada.

Iyad Rahwan (rahwan@mpib-berlin.mpg.de) is a director of the Center for Humans & Machines, Max-Planck Institute for Human Development, Berlin, Germany, and an associate professor at MIT Media Lab, Cambridge, MA, USA.

Watch the authors discuss this work in the exclusive *Communications* video. https://cacm.acm.org/videos/crowdsourcing-moral-machines

DOI:10.1145/3343146

When the value increases engagement, engagement increases the value.

BY DARJA SMITE, NILS BREDE MOE, MARCIN FLORYAN, GEORGIANA LEVINTA, AND PANAGIOTA CHATZIPETROU

Spotify Guilds

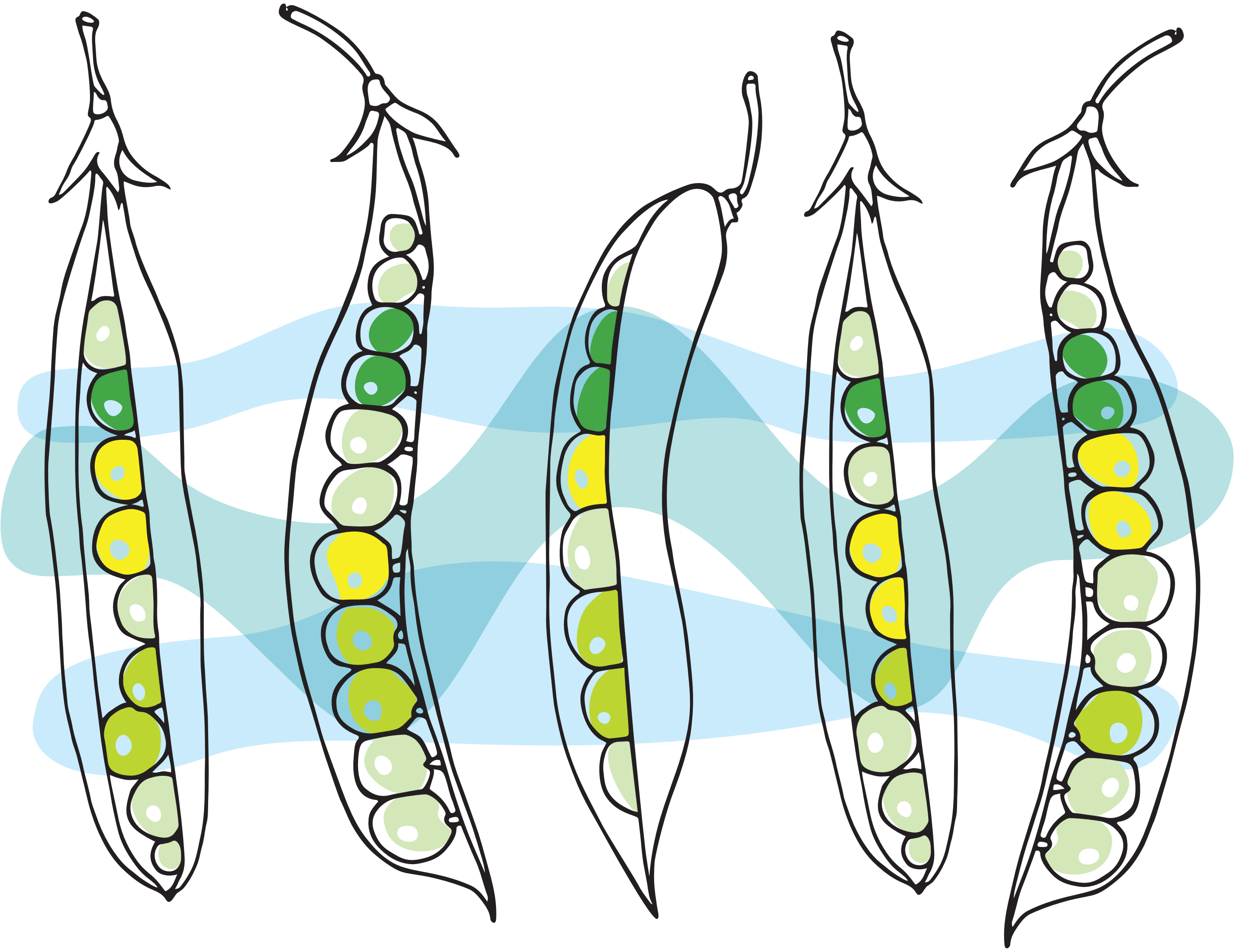
A COMMUNITY OF practice (CoP) is usually a group of people with similar skills and interests who share knowledge, make joint decisions, solve problems together, and improve a practice.[12] Communities of practice are cultivated for their potential to influence the knowledge culture[5–7] and bring value for individuals, teams, projects, and organization as the whole. Knowledge exchange in CoPs is enabled through various forms of scheduled and unscheduled social interaction, such as hallway and water-cooler conversations, meetings and conferences, brown bag lunches, newsletters, teleconferences, shared Web spaces, email lists, discussion forums, and synchronous chats.[6] Activity repertoires in different CoPs may differ significantly.[11]

Despite the assumed benefits, implementing successfully functioning CoPs is a challenge,[12] and even more so in large-scale distributed contexts. Research into CoPs in various disciplines has determined that successful CoPs highly depend on the organizational support on one hand (budget, incentives, awards, resources, and infrastructure[6]) and member engagement and regular interaction on the other.[5,9,12] Furthermore, researchers found a loop between member engagement and value creation—increased engagement helps a community to generate more value, and increased value stimulates more member engagement.[5] While much is known about organic small-scale communities (bottom-up initiatives), achieving member engagement and regular interaction, efficiently sharing knowledge, making joint decisions, and improving a practice collectively across multiple temporary separated locations may introduce significant challenges.

In this article, we report our findings from studying member engagement in large-scale distributed communities of practice at Spotify called guilds. Spotify is an innovative software company providing music streaming services, launched in 2008. It was established as a new generation agile organization with highly autonomous development teams (called squads), and a number of bottom-up coordination mechanisms, including communities of practice (guilds). Guilds at Spotify are designed beyond the formal structures and unite members with shared interests, whether leisure-related (cycling, photogra-

» key insights

- **Company growth threatened the guilds culture at Spotify as the number of engineers grew from a few hundred in one location to several thousand across six geographical locations. Scaling causes detachment, difficulty building a sense of a joint community, and coordination challenges.**
- **Guilds are found to provide more perspectives on problems, coordination and standardization across units, formation of knowledge alliances, forum for accessing and expanding expertise, and a sense of belonging.**
- **To succeed with scaling guilds, we recommend offering diverse value-adding activities and knowledge-sharing channels both regionally and across company locations.**



IMAGERY BY EUGENIA PETROVSKAYA

phy, or coffee drinking) or engineering-related (Web development, backend development, C++ engineering, or agile coaching). In the past 10 years, the company has grown to the size of six research and development offices in three countries and continues to flourish. Practicing C++ engineering, Web development, or any other engineering discipline probably will vary from one location to another, and between engineers with different experience levels. Further, technological and engineering advances might have a limited impact due to increased autonomy and separation of different organizational units. While guilds have successfully addressed the need for sharing knowledge and develop a joint practice when the company was small, there is a need to understand how to scale guilds, the core structures that concern cultivation of a shared practice and joint decisions across autonomous teams, in a way that promotes mutual engagement and collaboration among engineers from different organizational units. (For more information about how the study was conducted, see the sidebar "Overview of the Study.")

Guild Members and Engagement in Guild Activities

Guilds at Spotify are very diverse. There are non-sponsored guilds, such as members that enjoy like-minded activities, and sponsored guilds, such as the four guilds selected for our study—agile coaching, C++ engineering, backend development, and Web development. Sponsored guilds have an explicit sponsor and a budget per member, while the non-sponsored guilds do not receive direct funding. All guilds have open, voluntary membership. The members are commonly the ones representing the practice, for example, 80% of the Agile guild's members are agile coaches. Additionally, each guild has 10%–20% of peripheral members that do not represent the key practitioners but are curious about the practice. Spotify employees are free to join any guild, to follow any or none of the guild activities, and resign at any time, or remain inactive for as long as they wish. Of all Spotify employees, 60% are said to be in some capacity associated with at least one guild.

The four guilds we studied differ in size, offering a repertoire of activities and popularity (see Figure 1). Among the four guilds, only one guild involved members from only one country (C++ engineering guild) but was distributed across several locations within Sweden. Other guilds have members distributed across all Swedish and U.S. locations, and some also involved members from the U.K.

Most of the guilds have regular guild meetings and seminars, yearly unconferences, email groups, and Slack channels for knowledge sharing. Guild meetings serve as the venues for decision-making and exchange of ideas. Seminars are organized for knowledge sharing and learning from internal and

Figure 1. Overview of the guilds, members, repertoire, and engagement.

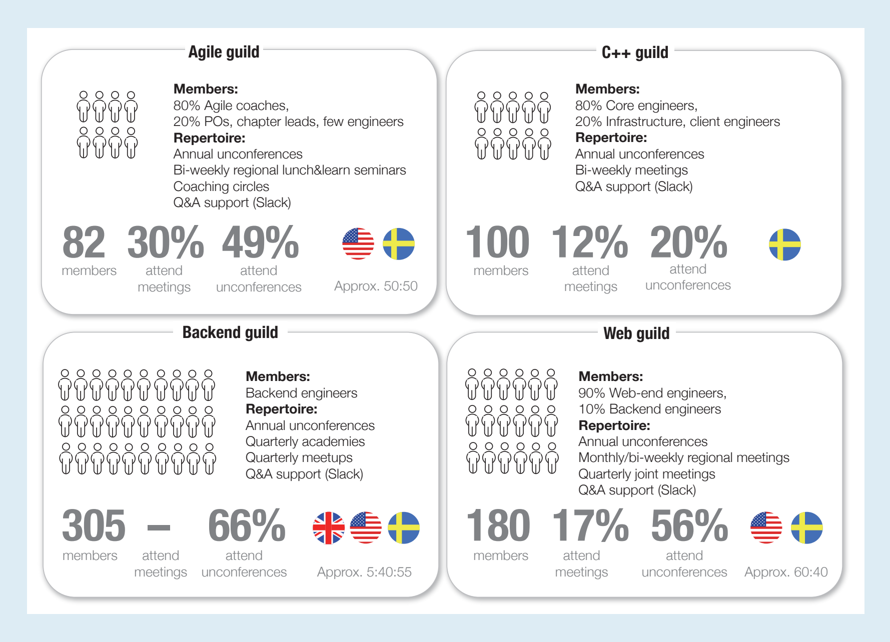
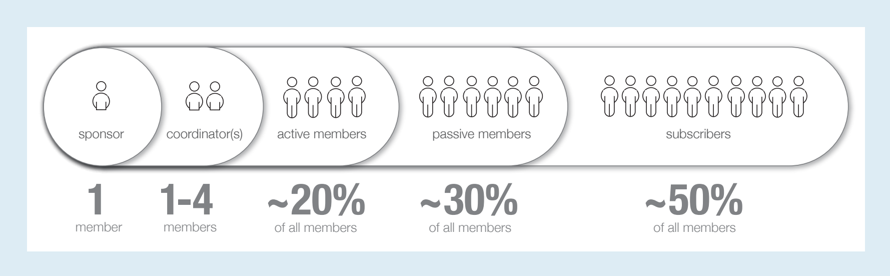


Figure 2. Different types of members.



external experts. To address distribution and inability to meet in person, many of the meetings and seminars are held regionally. This way, the Agile and the Web guilds turned into regionally divided independent sub-guilds, each with local coordinators and activities. Cross-site coordination and knowledge sharing happens primarily in the yearly unconferences, the largest and the most attended events, and in quarterly cross-site meetings, as in the case of the Web sub-guilds.

Participation in different Slack channels and guild activities varies. We detected five different types of members and identified the approximate ratio between the different types based on the numbers of members engaged in different activities and subscribed to different Slack channels, the interviewees' perception, and the characteristics of the survey respondents (see Figure 2). Similarly to Wenger et al.,[12] we identified a group of core members (sponsors and coordinators), active members, and peripheral members (passive members and subscribers). The latter group forms the majority of the community members, as in related studies.[12] Notably, the level of activity of individual members changes over time due to various reasons, such as, the coordinator role rotates, some active members become passive and vice versa, and those who change specialization turn into inactive users who merely subscribe to the latest news.

Perceived Benefits of Guilds

Communities are recognized for the diverse value they bring on different levels. To test the ability of the Spotify guilds to generate value for individual members and the organization as a whole, we asked guild members to select the benefits they believe their guilds create out of the list based on the work by Wenger et al.[12]

Similarly to related research,[5,12] our survey of guild members shows that guilds generate value on both organizational and individual levels (see Figure 3), and that even peripheral members benefit from the guild membership (see Figure 4). The most recognized benefits for Spotify include the ability for guilds to bring more perspectives on problems, facilitate coordination and standardization across units, and form knowledge alliances. For individuals, guilds provide access to expertise and a forum for expanding skills and expertise, a strong sense of belonging, and fun of being with colleagues. Interestingly, while many of the recognized benefits are associated with the potential decrease in unproductive work and time savings, Spotify respondents did not explicitly associate these benefits with operational efficiency that scored high in related studies.[5] This means that true benefits of the guilds are not yet well recognized or understood in the organization.

Interestingly, when analyzing responses from all guilds together, engaged members (sponsors, coordinators, and active members) have reported more benefits on average than the inactive members (passive members and subscribers, as illustrated in Figure 4). The differences in value perception among these groups were found statistically significant in both backend and Web guilds. Our findings therefore support existing research that suggests the association between value and participation.[5]

While guilds are clearly beneficial for their members, one may wonder what the role of such parallel structures is for the teams. Based on the survey results, it is fair to infer that Spotify guilds can be a great support for squads too. Guilds support the onboarding of new engineers minimizing the mentoring effort from colleagues. Guilds help to tackle problems that squads might not be able to solve alone. It also provides a network of experts to whom to turn to when help is needed. Moreover, guilds provide opportunities to network and grow professionally for members of highly cross-functional squads, who do not have local peers with the same competences.

Finally, while our study is not a full replication of a related multi-organizational survey of value creation in four work-based communities,[5] we can still infer that Spotify guilds seem to generate more benefits than reported by the respondents in the related study (the highest score on an individual benefit was 65%, with an average of 54%, and the highest score on an organizational benefit was 57%, with an average of 44%).

Barriers to Mutual Engagement When Scaling

We found the top challenge mentioned by the surveyed members was achieving engagement and attendance in guild activities. The number of active members attending regular guild meetings account for only 20% on average, which is relatively low in percentage but not necessary when it comes to the number of people attending a meeting. Coordinators and sponsors were all in agreement that increasing engagement was important to be able to make better decisions and accomplish the guild work tasks. Some even felt stressed because they assumed the responsibility for ensuring attendance.

Lack of dedicated time. The challenge with member engagement is not new. Similarly to many other companies,[6,7] members of Spotify guilds reported having a lack of dedicated time for attending meetings and participating in the guild work.

Organizational support and priorities. Some respondents associated the lack of dedicated time with the lack of organizational support. Others were worried that guild work is not particularly prioritized and their individual contribution to guilds is not recognized by management. As one member explained, "Guild volunteers feel that time spent is not valued by the rest of the organization and we lose them to the tribe work that is valued."

On top of these known challenges, we found that scaling guilds introduced new barriers for mutual engagement. In what follows, we describe the main challenges of operating guilds in large-scale environment that are associated with the large size and separation between guild members.

Detachment. Respondents associated the large number of members and separation with detachment, difficulty to build a full sense of a joint community, and coordination challenges. When the community feeling is missing across sites, there is little incentive to strive for joint activities.

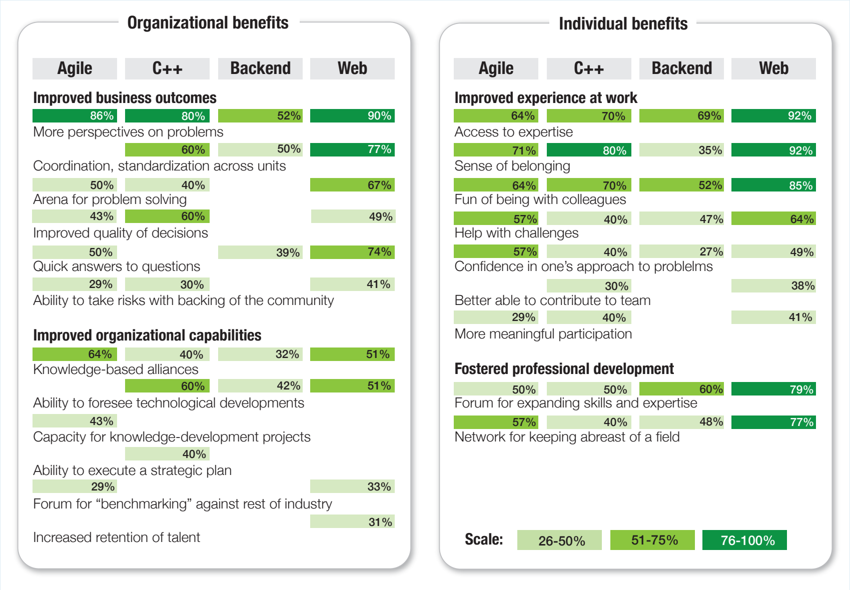
Fragmentation. Geographic distribution further impacts the way guilds operate. The lack of closeness and temporal distance across the U.S. and European sites challenges the ability to organize joint activities and, in some cases, has resulted in alternative guild structures—regional sub-guilds that act rather autonomously.

Difficulty to find common interests. Finally, we found that the higher the number and the diversity of guild members, the more challenging it is to find topics of mutual interest. When talking to the sponsor of the Web guild, we learned that one and the same practice can be understood differently by members from different organizational units or locations due to local traditions. Naturally, it has been difficult to choose discussion topics that are of relevance to everyone.

Mechanisms Fostering Engagement and Scaling

Although CoP researchers state that the majority of community members occupy peripheral roles, low member engagement in Spotify has practical negative implications. For example, the C++ engineering guild reported that not all impacted squads are represented in meetings, which makes it difficult to make good decisions about future development. Members of the Web guild complained that they fail to agree on what Web development is as a practice across the two main locations. When member who were absent in previous discussions

Figure 3. Heatmap of perceived individual and organizational value of the guilds.



Organizational benefits

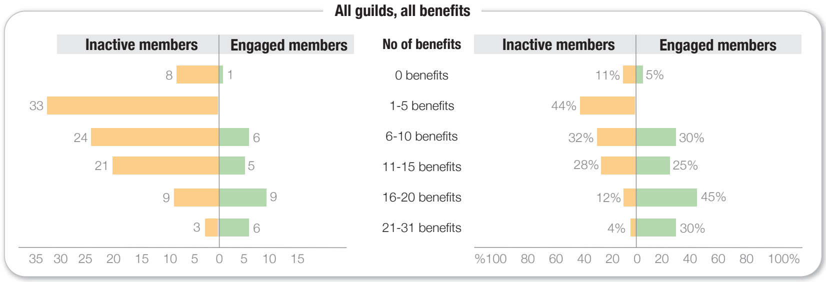
	Agile	C++	Backend	Web
Improved business outcomes				
More perspectives on problems	86%	80%	52%	90%
Coordination, standardization across units		60%	50%	77%
Arena for problem solving	50%	40%		67%
Improved quality of decisions	43%	60%		49%
Quick answers to questions	50%		39%	74%
Ability to take risks with backing of the community	29%	30%		41%
Improved organizational capabilities				
Knowledge-based alliances	64%	40%	32%	51%
Ability to foresee technological developments		60%	42%	51%
Capacity for knowledge-development projects	43%			
Ability to execute a strategic plan		40%		
Forum for "benchmarking" against rest of industry	29%			33%
Increased retention of talent				31%

Individual benefits

	Agile	C++	Backend	Web
Improved experience at work				
Access to expertise	64%	70%	69%	92%
Sense of belonging	71%	80%	35%	92%
Fun of being with colleagues	64%	70%	52%	85%
Help with challenges	57%	40%	47%	64%
Confidence in one's approach to problelms	57%	40%	27%	49%
Better able to contribute to team		30%		38%
More meaningful participation	29%	40%		41%
Fostered professional development				
Forum for expanding skills and expertise	50%	50%	60%	79%
Network for keeping abreast of a field	57%	40%	48%	77%

Scale: 26-50% 51-75% 76-100%

Figure 4. The amount of benefits reported by engaged and inactive respondents (absolute numbers and percentages).



join later, they often bring additional information and the guild is forced to revisit past discussions again. When analyzing the differences in member engagement, we found a number of coordination mechanisms that help to scale the guild activities and foster member engagement.

Yearly unconferences: Infrequent co-located gatherings of all guild members. Unconferences[3] are loosely structured conferences emphasizing the informal exchange of ideas according to the Open Space principles[8] and last for two to four days. These are the most engaging and most beneficial guild events facilitating knowledge sharing, networking and socialization, open for all members from all locations. As one of the survey respondents explained, “The conference every year really helps set the direction for what we want to accomplish as a community in the coming year.” The main weakness is that they happen only once a year, while the technology in certain areas and guilds changes very rapidly.

Lunch and learn seminars: Regular forums with specific topics. Some guilds organize lunch and learn seminars, in which internal or external experts talk about a selected topic of interest. As one of the survey respondents noted, “Lunch and Learns [are beneficial] to know more about new things that are being tried out.” Many guilds maintain a list of potential topics of interest on their Trello boards, where members can vote and prioritize the most relevant topics.

Slack channels: Electronically-mediated support forums. A lot of problems of individual guild members are solved through computer-mediated communication. For example, the biggest guild (the Backend guild) has no scheduled meetings, but has a group of volunteers, as large as 40+ members, who monitor and respond to questions posted in the guild’s support channel on Slack. As a survey respondent from the Backend guild explains, “Having Slack channels to ask questions has been the most helpful [for me], as a fairly inactive participant,” and a member of the Web guild explained, “Most valuable is simply chatting with other members of different Web organizations and seeing how they are solving the same problems we

Overview of the Study

We have performed an exploratory study of four out of eight Spotify guilds that receive organizational support (that is, sponsored guilds): Agile coaching, C++ engineering, Backend development and Web development. The selection was done to achieve a sample representing different types of active guilds with varying number of members and repertoire of activities (see general information about each of the guilds in Figure 1). The goal of our investigation was to understand what makes guilds successful. In particular, we were driven by the following research question:

How best to achieve mutual engagement and collaboration in guilds in large-scale agile organizations?

To answer this question, we explored the repertoire of guild activities, members engagement in these activities, the perceived value and benefits provided by the guilds for the organization and the individual members, what hinders and what fosters member engagement, and value creation in guild activities.

Data collection. We collected qualitative and quantitative data through interviews, observations, guild artifacts, and a survey (see the table here). We performed 11 semi-structured interviews with leaders of all guilds and four selected members of one guild. Interview questions were directed to understand a guild’s purpose, repertoire of activities, perceived benefits and challenges, and member engagement. We also received guild artifacts illustrating guild activities, and quantitative information regarding guild membership, and member attendance. Further, we conducted an online survey using the Mentimeter (www.mentimeter.com) tool to elicit member perception of guild value. Respondents were required to report their affiliation with one of the four selected guilds, their location, level of engagement, and then select benefits in four categories based on the value propositions suggested in prior research:[12] improved business outcomes, improved organizational capabilities, improved experience of work, and fostered professional development. In addition, respondents were given a chance to report, in a free-text format, what helps guilds to create value, and what hinders value creation.

Data collection methods.

Data Collected	Agile Coaching	C++ Engineering	Backend Development	Web Development	Total
Interviews	1 coordination	1 sponsor	1 coordinator	1 sponsor	11
		2 coordinators		1 coordinator	
		4 members			
Artifacts	Announcement of events, Screenshots of Guild Wiki and Trello Board, Unconference program	Screenshots of Guild Wiki and Trello Board, Unconference program	Screenshots of Guild Slack and Google mailing list, Unconference program	Announcement of events, Unconference program	
Survey					
Responded	14 members	10 members	62 members	39 members	125
Invited	82 members	100 members	305 members	180 members	667
Response rate	17%	10%	20%	22%	19%

Data analysis. Our data analysis strategy was twofold. First, our descriptory analysis aimed at explaining how different guilds function and what characterizes members and their engagement in selected guilds. The member division into different types (Figure 2) emerged when analyzing calendar invitations, meeting attendance and subscriptions to channels brought up by the interviewees. Then, exploratory analysis was preformed to identify what fosters and what hinders engagement and value creation. In doing so, the first two authors performed qualitative coding of the interview transcripts and qualitative survey responses. As a result, we built a table of hindrances and enablers for each guild with a frequency of occurrence, data sources (interviews and/or the survey), and quotations that provide explanations. We relied on methodological and data source triangulation to improve the validity of our findings. This was done by comparing data gathered through different means (interviews and survey), and from different types of members (active and inactive), and by focusing on the findings emerging from several rather than a single source.

To better understand if there are any associations and what differentiates the types of membership (for example, active and inactive members) we applied descriptive statistics to depict the benefits reported by different membership groups. Notably, the membership type was self-reported by the respondents. In particular, we used Chi-square test of association. To examine the strength of associations we used Cramer’s V test, which ranges in value from 0 (no association) to +1 (complete association). A value more than 0.5 indicates a strong association (guidelines according to Cohen[4]). Moreover, we performed Mann-Whitney U test,[1] a rank-based nonparametric test, to determine if there were any differences between active and inactive members on each one of the four categories of benefits proposed by Wenger et al.[12]

face. What technologies they are using, what standards they are employing, what practices they use."

Requests for comments: Electronically mediated opinion elicitation. The Request for Comments (RFC) procedure[10] is often used for eliciting opinions regarding specific technical changes. Any individual guild member can register a change using a shared template in a central repository and send it out to all guild members for review. Elicited questions, comments, and suggestions help to improve the RFC document, which remains publicly available. RFC approach enables guilds to have asynchronous and distributed decision-making on focused technical changes.

Conclusion and Recommendations

Our study shows that maintaining successful large-scale distributed guilds and active engagement is indeed a challenge. We found that only 20% of the members regularly engage in the guild activities, while the majority merely subscribes to the latest news. In fact, organizational size and distribution became the source of multiple barriers for engagement. Having too many members, and especially temporal distance, means that scheduling joint meeting times is problematic. As a respondent noted: "*Guilds seem bloated and diluted. There could be a need for a guild-like forum on a smaller scale.*" This is why regional sub-guilds emerged in response to the challenges of scale. At the same time, cross-site coordination meetings and larger socialization unconferences were recognized for their benefits. We therefore suggest that guilds in large-scale distributed environments offer both regional and cross-site activities.

Evidently, guild activities such as Spotify unconferences and meetups with external speakers require management support for covering traveling and organizational expenses. We found that management support, in fact, is very important for motivating guild members to engage in guild work. The traditional challenges such as the lack of dedicated time and the perception that the guild work is not prioritized or recognized by the organization, were also mentioned among the major barriers for engagement in Spotify. For a large and distributed organization this means that local management in each location shall have a common recognition of the importance of the knowledge sharing culture. We therefore emphasize that mutual engagement depends on the alignment of management attitudes and support across locations.

Yet, we found that guilds are well recognized for diverse benefits both for the organization and for the individual members. As we expected, engaged members reported more benefit than the passive members, but the vast majority of respondents reported at least some. Evidently, the very membership seems to generate valuable sense of belonging and fun of being with colleagues. This is due to the motivational potential of relatedness.[2] One interesting implication of our results is that having few attendants in the regular meetings is not necessarily a sign of failure. What matters is the diversity of value-adding activities. We therefore recommend offering different activities and channels for sharing knowledge and networking.

Last but not least, we found the guilds to be very diverse in terms of how they operate,[11] their members, and what value they create. The architecture of a guild depends on the practice it deals with, who is doing the practice, and how the members are distributed. This means that standardizing the way guilds operate and having the same expectations on the guild outcomes only make sense if the guilds concern the same practice and solve the same challenges.

So, do we recommend other companies to establish CoPs or guilds? The importance of implementing such parallel structures has been debated, and they do typically occupy the backseat in agile transformations and agile method implementations. However, Spotify experience shows that domain-specific, professional guilds is an important support for the squads and squad members. Guilds help new engineers get up to speed more quickly saving time for their colleagues. Guilds provide forums to tackle shared, emerging problems and opportunities with response times much shorter than individual experts would be able to provide. Besides, guilds' yearly events connect people across locations that would otherwise never meet. Therefore, we do recommend others consider cultivating participation culture in general and CoPs/guilds in particular. The barriers and mechanisms described in this article shall help companies—small and large—in this journey.

References

1. Agresti, A. *Categorical Data Analysis*, 3rd ed. John Wiley & Sons, 2013.
2. Bass, J., Beecham, S., Razzak, M.A. and Noll, J. Employee retention and turnover in global software development: Comparing in-house offshoring and offshore outsourcing. In *Proceedings of the Intern. Conf. Global Software Engineering*, 2018
3. Budd, A. et al. Ten simple rules for organizing an unconference. *PLoS Comput Biol 11*, 1 (2015).
4. Cohen, J. *Statistical Power Analysis for the Behavioral Sciences*, 2nd edn. Lawrence Erlbaum, 1988.
5. Millen, D.R., & Fontaine, M.A. Improving individual and organizational performance through communities of practice. In *Proceedings of the 2003 Intern. ACM SIGGROUP Conf. on Supporting Group Work*. ACM, Nov. 2003, 205–211.
6. Millen, D.R., Fontaine, M.A., and Muller, M.J. Understanding the benefit and costs of communities of practice. *Commun. ACM 45*, 4 (Apr. 2002), 69–73.
7. Oliver, S. and Reddy Kandadi, K. How to develop knowledge culture in organizations? A multiple case study of large distributed organizations. *J. Knowledge Management 10*, 4 (2006), 6–24.
8. Owen, H. *Open Space Technology: A User's Guide*. Berrett-Koehler Publishers, San Francisco, CA, USA, 2008; http://openspaceworld.org.
9. Paasivaara, M. and Lassenius, C. Communities of practice in a large distributed agile software development organization–Case Ericsson. *Information and Software Technology 56*, 12(2014), 1556–1577.
10. RFC Editor. Independent submissions; https://www.rfc-editor.org/about/independent
11. Šmite, D., Moe, N.B., Levinta, G., and Marcin, F. Spotify Guilds—Cultivating knowledge sharing in large-scale agile organizations. *IEEE Software*, Mar./Apr. 2019.
12. Wenger, E., McDermott, R.A., and Snyder, W. *Cultivating Communities of Practice: A Guide to Managing Knowledge*. Harvard Business Press, 2002.

Darja Smite (darja.smite@bth.se) is a professor of software engineering at the Blekinge Institute of Technology, Karlskrona, Sweden and a part-time research scientist at SINTEF ICT.

Nils Brede Moe (Nils.B.Moe@sintef.no) is a senior researcher at SINTEF, Trondheim, Norway.

Marcin Floryan (mfloryan@spotify.com) is director of engineering at Spotify, Stockholm, Sweden.

Georgiana Levinta (georgiana@spotify.com) is a senior engineering manager at Spotify, Stockholm, Sweden.

Panagiota Chatzipetrou (pchatzip@gmail.com) is an assistant professor at Örebro University, Örebro, Sweden.

Watch the authors discuss this work in an exclusive *Communications* video. https://cacm.acm.org/videos/spotify-guilds

DOI:10.1145/3368856

A codable computer half the size of a credit card is inspiring students worldwide to develop core computing skills in fun and creative ways.

BY JONNY AUSTIN, HOWARD BAKER, THOMAS BALL, JAMES DEVINE, JOE FINNEY, PELI DE HALLEUX, STEVE HODGES, MICHAŁ MOSKAL, AND GARETH STOCKDALE

The BBC micro:bit—From the U.K. to the World

IN 2015, THE BBC launched the Make It Digital initiative, aiming to encourage a new era of creativity in the young using programming and digital technology as its medium. Simultaneously, the initiative also would support the U.K.'s mandate to teach computer science concepts at all grade levels.[13]

The micro:bit is a small programmable and embeddable computer designed, developed, and deployed by the BBC and 29 project partners to approximately 800,000 U.K. Year 7 (11/12-year-old) school children in 2015–2016. Referring back to its work with the BBC Micro,[4] the BBC described the micro:bit as its "most ambitious education initiative in 30 years, with an ambition to inspire digital creativity and develop a new generation of tech pioneers."[1]

Embracing a constructionist approach to computing education,[11] the micro:bit has moved from a local educational experiment in the U.K. to a global effort driven by the Micro:bit Educational Foundation (microbit.org), a nonprofit organization established in September 2016. There are now over four million micro:bits in the market in over 60 countries with many hardware, content, and education partners participating.

The BBC and its partners developed the micro:bit as an inexpensive, powerful, and easy-to-use learning tool guided by five major design goals:

1. *Have a low barrier to entry.* Financial cost and simplicity are important considerations for any technology, but even more so in an educational setting. The micro:bit needed to be affordable, easy to deploy, intuitive to use, simple to program, and integrate well with existing school IT infrastructure.

2. *Be fun and creative.* The micro:bit itself needed to offer an exciting, engaging, inclusive introduction to coding and making. Inspired by Arduino and the Maker movement,[7] the project sought to turn teachers and students from digital consumers into digital creators by integrating the micro:bit into their own real-world, physical creations.

3. *Have a low floor, high ceiling, and wide walls.* When designing the micro:bit, providing good educational value to students and teachers was the prime consideration. It needed to be easy for inexperienced learners to get started (low floor); enable rich learning opportunities that grow with user expertise, provide progression in both programming language and application complexity (high ceiling); and enable students to reach the ceiling via multiple pathways to embrace a diverse audience (wide walls).[11,15]

4. *Open a window into the future.* Computing technology is becoming ever more ubiquitous, connected, and embedded. In the 1980s, the BBC Micro[4] captured the essence of the devices that were to come over the next

Girlsday, hosted by Microsoft, The Netherlands, drew many happy participants.

PHOTO COURTESY OF PETER HELDEN, @PETERHELDENS.

30 years: the desktop PC. The micro:bit was designed as a modern-day equivalent, capturing the connected, embedded nature of devices that are to come for the *next 30 years*.

5. *Be applicable beyond computer science.* Cross-curricular activities can offer diverse and inclusive learning.[3,12,16] This is important when we consider the gender disparity in computing today. The micro:bit project aimed to stimulate curiosity about how computing can be applied across a variety of disciplines, ranging from science and technology/engineering to the arts and mathematics (STEAM).

In this article, we describe the design of the BBC micro:bit and the realization of these goals, exemplified through a sample set of diverse projects. We review the project's history as it transitioned from a U.K.-centric to a worldwide project, concluding with lessons learned and project outcomes.

The BBC micro:bit

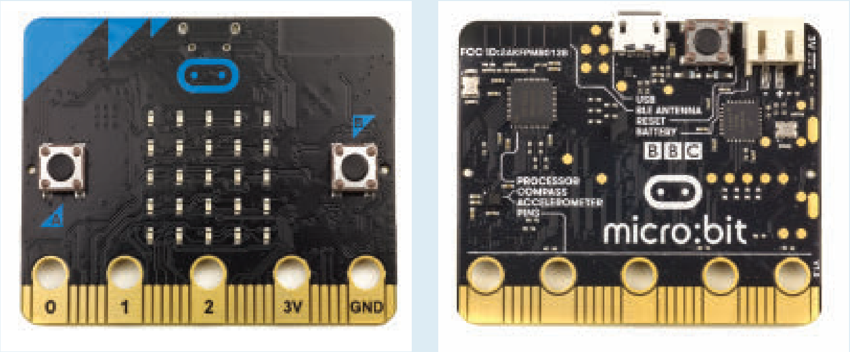
The BBC spent two years investigating previous work and new ideas to get more children coding and to improve digital literacy. Research shows that physical computing—combining software and hardware to build interactive physical systems that sense and respond to the real world—can engage a diverse range of students.[10] The simultaneous global interest in the maker movement also suggests an appealing way to engage children is to incorporate making, creating, and inventing as part of the software development process.[7,9]

However, the BBC observed there was no prior technology on the market that suited the complete novice and that had been designed as an educational tool from the outset. For example, Arduino[19] set a new standard in the field, but requires wiring for virtually all of its projects as well as the installation of a custom IDE and device drivers. The Raspberry Pi is a highly capable device that runs a full operating system, but also has a reliance on additional peripherals to enable physical computing. Its associated high power consumption and complexity also means it cannot be easily run from battery power and embedded into children's projects. There also are cost implications for children, parents, and schools wanting to start making: devices and accessories need to be affordable enough to be accessible by children and parents from a variety of backgrounds.

Engaging, capable, hardware. Figure 1 shows (a) the front and (b) the back of the micro:bit, which measures 4cm x 5cm. Like many "development boards," the micro:bit is an exposed printed circuit board with all its components visible (in fact, explicitly labeled, as a learning opportunity). The micro:bit is designed to be engaging and interactive from the start: the front is designed to resemble a face with colored streaks of hair (upper left) and eyes as the logo (upper middle).

This playful design should not be mistaken for a lack of capability. The board is based around a modern 32-bit ARM Cortex-M processor (16kB RAM; 256kB non-volatile flash) and hosts an array of input/output capabilities including a 5x5 LED matrix,

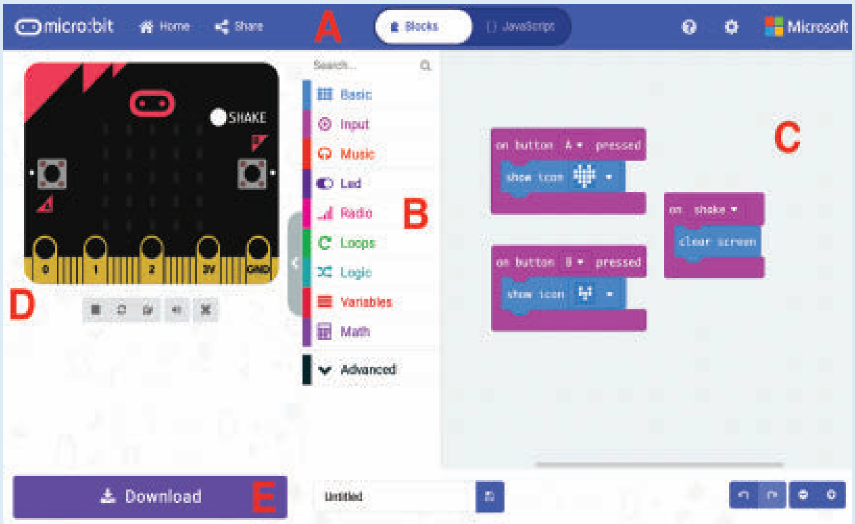
Figure 1. The BBC micro:bit.



(a)
front, with two buttons,
5x5 LED display,
and edge connector (bottom)

(b)
back, with processor, accelerometer,
compass, Bluetooth,
USB and battery connectors

Figure 2. The MakeCode Web app for the micro:bit (https://makecode.microbit.org).



two programmable buttons, the ability to sense motion, gestures, magnetic fields, temperature and light. The device also includes a USB interface and edge connector with touch sensitive, digital/analog pins that allow external sensors, and actuators to be connected via crocodile clips or banana plugs. Finally, the device can communicate with phones, tablets, and computers via Bluetooth Low Energy (BLE) or directly with other micro:bits using a low-level 2.4GHz radio protocol. The ability to run on battery power and an ecosystem of micro:bit hardware peripherals that plug into the micro:bit's edge connector further expand its capabilities.

Engaging, simple, software. The design of the micro:bit coding tools is oriented toward a simple and inclusive starting experience with room for progression. In-school trials with a micro:bit prototype validated the BBC's approach of using a Web app based on the popular Blockly framework[8] for students to create scripts via the block-based visual programming paradigm pioneered by Scratch,[14] and providing a simulator for students to execute and debug their programs, all inside a Web browser.

In addition to block-based visual coding, support for text-based coding via scripting languages was identified as an important feature. As the micro:bit would be incorporated into standalone projects, it was essential for the user's program to be stored on the device for future untethered execution via battery power. This allows a student to unplug their micro:bit from a computer and show their creation to a teacher, parent or friend wherever or whenever they want.

The solution delivered by the BBC's partners includes support for Blockly, JavaScript and Python, all via Web apps. Figure 2 shows a screen snapshot of Microsoft's MakeCode (https://makecode.com) Web app for the micro:bit, which supports programming via both Blockly and JavaScript. The Web app has five main sections: (A) menu bar with access to projects/examples and switching between Blockly and JavaScript editors. To support progression, the editor also supports conversion of programs between Blockly and JavaScript—users can round-trip programs to see their code in visual or text-based representations; (B) Blockly toolbox of micro:bit API categories, representing the hardware capabilities of the micro:bit. This toolbox can be expanded through third-party extensions; (C) Blockly programming canvas showing a simple reactive program. MakeCode enables event-based programming through a lightweight scheduler in the underlying micro:bit runtime; (D) micro:bit simulator for execution of the user's program in browser; (E) download button, which invokes an in-browser compiler/linker to produce a binary executable (a "hex file").

The Python solution for the micro:bit is based on MicroPython (https://micropython.org), an implementation of Python 3.0 for microcontrollers. It includes a full Python compiler and runtime that executes on the micro:bit and supports a read-eval-print loop to execute commands sent via a terminal, for interactive use. This solution also allows a Python script to be embedded alongside the compiler/runtime and downloaded as a hex file from the Python Web app for the micro:bit (https://python.microbit.org).

A low-friction end-to-end experience. Figure 2C illustrates a simple coding example for the micro:bit, which displays a large heart when button A is pressed, a small heart when button B is pressed, and clears the display when the user shakes the micro:bit (shake detection is implemented using the accelerometer). The interactive micro:bit simulator (Figure 2D) models all functions of the micro:bit and allows the user to test that the program works as expected. The shake event can be fired using a virtual button (white circle labeled "SHAKE"), or by moving the mouse back and forth rapidly over the simulator.

To generate a binary executable for the micro:bit, the user simply presses the "Download" button (Figure 2E), which invokes an in-browser compiler tool chain that translates the Blockly program to JavaScript and then to machine code, linking the user's compiled code against a pre-compiled C++ runtime.[6] This means that no C++ compiler is required for compiling the user's program into an executable binary; the same is true of the MicroPython solution.

When plugged into a host computer via USB, the micro:bit appears as a 'memory stick' storage device. A compiled program can be transferred (flashed) to the micro:bit by a simple file copy operation, installing the executable binary into the micro:bit's non-volatile flash memory. This makes it compatible out-of-the-box with almost all school computers and eliminates the complexity of installing device drivers—something that teachers and children rarely have permission to do. Once flashed, the micro:bit then can be embedded into projects where it runs on battery power.

Design summary. We conclude this section with a reflection on the five design goals stated earlier. (1) The micro:bit's inexpensive hardware lowers the financial barrier to entry for students, parents and teachers. Its Web-based software requires no installation, lowering technical barriers to adoption in schools and homes. The micro:bit's integrated sensors and outputs allow students to explore a range of lessons and projects without the need for external electronic components. (2) The design of the device prompts a sense of fun, alongside colorful programming blocks that allow for complete control over the device and its peripherals, backed up by a range of creative learning materials and projects. (3-4) Programming experiences spanning Blockly, JavaScript and Python provides a clear progression path when combined with project-based learning. Radio and Bluetooth networking allow further progression to more complex projects with other micro:bits, smartphones and other Internet connected devices. (5) Finally, the ability to run on battery power combined with sensors, non-volatile storage and edge connector allows for the integration of the micro:bit into areas of the curriculum that make use of physical experiments and data collection.

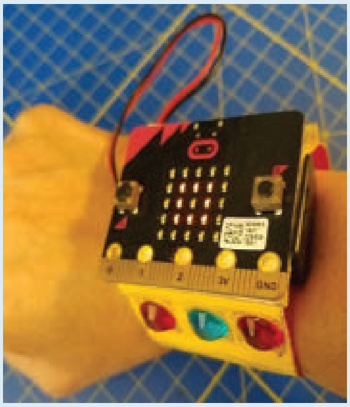
Projects

A wide array of curriculum-aligned lessons are available for the micro:bit. However, physical computing devices also lend themselves to creative (and often collaborative) projects that promote deep problem-based learning. Here we provide some examples of such educational projects for the micro:bit, grouped into four broad classifications of use, each showing many of our design goals in action.

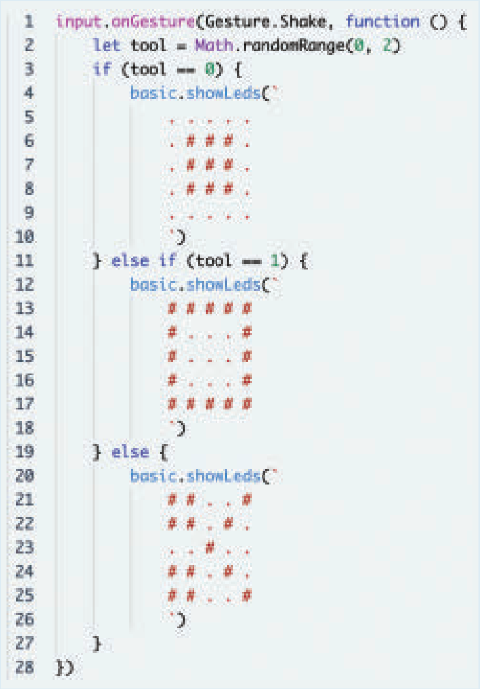
Wearables and interactive play. Many projects involve the use of a micro:bit as an interactive mobile device—either as a handheld or wearable. Figure 3 shows a simple but highly popular micro:bit project: a micro:bit 'watch' that plays the rock/paper/scissors game by randomly displaying a rock (3x3 square), paper (5x5 square with center empty) or scissor icon on the 5x5 LED display when the device is shaken. Other popular examples include name and emoji badges, a graphical compass that points North based on magnetometer data, and gesture-based games such as 'Snake' which use the tilt of the device to control the behavior of objects shown on the LED display.

Digital crafting. Other popular projects augment a micro:bit with simple classroom supplies, allowing students to quickly create low cost, playful and practical digital artifacts. For example, Figure 4(a) shows how cardboard and aluminum foil can be used to build a competitive game known as 'Reaction.' Crocodile clips are used to connect pins P0, P1, P2, and GND of the micro:bit to conductive aluminum foil pads glued to a cardboard gameboard. Users are challenged to be the first to complete a circuit by touching the GND pad and one of other pads when the micro:bit display lights up. Note the blending of form and function evident in this design, including the positioning of the interface for multiuser

Figure 3. A micro:bit watch.



(a)
wearable form-factor
rock/paper/scissors game



```
input.onGesture(Gesture.Shake, function () {
    let tool = Math.randomRange(0, 2)
    if (tool == 0) {
        basic.showLeds(`
            . . . . .
            . # # # .
            . # # # .
            . # # # .
            . . . . .
            `)
    } else if (tool == 1) {
        basic.showLeds(`
            # # # # #
            # . . . #
            # . . . #
            # . . . #
            # # # # #
            `)
    } else {
        basic.showLeds(`
            # # . . #
            # # . # .
            . . # . .
            # # . # .
            # # . . #
            `)
    }
})
```

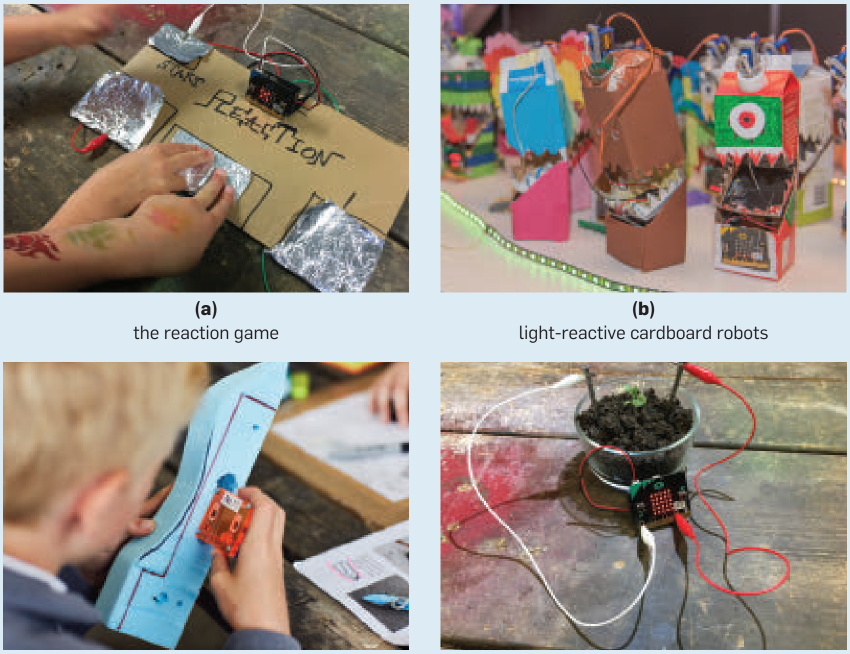
(b)
the JavaScript of the game

access and the additional touch pad labeled "START."

Actuation adds a further dimension: the micro:bit can control servos and motors via its edge connector. This has resulted in the creation of inexpensive, cardboard based creations that can react to their environment, such as those shown in Figure 4(b): these simple robots open and close their mouths in response to light stimulus. Other similar examples include musical instruments, such as a 'guitar' that changes pitch based on its physical orientation, and goal line technology for tabletop football games.

Science and measurement. The micro:bit's small size and built-in sensors make it well suited to being embedded into science and technology projects for undertaking data measurement. A great example of this is provided by the Bloodhound project (http://www.bloodhoundssc.com), a U.K. initiative to set a new land speed world record. As part of their remit to inspire students about STEM subjects, the 'Race to the Line' project was launched across the U.K. In this project, students design, build, and race model rocket cars in competition, learning about physics, aerodynamics, engineering, and measurement. A micro:bit is integrated into the car's design, as shown in Figure 4(c). The micro:bit captures three-axis accelerometer data of the rocket car during its race. After the race, students upload the data from the micro:bit and analyze the performance of their cars.

Figure 4. Example projects.



(a)
the reaction game

(b)
light-reactive cardboard robots

(c)
a Bloodhound model rocket car instrumented with a micro:bit

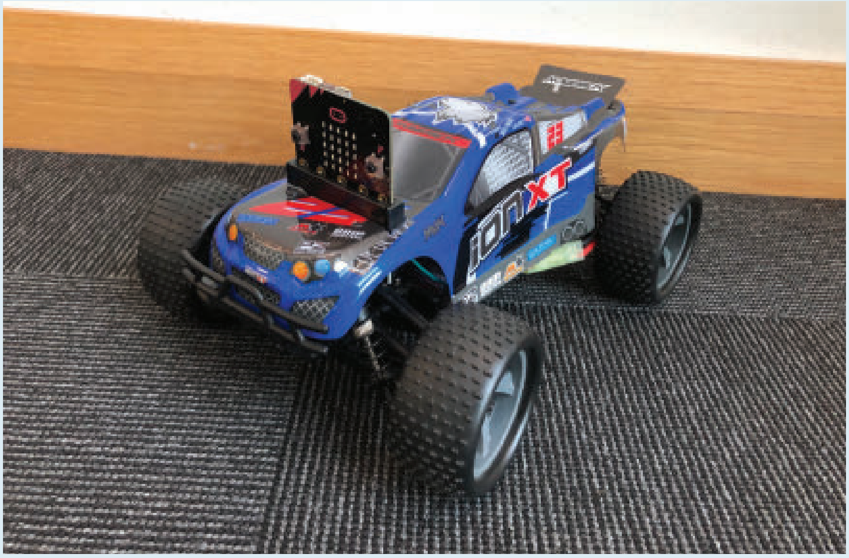
(d)
measuring soil moisture via micro:bit pins

Similarly, Figure 4(d) illustrates an environmental project that uses the micro:bit to measure soil moisture. The combination of water and nutrients in soil affect its conductivity—the more water, the greater the conductivity. This can be directly measured using metallic probes (note the use of inexpensive nails as probes here) and the micro:bit's integrated analog voltage sensor. Then, the micro:bit is programmed to periodically take a moisture reading and record the results into the device's internal flash file system for later analysis.

Interconnected devices. Our final class of projects are those that make use of multiple, wirelessly interconnected devices. The micro:bit has an inbuilt Bluetooth low energy (BLE) compatible 2.4GHz radio. BLE provides a private and secure mechanism through which the micro:bit can be programmed over-the-air from mobile phones and tablets, and also provides an API through which the micro:bit can be paired, and its sensors and actuators made available to applications running on such devices through a well-defined Bluetooth profile. MIT's Scratch 3.0 includes micro:bit support through this API, for example.

However, we observed the greatest level of innovation emerged from a simpler, custom-built packet radio protocol running on the same 2.4GHz hardware. With the micro:bit radio API, micro:bits can form low-level peer-to-peer multicast groups. Any data sent from one micro:bit is seen by all members of their group—thus

Figure 5. A micro:bit-based vehicle controlled wirelessly by a second micro:bit.



enabling a simple yet powerful basis for projects involving group collaboration in a way not feasible with BLE. Examples here include remote control vehicles, such as that illustrated in Figure 5. This example uses two micro:bits sharing data over radio: one integrated into the vehicle to control steering and speed, and a second integrated into a handheld steering wheel that is used as a remote control. A second popular example is illustrated in Figure 6, which mimics how fireflies synchronize their blinking over time, as described in the accompanying Python program.

From the U.K. to the World

Here, we detail the history of the project, the approach taken to deliver 800,000 devices to students and their teachers in the U.K., and how the Micro:bit Educational Foundation is taking the micro:bit worldwide.

BBC micro:bit partnership. The BBC invited 29 partners to contribute hardware, software services, teaching materials, packing/distribution, logistics, events, and funding. These partners were not directly funded with public money by the BBC for their work on the project. Rather, partners contributed their own resources to make the micro:bit vision a reality.

The BBC looked for three types of partner for the original project: those who could help on the technical development; the manufacturing/distribution of the micro:bit, and, very importantly, those who could help with education—both child and the teacher. A large proportion of the 29 partners played a role in creating teaching resources, delivering teacher training and on-the-ground support in collaboration with grass roots organizations such as the U.K. Computing At School (CAS) network[5]—a group of over 30,000 computing teachers, supporters, and enthusiasts. These partners used such networks of practice to engage with over 90% of the 8,000 secondary schools in the U.K. even before the micro:bit was manufactured.

This activity was orchestrated and supported by a broad BBC team dedicated to raising awareness of the project. This essential activity ensured the teachers and students understood the aims and potential of the micro:bit ahead of the devices being available in schools. This team developed broadcast TV and media content with BBC talent like Peter Capaldi, will.i.am, and Paloma Faith, and major BBC brands including *Doctor Who*, *The Voice*, *Robot Wars*, and *Wolfblood*. They also organized nationwide public engagement activities including Make It Digital Roadshows that took place in 10 cities around the U.K. that had a combined footfall of approximately 100,000 people in the summer of 2015.

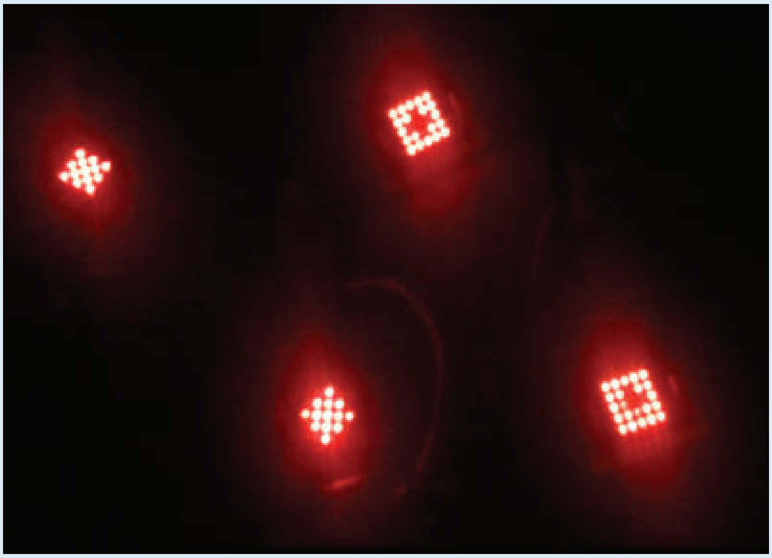
Once product manufacturing, testing, and certification was complete, 800,000 micro:bit devices were delivered into U.K. schools in March 2016—one device for every Year 7 (11/12 year old) child in the U.K. The micro:bit was well received by U.K. teachers and children, with high levels of engagement. In the first six months, there were approximately 13 million visits to the website, 10 million runs of the online micro:bit simulator, and two million programs downloaded to a micro:bit.

Expanding the scope. The U.K. micro:bit deployment sparked interest from around the world. In October 2016, the Micro:bit Educational Foundation was formed—a U.K.-based nonprofit organization that now serves as the custodian of the micro:bit legacy. Its vision is to inspire every child to create their best digital future, with focus on widening participation around gender and disadvantaged groups across the globe. It is funded through a small royalty on every micro:bit sold and by corporate sponsorship.

The foundation strives toward its vision by coordinating work across a broad partnership of educators, technologists, enthusiasts, and governments to bring about global change. Partnership has proven to be the vital heart of all the foundation's activities. More specifically, the foundation maintains partnerships with over 132 global organizations. These can be grouped into:

Hardware partnerships with manu-

Figure 6. Fireflies example: A visual representation of emergent behavior from a distributed algorithm, implemented in Python.



```
clock = 0
flash = []
for i in range(9, -1, -1)]:
    flash.append(Image().invert()*(i/9))

while True:
    incoming = radio.receive()
    while incoming == '0':
        clock = clock + 1
        incoming = radio.receive()
    if clock >= 8:
        radio.send('0')
        display.show(flash, delay=100, wait=False)
        sleep(200)
        clock = 0
    else:
        sleep(100)
        clock = clock + 1
```

micro:bit Artifacts

The BBC micro:bit hardware and software assets are open source at https://github.com, at the following locations:

- **/bbcmicrobit for the micro:bit hardware design, the micro:bit software prototype and theMicroPython port for the micro:bit;**
- **/microsoft/pxt-microbit for MakeCode for the micro:bit;**
- **/lancaster-university/microbit-dal for the micro:bit C++ runtime.**

A wide variety of educational resources for the micro:bit can be found at https://microbit.org. Resellers of the micro:bit and accessory manufacturers can be found at https://microbit.org/resellers.

facturers, suppliers, and resellers ensure a pipeline of micro:bits are available in 60 countries to date. Also essential are partnerships with accessory makers who create kits that enable projects such as those described earlier. Hundreds of third-party accessories have been created for the micro:bit. The most common examples include additional sensors (sound level, moisture, particulate, and ultrasonic ranging sensors), actuators (motor drivers, audio speakers, addressable LED, light strips), hardware prototyping kits that interface to breadboards, and finally, application-specific peripherals such as wheeled robots, remote controlled vehicles, and games.

Software partnerships with organizations including Lancaster University, Microsoft, and the Python Software Foundation ensure a diverse offering of programming languages and highly reliable, state-of-the-art editors for the micro:bit.

Countrywide partnerships with governments, charitable organizations, and regional companies enable trials and rollouts to schools around the world. To date, this has resulted in national scale deployment in Canada, Croatia, Denmark, Norway, Iceland, Singapore, Hong Kong, Uruguay, and the Western Balkan states, in addition to the U.K., with the British Council planning similar activity in the Western Balkan states in 2019.

Community partnerships that provide essential 'on the ground' support. Examples here include the generation and sharing of learning resources and experiences between teachers, crowdsourced translation of teaching materials into languages other than English, social media activists who reach out to hard-to-reach demographics, and volunteers that provide technical support.

Lessons Learned

We learned a number of lessons through the micro:bit project that might be helpful to others working in the space of CS education and physical computing:

Community-centered design. User-centered design was a key part of the BBC's approach to the micro:bit, considering a broad range of stakeholders including children, teachers, product developers, manufacturers, enthusiasts, and support organizations. This process identified physical computing as the main focus area and pointed the way to the key design decisions including the integrated board design (inspired by Arduino), and the need for block-based and scripting languages rather than the C-based sketches used by Arduino. Yet, as the project developed it became something greater—a process by which a community of practice emerged, consisting of those individual stakeholders. This enabled the strong and sustained ecosystem around the micro:bit, long after the initial U.K. project was completed.

Depth is just as important as ease of use. Although providing multiple layers of abstraction was central to realizing the "low-floor, high-ceiling" concept, we did not sacrifice the opportunity to dig deeper, learn key computing concepts underneath, and learn from the realities of the computing world. Take the packet-based broadcast radio interface, for example. This interface is entirely lossy, incorporating a simple checksum and providing no reliability guarantees. A user could send numbers using the radio to indicate states within a distributed application, but would need to include device identifiers as it scales, and if needed, algorithms for reliability. Before long the user has implemented their own networking protocol with real-world applicability.

An always connected experience is restricting. In the BBC prototype for the micro:bit, the text of a user's program was submitted to a compile service in the cloud that returned a final executable to be copied onto a micro:bit. We originally adopted this architecture for the micro:bit, but in trials across many schools in the U.K. found the assumption of "always connected" was not a good one. As a result, we eliminated the need for a cloud service by writing a compiler and linker in JavaScript that would produce the needed binary directly in the Web app. Once the Web app loads, no further connectivity is needed in order to edit, compile, and flash the program to the micro:bit.

Partnerships and localization are key to global expansion. The national scale deployments of micro:bit and further large-scale trials in countries such as Sweden, Taiwan, and Uruguay taught us that localization of all aspects of the approach is essential for success. Beyond the predictable challenges of language translation, there are many other aspects related to funding, educational priorities, and cultural diversity. For example: the U.K. rollout was funded entirely through donations from industry and charities; Iceland's rollout was funded by its government; and Croatia's rollout stemmed from a crowdsourced fund set up by a motivated regional entrepreneur. Likewise, teaching materials designed for the U.K. do not readily translate to other countries. Moreover, school projects are often based around local cultural events. We have learned that a combination of local and global partnership is the key to embracing diversity.

Compromises must always be made. To make the micro:bit hardware available to as many people as possible, it was critical to keep the cost low. This influenced the choice of components and capabilities of the hardware, which inevitably means trade-offs and limitations. For example, the micro:bit's 16kB of RAM is quickly consumed, especially when the full Bluetooth stack is loaded, which can be frustrating for those who want to build larger applications. The 5x5 LED display is optimized to display a single ASCII character, but falls short when non-Latin character sets are considered, creating challenges for international adoption. The design of the micro:bit edge connector makes it easy to plug the micro:bit into another board, but is inherently non-compositional, compared to the approach of Arduino that allows stacking of boards via its headers. Without these difficult choices, the micro:bit would not have become a reality.

Timing is critical. As with any complex endeavor, luck favors the prepared. The U.K. was the first country to mandate computing education for K–12, there was a large group of volunteer computing organizations in the U.K. to call upon, and Moore's Law had brought microcontroller and networking technology (such as Bluetooth) down in cost so as to enable delivery at scale economically.

Outcomes

Since the initial distribution of micro:bits in 2016 we have observed significant interest, enthusiasm, and adoption. The BBC micro:bit is now available in 60 countries and 24 languages, and in excess of four million devices have been delivered to end users globally with an increasing demand year over year. The online editors have hundreds of thousands of independent sessions every month. Activity on social media and support networks also indicate high levels of use for micro:bit resources in schools, particularly those related to the constructionist pedagogy.

Independent research undertaken in the U.K. (see https://microbit.org/ta/2017-07-07-bbc-stats by Discovery Research) supports these observations.[17,18] An independent survey[2] of 405 U.K. school children and their teachers concluded that:

- 86% of students said the micro:bit made computer science more interesting;
- 70% more girls said they would choose computing as a school subject after using the micro:bit;
- 85% of teachers agreed it made ICT/computer science more enjoyable for their students;
- Half of teachers who have used the micro:bit said they felt more confident as a teacher, particularly those who said they were not very confident in teaching computing.

Although preliminary studies are encouraging and guide our thinking, they are small compared to the scale of the BBC micro:bit project. It will take more time to determine the full impact of the micro:bit. We look forward to further research studies that will investigate the advantages and challenges of using the micro:bit in supporting teaching and learning, both within computing and in wider cross curricular ways.

Acknowledgments

The BBC micro:bit project's 29 partners were Arm, Barclays, Bluetooth Special Interest Group, Cannybots, Creative Digital Solutions, Cisco, Code Club, Code Kingdoms, CoderDojo, CultureTECH, element14, the Institute of Engineering and Technology, Kitronik, Lancaster University, London Connected Learning Centre, Microsoft, MyMiniFactory, National STEM Centre, Nordic Semiconductor, NXP, the Python Software Foundation (PSF), Samsung, ScienceScope, STEMNET, Tangent Design, Technology Will Save Us, Teen Tech, the Tinder Foundation, and the Welcome Trust.

The authors also recognize the contributions of Jeannette Wing, Zach Shelby, the Computing At School organization and its members, Damien George and Nicholas Tollervey at the PSF, Clare Riley, Jonathan Protzenko, Michael Braun and the MakeCode team at Microsoft, Martin Wooley at the Bluetooth SIG, Mike Powell and Jonathan Smith at Premier Farnell, Jo Claessens, Cerys Griffiths and the many other colleagues across the BBC who supported this initiative. Without the tireless contributions from all these organizations and individuals the micro:bit project would not have been a success.

References

1. BBC. The BBC micro:bit, 2015; http://www.bbc.co.uk/programmes/articles/4hVG2Br1W1LKCmw8nSm9WnQ/the-bbc-micro-bit.
2. BBC. BBC micro:bit celebrates huge impact in first year, with 90% of students saying it helped show that anyone can code, 2017; https://www.bbc.co.uk/mediacentre/latestnews/2017/microbit-first-year.
3. Blikstein, P. Digital fabrication and 'making' in education: The democratization of invention. *FabLabs: of Machines, Makers and Inventors*, 2013, 4:1-4:21.
4. Blyth, T. The legacy of the BBC micro: Effecting change in the U.K.'s cultures of computing. May 2012; https://media.nesta.org.uk/documents/the_legacy_of_bbc_micro.pdf.
5. Crick, T. and Sentance, S. Computing at school: Stimulating computing education in the U.K. In *Proceedings of the 11th Koli Calling Intern. Conf. Computing Education Research.* ACM, 2011, 122–123.
6. Devine, J., Finney, J., de Halleux, P., Moskal, M., Ball, T., and Hodges, S. Makecode and CODAL: Intuitive and efficient embedded systems programming for education. In *Proceedings of the 19th ACM SIGPLAN/SIGBED Intern. Conf. on Languages, Compilers, and Tools for Embedded Systems.* LCTES 2018, 19–30.
7. Dougherty, D. The maker movement. *Innovations: Technology, Governance, Globalization 7*, 3 (2012), 11–14.
8. Fraser, N. Ten things we've learned from Blockly. In *Proceedings of the 2015 IEEE Blocks and Beyond Workshop*, 49–50.
9. Halverson, E.R. and Sheridan, K. The maker movement in education. *Harvard Educational Review 84*, 4 (2014), 495–504.
10. Hodges, S., Scott, J., Sentance, S., Miller, C., Villar, N., Schwiderski-Grosche, S., Hammil, K., and Johnston, S. .NET.Gadgeteer: A new platform for K-12 computer science education. In *Proceeding of the 44th ACM Technical Symp. Computer Science Education*, ACM, New York, NY, USA, 2013, 391–396.
11. Papert, S. *Mindstorms: Children, Computers and Powerful Ideas.* Basic Books, 1993.
12. Peppler, K. STEAM-powered computing education: Using E-textiles to integrate the arts and STEM. *IEEE Computer*, (2013), 1.
13. Peyton Jones, S. Computer science as a school subject. In *Proceedings of the 18th ACM SIGPLAN Intern. Conf. Functional Programming.* ACM, 2013, 159–160.
14. Resnick, M. et al. Scratch: Programming for all. *Commun. ACM 52*, 11 (Nov. 2009), 60–67.
15. Resnick, M. and Silverman, B. Some reflections on designing construction kits for kids. In *Proceedings of the 2005 Conf. Interaction Design and Children.* ACM, 2005, 117–122.
16. Robelen, E.W. STEAM: Experts make case for adding arts to STEM. *Education Week 31*, 13 (2011), 8.
17. Sentance, S., Waite, J., Hodges, S., MacLeod, E., and Yeomans, L. 'Creating cool stuff:' Pupils' experience of the BBC micro:bit. In *Proceedings of the 2017 ACM SIGCSE Tech. Symp. Computer Science Education.* ACM, 2017, 531–536.
18. Sentance, S., Waite, J., Yeomans, L., and MacLeod, E. Teaching with physical computing devices: The BBC micro:bit initiative. In *Proceedings of the 12th Workshop on Primary and Secondary Computing Education.* ACM, 2017, 87–96.
19. Severance, C.R. Massimo Banzi: Building Arduino. *IEEE Computer 47*, 1 (2014), 11–12.

Jonny Austin is chief technology officer at the Micro:bit Educational Foundation, London, U.K.

Howard Baker is an education researcher at the Micro:bit Educational Foundation, London, U.K.

Thomas Ball is a partner researcher at Microsoft Research, Redmond, WA, USA.

James Devine is a Ph.D. student in the School of Computing and Communications at Lancaster University, U.K.

Joe Finney is a professor in the School of Computing and Communications at Lancaster University, U.K.

Peli de Halleux is a principal research software development engineer at Microsoft Research, Redmond, WA, USA.

Steve Hodges is a senior principal researcher at Microsoft Research, Cambridge, U.K.

Michał Moskal is a principal research software development engineer at Microsoft Research, Redmond, WA, USA.

Gareth Stockdale is chief executive officer at the Micro:bit Educational Foundation, London, U.K.

DOI:10.1145/3326601

Technologies for manipulating our digital appearance alter the way the world sees us as well as the way we see ourselves.

BY OHAD FRIED, JENNIFER JACOBS, ADAM FINKELSTEIN, AND MANEESH AGRAWALA

Editing Self-Image

SELF-PORTRAITURE HAS BECOME ubiquitous. Once an awkward feat, the "selfie"—a picture of one's self taken by one's self, typically at arm's length—is now easily accomplished with any smartphone, and often shared with others through social media. A 2013 poll indicated selfies accounted for one-third of photos taken within the 18-to-24 age group. Google estimated in 2014 that 93 billion selfies were taken per day just by Android users alone.[10] More recently, selfie taking has begun to influence human behavior in the physical world. Museums[26] have started to develop environments that cater specifically to Instagram and Snapchat users. Even facial plastic surgeons have observed an increase in the number of patients that seek plastic surgery specifically to look better in selfies (55% of surgeons had such patients in 2017, up 13% from 2016).[2] Perhaps most strikingly, plastic surgeons have begun reporting a new phenomenon termed "Snapchat dysmorphia," where patients seek surgery to adjust their fea-

» key insights

- Nowadays, anyone with a sufficiently powerful smartphone is able to make sophisticated edits to their photos and videos.
- By editing selfies, we change the way people perceive us and the way we perceive ourselves.
- The change can be positive or negative, and we must be mindful how we edit and evaluate photos.
- Researchers and software developers in the field should have a broad view that considers not only technology, but also cognitive science, psychology, and ethics.

tures to correspond to those achieved through digital filters.[28]

Photographs have long played a role in shaping our perception, and self-portraiture has existed almost as long as photography itself. Even early analog portrait photography offered powerful opportunities for personal identity formation and expression.[35] Digital photography built on these opportunities by providing new ways of capturing, disseminating, and editing personal photos. Camera-equipped smartphones greatly increased the number of people who could photograph themselves. Similarly, social media platforms amplified the ability to share personal portraits with others. Selfies represent a culmination of the personal and social dimensions of digital photography. Yet, while the selfie phenomenon demonstrated the ease of capturing and sharing self-portraits, until recently, the process of *editing* self-portraits has required extensive professional experience and skill.

This is beginning to change. A new class of digital photo manipulation technologies has begun to emerge—ones that enable complex, realistic, and *automatic* edits to digital portraits. The speed and ease offered by these new tools means that anyone with a sufficiently powerful smartphone is able to make sophisticated edits to their image. These editing technologies have implications, not only for the photos people share, but also for how the takers of those photos see themselves. As the Snapchat dysmorphia phenomenon illustrates, the act of editing one's selfie can change one's expectations for physical appearance in real life.

Our objective in this article is twofold: to provide an overview of state-of-the-art techniques for portrait manipulation, and to explore the implications of widespread use of these techniques on self-perception. In doing so, we seek to start a dialog on how potential consequences of these technologies—both positive and negative—should factor into decisions about how and why we choose to develop similar technologies in the future.

We discuss six categories of automated portrait editing technologies and the impact these approaches can have on self-perception. The ability to adjust the perspective and pose of a portrait will enable people to disguise the fact they have taken a selfie. Digital makeup suggests ways to increase self-esteem and one's professional appearance, but it could also increase the narcissistic perception of selfies in general. Facial adjustment algorithms offer ways to improve people's satisfaction with their digital portraits while also suggesting the potential to normalize certain facial proportions and

PHOTO BY DEAN DROBOT

features. Technologies for automatically swapping hair and wardrobe in photographs provide a new form of online identity exploration while also opening new risks for appropriation. Algorithms for shifting the age of a person’s photograph will enable people to selectively choose how old they appear for different contexts online and change peoples’ expectations for how they will look in the future. Techniques that turn still photos into video portraits can enhance photos with dynamic expressions, but the expressions might be taken from other people, raising questions about the authenticity of the emotions in the video. We follow with a discussion of the broader impacts of widespread portrait editing on media consumption, trust, and personal appearance.

Portrait Manipulation

Portrait photography is a complex process, for which many elements determine the final result. First and foremost, the subject of the photograph, their head pose and expression, their makeup and their clothes are all reflected in the photograph. Scene elements such as lighting conditions, camera location, and focus also play a substantial role. Capable photographers also take into account more “technical” details such as sensor sensitivity (ISO), aperture, and shutter speed in order to compose an effective shot.

With traditional print photography, these attributes of a portrait were largely baked into the photo at the time the shutter was closed. Afterward, skilled photographers could “dodge and burn” to locally modify exposure during printing, and for high-value shots like fashion photographs artists would even paint over a print using an airbrush in order to modify it. Today, with digital editing software like Adobe Photoshop, such operations are commonplace. But it may surprise some readers to learn that expression, makeup, pose, and even the *identity of the subject* can now, or in the near future, be easily modified in post processing, with no need for domain expertise or advanced image manipulation skills. Very soon, digital face and body editing will be as facile as Instagram filters are today—immediately accessible to anyone who can take a selfie.

Very soon, digital face and body editing will be as facile as Instagram filters are today—immediately accessible to anyone who can take a selfie.

Perspective and pose. Subtle details in how a photo is taken can have a substantial impact on how the subject of the photo is perceived by others. The distance between the camera and subject plays a key factor in perception. Faces imaged from closer distances appear to be more benevolent (good, peaceful, pleasant, approachable), while larger distances correlate with smart and strong appearance.[27] Furthermore, people rate photographs of faces taken from within personal space (that is, “too close”) as less trustworthy, competent, and attractive.[5] Selfies, one of the most prevalent forms of modern personal photography, are taken by definition at closer distances and exhibit noticeable perspective phenomena. As a result, while the convenience, affordability, and ease of selfies has allowed a broader range of people to participate in personal photography, the limits of photography at close distances means these same people are fundamentally constrained in the ways they can portray themselves to others.

We created a system that, given a single photograph as input, can virtually change the location of the camera to produce a new image, with different perspective.[13] Our system produces photorealistic results through a combination of 2D and 3D techniques. We use commonalities in the appearance of heads to estimate the photo’s 3D structure, and then move pixels around on the 2D image plain to produce the final result. This approach allows for arbitrary pose changes, and the creation of 3-dimensional heads from 2-dimensional photos (Figure 1). The estimated 3D model is a rather weak approximation of the true head shape but is enough to describe a convincing 2D warp that produces a realistic result. This 2D-3D hybrid approach has also proved successful for other face manipulation tasks such as expression transfer.[39] As capture hardware and algorithms improve, we expect better 3D models from single or multiple photos, which will further improve 3D-based photo editing.

Automatic perspective adjustment will allow selfie takers to distinguish between the impacts of the camera lens, angle, and distance, and their actual facial proportions. Individuals who assume that they have undesirable facial characteristics can now view their pictures from multiple perspectives and get a more accurate sense of how their faces appear to others. These techniques will also increase the expressiveness of the selfie as a tool for self-presentation. From minor changes, such as shifting ones' pose to a more attractive angle, to major changes like adjusting the perspective of one's face to appear more competent and intelligent for a LinkedIn profile, more people will be able to make perspective adjustments that align with how they wish to be perceived in different online environments.

Makeup. Physical makeup can alter our own perceptions of ourselves, and also change how others see us. Bloch and Richins demonstrated that makeup can temporarily increase the wearer's self-esteem[4] and Etcoff et al. showed that people wearing minimal amounts of physical makeup are often perceived as more likeable and competent.[11] Today, makeup use has become prevalent among the general public as cosmetic products have become cheaper and more widely available.[17] Yet successfully applying physical makeup can be difficult—requiring skill in both selecting the right products and applying them correctly.

Since the advent of portrait photo editing, makeup has also been applied to photos—first through physical retouching methods and later through digital tools like Photoshop. Like physical makeup application, digital makeup creation has, until recently, also required specialized skill and expertise. Recent developments in automated portrait editing have greatly lowered the effort necessary to apply digital makeup. In one example, we introduced a system that can apply and remove makeup.[9] Given a pair of photos—a *source* photo s without makeup and a *reference* photo $\bar{r}$ showing a makeup style—we automatically generate a new picture $\bar{s}$ showing s wearing makeup in the style of $\bar{r}$ (Figure 2). The approach leverages recent advances in image style transfer based on deep learning. As is typical in machine learning projects, a good data set is essential. However, for this project it would be very difficult to acquire ground truth triplets $(s, \bar{r}, \bar{s})$. Our approach instead learns two functions: makeup transfer function $T(s, \bar{r}) \rightarrow \bar{s}$, and makeup removal function $R(\bar{r}) \rightarrow r$ that can remove makeup. The key insight that permits us to train these functions is that we can actually apply them twice sequentially, yielding the original image pair. This insight relies on the observation that $T(r, \bar{s}) \rightarrow \bar{r}$ and $R(\bar{s}) \rightarrow s$. This allows us to train with image pairs of different people.

Given the impacts that physical makeup has on self-image, it is likely that digital makeup will also have an effect on how we view ourselves. The professional edge offered by physical makeup is arguably easier to attain

Figure 1. Given a single input photograph (a) we can change perspective and pose. We remove the "selfie effect" caused by a short camera-to-subject distance (b), rotate the head (c), and create 3D anaglyphs from a 2D photo (d, use red-cyan glasses to view).

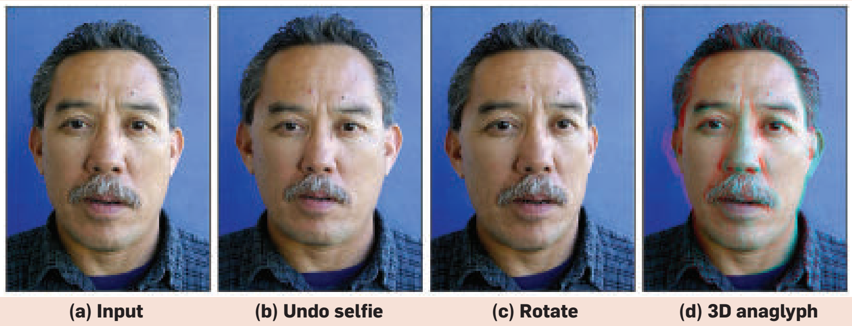

(a) Input
(b) Undo selfie
(c) Rotate
(d) 3D anaglyph

Figure 2. Source photos (top row) are each modified to match reference makeup styles (left column) to produce nine different outputs (3 × 3 lower right).[9]

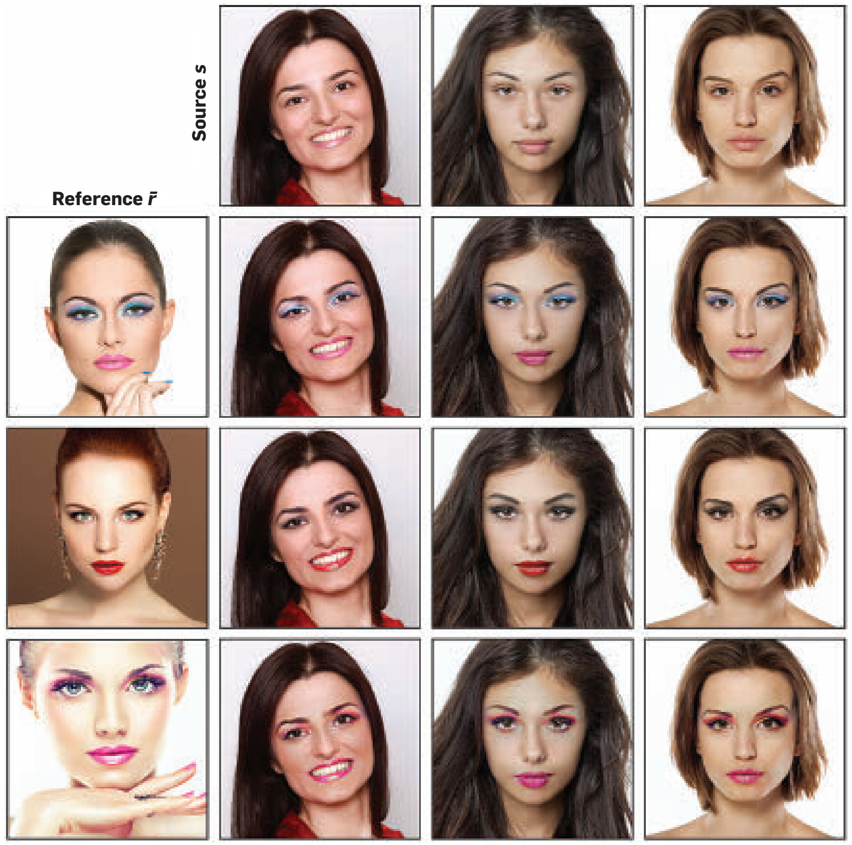
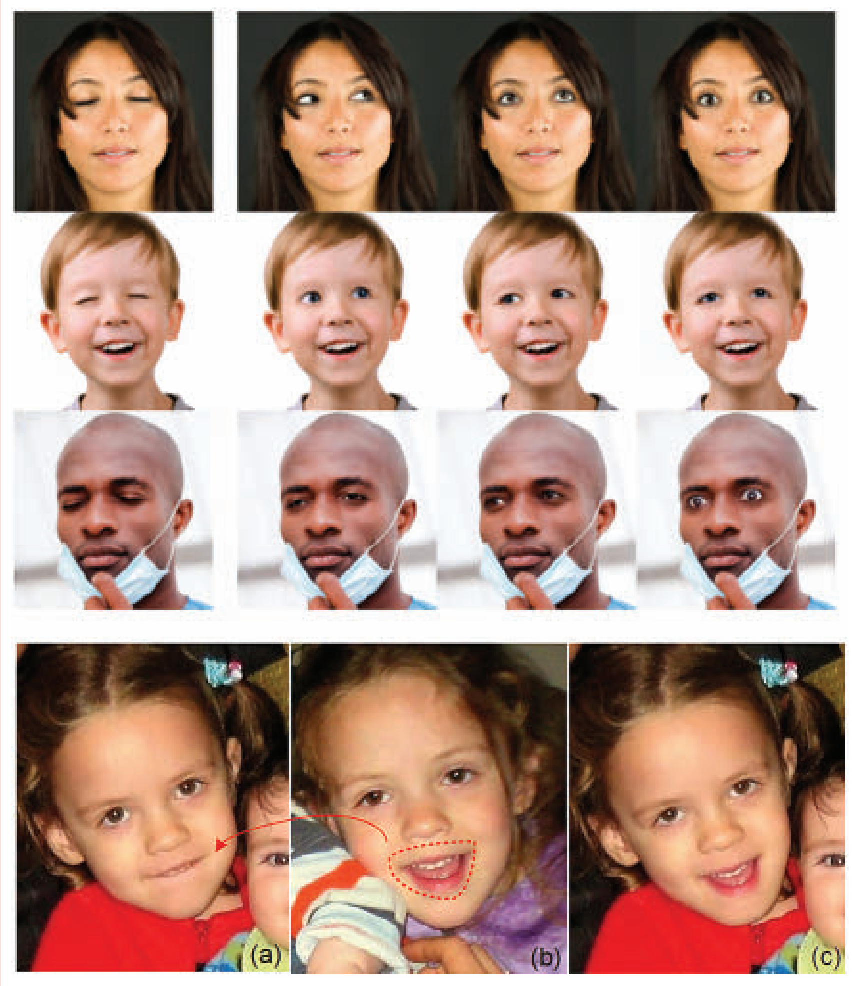


Figure 3. Top: Closed eyes (left) are automatically opened (right, showing three example results).[32] Bottom: a photos with no smile (a) is enhanced by using a smile from another photo (b) to create a final composition (c).[39]



(for digital contexts) through automated makeup filters. These same filters may offer smartphone users quick self-esteem boosts at the touch of a button. More broadly, the ease of digital makeup transfer will make it easy for people to experiment with a range of different makeup styles. This flexibility could have multiple benefits. It suggests an opportunity for more people to engage in playful experimentation with their appearance and build confidence in their online portrayal. Furthermore, digital makeup could provide an opportunity to preview an effect before investing the time and money to recreate it in real life. The benefits of moderate amounts of physical makeup suggest that automated makeup filters may lower the threshold for presenting oneself as competent and confident when online. Conversely, large amounts of makeup, while increasing attractiveness, can also lead to perceptions that a person is untrustworthy or narcissistic.[11] People already view posting selfies as a narcissistic act,[10] therefore increasing prevalence of digital makeup in selfies may perpetuate negative attitudes towards selfie takers.

Facial features. The shape and relative location of facial features define how we look. Characteristics such as a pointy nose, big eyes or an elongated face are all derived from facial features. Some features can be changed at will (a smile), some can be changed over time (a skinny face) and some are tightly coupled with bone structure (weak jaw). In the physical world, the latter can only be changed via plastic surgery, and not all results are achievable.

Until recently, major digital edits to the face, like reshaping the eyes and nose, required substantial skill and knowledge. Whereas previous digital editing paradigms required users to select from low-level, general tools like digital paintbrushes, and skillfully apply the tools to produce believable results, new automated digital approaches make it possible to immediately transform individual features with believable results. Unwanted eye-blinks and sideways glances are common in photos of individuals, and even more likely to appear in group photos. Shu et al.[32] automatically edit eyes in photographs by leveraging a user's personal photo collection. They find good reference eyes in the personal collection and transfer them to the target (Figure 3 top). Transferring features between photos is not limited to eyes, nor to portraits.[1] Yang et al.[39] transfer facial expressions from one photo to another (Figure 3 bottom). In addition to making local edits to the target feature, this method has the important effect of also enacting subtle adjustments to adjacent features and face shape.

An alternative approach holistically considers all face features simultaneously. Leyvand et al.[23] created a data-driven technique for face beautification. Their system is trained to warp images so the relative location of facial features matches images of faces that people rated as more appealing. The warp is trained to stay close to the input and users can adjust the modification amount, resulting in portraits that preserve characteristics of the original face. However, because Leyvand's approach does not use a physically based model, it can produce facial transformations that are either impossible, or would require extensive facial surgery to achieve in real life. Leyvand's approach is also distinguished from Shu and Yang's; rather than optimizing portraits based on features drawn from images of the same person, Leyvand's algorithm adjusts images according to optimum derived from images of other people. The automated nature of facial-feature editing also means that facial editing can now be directly integrated into the camera viewfinder.[24] In some cases, a suite of effects is applied by default in real time, meaning that from the moment the user opens the application, they are presented with an adjusted image of their face.

The integration of automated facial adjustment algorithms with personal cameras will affect our perception of self attractiveness. Dissatisfaction with aspects of one's appearance is part of being human, and cultural beauty ideals existed well before digital photography. In one sense, tools that enable people to optimize their portraits by combining personal images are poised to broaden the range of people who can produce photos that represent them at their best. The use of tools that adjust facial features according to the photos of others presents a less clear-cut outcome with regards to self-image. Flipping between an untouched image of their face, and one adjusted to some external standard could lead to people identifying "flaws" in their appearance that they were previously unaware of.

Hess argues that beautification filters create a situation where people compare their image to an idealized version of themselves, rather than to external ideals like celebrities or models.[16] The before and after comparison afforded by these technologies may also refine people's understanding of how far their individual facial features are from an idealized norm. Rather than having a vague sense that one's chin is too big, a person can now immediately see how small an algorithm thinks their chin should be. All algorithms, by definition, contain built-in biases determined either by the preferences of the algorithm designers or by the data used to train the algorithm. Whereas previously beauty norms were influenced by people in the fashion and marketing industries, new norms will be determined by the algorithms themselves. It's important to recognize that, like human biases, algorithmic biases can unfairly discriminate against minority groups and can reinforce or amplify existing racial and gender stereotypes.[6]

Age. Age shapes both how we perceive others, and the way we perceive ourselves. Age can affect attitudes toward a person's competence as demonstrated in one study where younger raters rated older workers as less qualified and as having less potential for development in comparison to younger workers.[12] Age also affects how we perceive attractiveness. Culturally, we often associate beauty with youth, identifying attractive people as younger than they actually are, or characterizing young people as more beautiful than older people.[22] Until now, personal photos have primarily reflected the physical age of the person relative to date they were taken. This quality has largely defined the role portraits have served in family life, by providing a way to document family members' age progression over time and mark key moments of coming of age. The act of reviewing personal portraits from different stages in one's life plays an important function in personal commemoration and memory.[35]

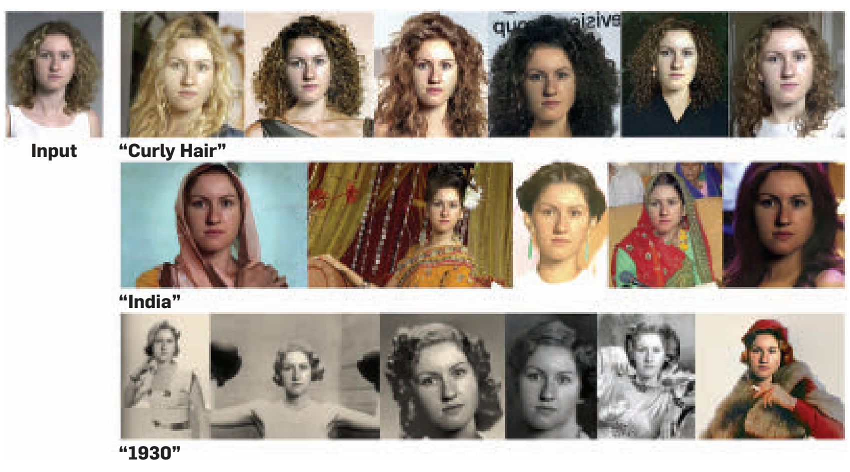
Altering the age of a person in a digital photo, even by a few years, is a difficult task. A person's future self depends on their current appearance, but also on invisible genetic traits and unforeseeable environmental conditions. Nevertheless, recent tools for automatic age adjustment have emerged that make it feasible for anyone to make extreme shifts in age of their portrait. Most notably, Kemelmacher-Shlizerman et al.[20] use a large dataset of photos of various ages to calculate typical differences between age groups. They then apply the differences to a new photo of a baby, producing age-progressed result from childhood to old age.

Automatic age adjustment fundamentally broadens the nature of personal photography. Whereas photos previously served as a tool to document a person's appearance at a specific moment, they will now provide a starting point for projecting how a person looks across multiple points in time. Photographic age will become something anyone can actively manipulate and control in a digital context. Just as people currently falsify their age on online dating sites to appear more desirable,[15] people will now be able to alter their photographic age to appear more attractive, professional, mature, or youthful, depending on the context. Automated portrait aging will also affect young people in important ways. Children who transform their own portraits will have a different understanding of how their appearance will change as they grow older. They will be able to preview the effects of aging immediately, rather than experience them gradually over time, and have different expectations about how their features will change as they age.

This technology could also be used to motivate lifestyle change. With the right data, we could present alternate futures. A person could forecast how they might look in 10 years if they engage in healthy behaviors like regular exercise, or harmful behaviors like smoking.

Hair, wardrobe, and style. In the physical world, people experiment

Figure 4. Given an input photo and a target style (text string), the system of Kemelmacher-Shlizerman[19] automatically retrieves Internet photos and swaps faces to produce the input person in the target style.



with different hairstyles and clothing choices to express different aspects of their identity. Psychologists have theorized that, particularly for younger people, low-risk experimentation with self-presentation can serve an important role in personality development. Digital communities have acted as an extension for physical forms of identity experimentation by providing a virtual environment where people can inhabit different avatars, or present different personas in online communities.[34]

Today people can also experiment with their wardrobe and hairstyle of their digital self-portraits. Kemelmacher-Shlizerman[19] introduced a system to automatically swap the face of an existing photo with a target portrait (Figure 4). The inputs to the system are a photo of a person and a search term, such as "curly hair" or "1930." The system retrieves Internet photos that match the search term and blends the input face to the Internet photos. The result is a photo with a style that matches the search term, containing the given face. The key here is that styles are often determined by hair or clothing, thus we can swap faces without drastically changing style.

In some ways, the ability to digitally alter our clothing and hairstyle offers a new channel to extend benefits of fashion experimentation in the physical world by providing people with an easier, faster, and cheaper method to try out different looks. Furthermore, because photo-based methods of hair and clothing transfer enable styles to be transferred from photos found via Internet search, and applied to images of an actual person, rather than an avatar, this technique could avoid some of the limitations imposed by avatar based systems where either system designers or skilled users have control of the range of options available to users.[18]

This approach also has important constraints that can affect the self-perception of the people who use it. The facial transfer algorithm works best for images with similar looking faces. The effectiveness of a search for "movie star" or "scientist" will reflect the range and number of online images in these categories that most closely match the gender and ethnicity of the user, thereby reflecting and reproducing established trends and biases in online photo repositories. A similar issue emerged when Google released an app that matched people with similar faces within classical art and many non-white users found themselves matched with artworks reflecting racial stereotypes.[31]

Hairstyle and clothing transfer also have broader cultural and political implications for how we present and perceive identity. In countries with racial and ethnic diversity, trends in fashion often intersect with social tensions like racial stereotyping and cultural appropriation. Stereotyping and cultural appropriation in the real world can reduce self-esteem among disenfranchised minority groups who experience it.[14] Digital techniques that enable people to experiment with clothing, hairstyles, and albeit unintentionally, skin-tone, from photographs will dramatically increase opportunities for people to represent themselves with styles of other subcultures. While this could prove empowering for the people doing the experimenting, it could have the opposite effect for the minority groups whose cultural styles are appropriated.

Video portraits. All the methods introduced thus far operate on photos—a moment frozen in time. Similar elements determine how we look in videos, with an added temporal dimension. For example, a smile is no longer just one photo taken at the apex of the smiling process, but a *trajectory* of motion, starting with a hint and ending with an ear-to-ear smile.

Figure 5. The method of Averbuch-Elor et al.[3] can create moving portraits from still photos. Given a single input photograph (top) and a reference video (not shown), a new video is created with dynamic expressions from the reference (selected frames shown, bottom 3 rows).

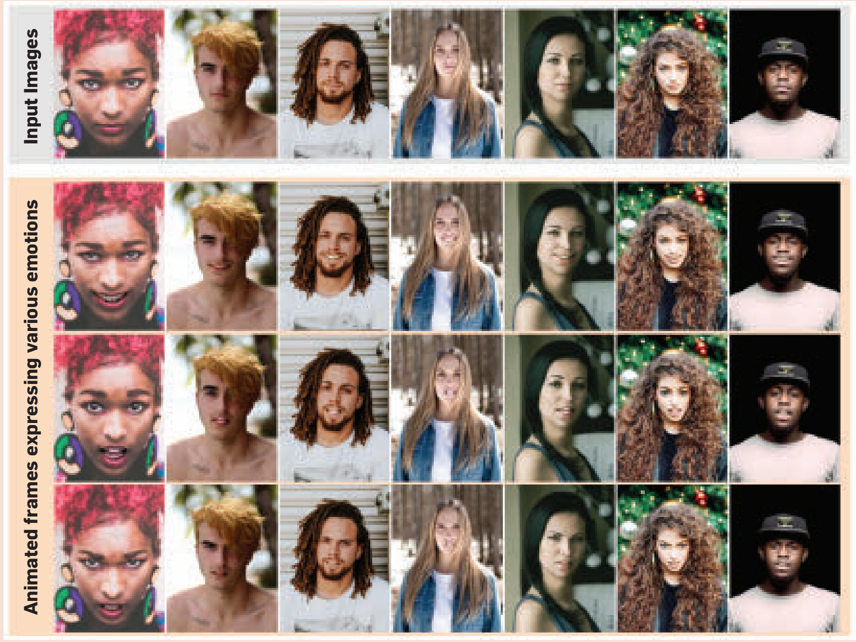
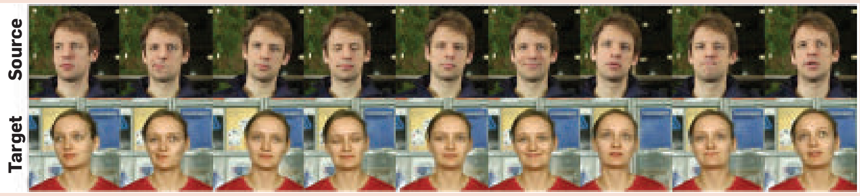


Figure 6. Deep Video Portraits[21] transfer pose, expression and eye gaze from a source video (top) to a target video, producing convincing results (bottom). Resulting frames are generated by the method, and need not appear in the original video.



When considering videos, the added temporal dimension introduces both opportunity and challenge. Moving portraits can be more expressive, but more difficult to produce and manipulate compared to static photos. Averbuch-Elor et al.[3] introduced a method that can animate an input photo, producing results akin to the moving portraits in the Harry Potter series (Figure 5). They took upon themselves the challenge of using only a *single* input photo of the person to animate. Their key contribution is in finding a way to transfer another person's motion to the input photo, producing compelling results that can be applied to both current photos and historic figures, for which video footage is unavailable. Interestingly, since the driving video is of a different person, the result might couple the facial appearance of one person with the mannerisms of another.

Instead of limiting the input to a single photo, other methods try to learn what a person looks like in a video, and use that knowledge to generate synthetic head motion, expressions, and speech. Deep Video Portraits[21] puppeteer one person using a video of another, allowing control over head pose, expressions, and eye gaze (Figure 6). They train a neural network to convert synthetic head renderings to a photo-realistic video frame. They then perform puppeteering by rendering heads with the identity of one person and other parameters (pose, expression) of another, producing the final video using their neural network. Input modalities other than head renderings can also be converted to video portraits. Wang et al.[36] show sketch-to face video results, allowing a few brush strokes to control facial appearance. Suwajanakorn et al.[33] convert an audio speech to a video of a person giving that speech. Improved controls for dynamic faces remains an opportunity for future research.

The emergence of automated video manipulation algorithms will make editing videos of our faces and bodies ubiquitous. At present, much of the attention on algorithmic video synthesis focuses on the risks this technology poses for information falsification, concerns we discuss later. However, it is also important to recognize the impact that ubiquitous video editing will have on self-perception. Each technique we described—adjusting pose, makeup, facial features, age, and style—will be adapted for video. Moreover, we will be able to alter temporal expressions of emotion. People may choose to amplify the emotional quality of a video, for example editing a karaoke video to correspond with the posture and poise of a professional pop star. Or, they may choose to replace the recorded emotions, swapping the disapproving head shake of a relative in a home movie with a nod and a smile.

The emergence of automated video manipulation algorithms will make editing videos of our faces and bodies ubiquitous.

Implications

As we demonstrate, most, if not all portrait elements can be digitally manipulated. A person in a photo might, in real life, be older, or have a different facial structure. A photograph of a person in an exotic location may, in reality, portray someone who never left their hometown. If the subject is moving, that does not mean that a real video was ever captured.

These forms of photo manipulation were possible before the development of the techniques we describe. More than 20 years ago the special effects team of *Forrest Gump* (1994) were able to create convincing videos of the movie's eponymous protagonist sitting with John Lennon and shaking hands with President Kennedy. More recently, the actor Paul Walker was digitally inserted into *Furious 7* scenes after his death (2015), and a young version of Arnold Schwarzenegger appeared in *Terminator Genisys* (2015). In fact, manipulation in the movie industry is now commonplace, producing convincing virtual characters or digitally de-aging famous actors. The important difference between visual effects in mainstream movie production and techniques presented here is *the amount of labor and expertise necessary to achieve them*. Rapid, automatic methods for portrait editing will broaden the range of people who can use these techniques, extend the domains and contexts in which they will be applied, and amplify impacts that digital manipulation has on self-image as a whole.

Democratization vs. distortion. As individuals, and as a society, we should strive to judge people for qualities beyond how they look. Unfortunately, at present, our physical appearance measurably impacts how we are treated by others. As we reflect on ways to change this, we must also recognize the desire to reshape personal appearance is a reasonable response in a world where beauty standards still exist. Moreover, the growing presence of social media and the Internet in daily life has created new expectations for how we present ourselves digitally, and new consequences for failing to adhere to cultural appearance standards. From this viewpoint, the democratization of tools to alter our digital appearance is important for individual empowerment. Yet making it easier to modify one's digital self will increase the number of manipulated portraits people encounter overall. This, in turn, could increase dissatisfaction with one's physical appearance and amplify the pressure to change it.

Take for example the interaction between social media use and adolescent body image. Salomon and Brown[29] found that self-objectifying social media use predicted greater body shame among youth. Looking specifically at photo editing, McLean et al.[25] found an association between self-photo editing and body dissatisfaction in adolescent girls. One explanation for this connection is that people who are already dissatisfied with their bodies naturally look for opportunities to digitally edit their online image. If true, this suggests that portrait editing tools can be empowering. They are a response to flawless fashion spreads, allowing everyone to compete in an ultra-Photoshopped society. This connection between photo editing and negative body image might also lead to an alternate conclusion: that the ability to edit one's photos can increase body dissatisfaction by highlighting the gap between reality and the perceived ideal.

In the physical world, people must often walk a difficult line between being perceived as putting adequate effort into one's appearance versus being perceived as deceptive. Similar challenges will present themselves when relying on digital forms of portrait manipulation. Algorithms that perform subtle adjustments may be more socially acceptable than those that produce realistic but dramatic differences between photo and reality. People who choose to substantially alter their appearance digitally may learn to portray such behavior as playful in an effort to avoid being seen as inauthentic or narcissistic.

Algorithms that perform subtle adjustments may be more socially acceptable than those that produce realistic but dramatic differences between photo and reality.

Synthesized storytelling. Automated portrait editing may also change the ways mainstream media delivers information to the public. News outlets have begun to experiment with virtual anchors to deliver news.[38] The press release stated: "[The virtual anchor] has become a member of its reporting team and can work 24 hours a day on its official website and various social media platforms, reducing news production costs and improving efficiency." Virtual anchors are still experimental, and it is not clear if an audience will find them engaging or trustworthy. Yet the potential advantages of such techniques are abundant; unlike traditional recording, synthesized anchors would enable dynamic changes to the news report to correct mistakes, translate a story into multiple languages, or respond on the fly as updates emerge. Such advantages could also transfer into other forms of information delivery including education and professional training.

Concerns over media manipulation are at a peak in many parts of the world, and the prospect of synthetic video has exacerbated fears that malicious actors will be able to deceive the public more easily.[7] Given these concerns, and the fact synthetic video is one method among many existing means to manipulate information, it is useful to unpack the specific issues of video synthesis from the broader challenges of media falsification. The forms of portrait editing we describe in this article will undeniably expand the range of people who, if they choose to do so, can generate malicious false video content. It is the responsibility of the researchers who develop such technologies, ourselves included, to acknowledge this fact, and weigh the risks and benefits of developing such algorithms as we proceed in this research.

At the same time, the consequences of any malicious media creation, fake video or otherwise, are shaped by many different factors. Human editorial decisions, social media and search algorithms, and individual patterns of consumption determine the content people see. Cultural and political alignments, religion, education, family history, and many other complex factors shape what forms of media different people choose to trust. In our increasingly media-rich world, addressing the challenge of fake content will require systematic efforts to enact policy for how content is created, manipulated, and distributed. We must also get people to think critically about the media they see. There is already evidence that people have difficulty distinguishing between different types of media content—for example, an ad versus a news story.[37] Distinguishing between "real" and manipulated photographs may pose an even greater challenge. This paper is one attempt to address this challenge by demystifying the state of the art in portrait manipulation. A broader solution might involve augmenting media studies curriculum, or even general education, with image processing techniques and algorithm design.

Conclusion

We have outlined emerging technologies for manipulating our facial structure, expressions, hair, makeup, clothing, and age, using state-of-the-art image and video synthesis methods. At an individual level, these techniques can enable one person to quickly and easily change their appearance. On a collective level, however, these technologies will fundamentally change the ways in which people present themselves to one another. As researchers continue to develop new technologies for manipulating the human face, it is critical to consider the magnitude of these changes and their impact on others. This requires considering the biases inherent in the data we rely on to drive these technologies. It necessitates constantly evaluating consequences of such technologies and be aware of the potential for unintentional harm. As we develop tools that are easier to use, we must also consider how automatically limiting some choices and enabling others will encourage some forms of self-expression and discourage others. One thing is clear, these technologies are bound to change the *face* of society.

References

1. Agarwala, A. et al. Interactive digital photomontage. *ACM Trans. Graphics 23*, 3 (2004), 294–302.
2. American Academy of Facial Plastic and Reconstructive Surgery. Annual Survey Unveils Rising Trends In Facial Plastic Surgery (2017); https://www.aafprs.org/media/stats_polls/m_stats.html
3. Averbuch-Elor, H., Cohen-Or, D., Kopf, J., and Cohen, M.F. Bringing portraits to life. *ACM Trans. Graph. 36*, 6, Article 196 (Nov. 2017); https://doi.org/10.1145/3130800.3130818
4. Bloch, P.H. and Richins, M.L. You look 'mahvelous:' The pursuit of beauty and the marketing concept. *Psychology & Marketing 9*, 1 (1992), 3–15; https://doi.org/10.1002/mar.4220090103
5. Bryan, R., Perona, P., and Adolphs, R. Perspective distortion from interpersonal distance is an implicit visual cue for social judgments of faces. *PloS one 7*, 9 (2012), e45301.
6. Buolamwini, J. and Gebru, T. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Proceedings of Conf. on Fairness, Accountability and Transparency*, (2018), 77–91.
7. BuzzFeed. You Won't Believe What Obama Says In This Video!, 2018; https://www.youtube.com/watch?v=cQ54GDm1eL0
8. Cao, C., Weng, Y., Zhou, S., Tong, Y., and Zhou, K. Face-warehouse: A 3D facial expression database for visual computing. *IEEE Trans Visualization and Computer Graphics 20*, 3 (2014), 413–425; https://doi.org/10.1109/TVCG.2013.249
9. Chang, H., Lu, J., Yu, F., and Finkelstein, A. Pairedcyclegan: Asymmetric style transfer for applying and removing makeup. *Proceedings of 2018 IEEE Conf. on Computer Vision and Pattern Recognition.*
10. Diefenbach, S. and Christoforakos, L. The selfie paradox: Nobody seems to like them yet everyone has reasons to take them. An exploration of psychological functions of selfies in self-presentation. *Frontiers in Psychology 8* (2017), 7; https://doi.org/10.3389/fpsyg.2017.00007
11. Etcoff, N.L., Stock, S., Haley, L.E., Vickery, S.A., and House, D.A. Cosmetics as a feature of the extended human phenotype: Modulation of the perception of biologically important facial signals. *PLOS ONE 6*, 10 (2011), 1–9; https://doi.org/10.1371/journal.pone.0025656
12. Finkelstein, L.M., Burke, M.J., and Raju, M.S. 1995. Age discrimination in simulated employment contexts: An integrative analysis. *J. Applied Psychology 80*, 6 (1995), 652.
13. Fried, O., Shechtman, E., Goldman, D.B., and Finkelstein, A. Perspective-aware manipulation of portrait photos. *ACM Trans. Graph. 35*, 4 (July 2016), 128:1–128:10; https://doi.org/10.1145/2897824.2925933
14. Fryberg, S.A., Markus, H.R., Oyserman, D., and Stone, J.M. Of warrior chiefs and Indian princesses: The psychological consequences of American Indian mascots. *Basic and Applied Social Psychology 30*, 3 (2008), 208–218. https://doi.org/10.1080/01973530802375003arXiv:https://doi.org/10.1080/01973530802375003
15. Hancock, J.T., Toma, C., and Ellison, N. The Truth About Lying in Online Dating Profiles. In *Proceedings of the 2007 SIGCHI Conf. Human Factors in Computing Systems*. ACM, New York, NY, USA, 449–452; https://doi.org/10.1145/1240624.1240697
16. Hess, A. The ugly business of beauty apps. *The New York Times* (2017); https://nyti.ms/2O4deuK
17. Jones, G. Globalization and beauty: A historical and firm perspective. *EurAmerica 41*, 4 (Dec. 2011), 885–916.
18. Kafai, Y.B., Cook, M.S., and Fields, D.A. 'Blacks deserve bodies too!' Design and discussion about diversity and race in a tween virtual world. *Games and Culture 5*, 1 (2010), 43–63; https://doi.org/10.1177/1555412009351261.
19. Kemelmacher-Shlizerman, I. 2016. Transfiguring portraits. *ACM Trans. Graph. 35*, 4, Art. 94 (July 2016); https://doi.org/10.1145/2897824.2925871.
20. Kemelmacher-Shlizerman, I., Suwajanakorn, S., and Seitz, S.M. Illumination-aware age progression. In *Proceedings of the IEEE Conf. Computer Vision and Pattern Recognition*. 2014, 3334–3341.
21. Kim, H. et al. Deep video portraits. *ACM Trans. Graph. 37*, 4, Art. 163 (July 2018); https://doi.org/10.1145/3197517.3201283
22. Kwart, D.G., Foulsham, T., and Kingstone, A. Age and beauty are in the eye of the beholder. *Perception 41*, 8 (2012), 925–938; https://doi.org/10.1068/p7136.
23. Leyvand, T., Cohen-Or, D., Dror, G., and Lischinski, D. Data-driven enhancement of facial attractiveness. *ACM Trans. Graph. 27*, 3, Art. 38 (Aug. 2008); https://doi.org/10.1145/1360612.1360637
24. Lightricks LTD. 2013–2018. Facetune. https://www.facetuneapp.com/
25. McLean, S.A., Paxton, S.J., Wertheim, E.H., and Masters, J. Photoshopping the selfie: Self photo editing and photo investment are associated with body dissatisfaction in adolescent girls. *Intern. J. Eating Disorders 48*, 8 (2015), 1132–1140.
26. Pardes, A. The rise of the made-for-instagram museum. *Wired* (Sept. 2017); https://www.wired.com/story/selfie-factories-instagram-museum/
27. Perona, P. A new perspective on portraiture. *J. Vision 7*, 9 (2007), 992–992.
28. Rajanala, S., Maymone, M.C., and Vashi, N.A. Selfies—living in the era of filtered photographs. *JAMA Facial Plastic Surgery* (2018); https://doi.org/10.1001/jamafacial.2018.0486
29. Salomon, I. and Brown, C.S.O. The selfie generation: Examining the relationship between social media use and early adolescent body image. *J. Early Adolescence*; https://doi.org/10.1177/0272431618770809
30. Saragih, J.M., Lucey, S., and Cohn, J.F. Face alignment through subspace constrained mean-shifts. In *Proceedings of IEEE 12th Intern. Conf. Computer Vision*. IEEE, 2007, 1034–1041.
31. Shu, C. Why inclusion in the Google Arts & Culture selfie feature matters, 2018; https://tcrn.ch/34UovEz.
32. Shu, Z., Shechtman, E., Samaras, D., and Hadap, S. EyeOpener: Editing eyes in the wild. *ACM Trans. Graph. 36*, 1, Art. 1 (Sept. 2016); https://doi.org/10.1145/2926713
33. Suwajanakorn, S., Seitz, S.M., and Kemelmacher-Shlizerman, I. Synthesizing Obama: Learning lip sync from audio. *ACM Trans. Graph. 36*, 4, Art. 95 (July 2017); https://doi.org/10.1145/3072959.3073640
34. Turkle, S. The Second Self: Computers and the Human Spirit. MIT Press, Cambridge, MA, 2005; https://books.google.com/books?id=UVXtBAAAQBAJ
35. van Dijck, J. Digital photography: communication, identity, memory. *Visual Communication 7*, 1 (2008), 57–76; http://bit.ly/32J4X4y
36. Wang, T. et al. Video-to-video synthesis, 2018; arXiv preprint arXiv:1808.06601
37. Wineburg, S., McGrew, S., Breakstone, J., and Ortega, T. Evaluating information: The cornerstone of civic online reasoning. Stanford Digital Repository, 2016. Accessed Jan. 8, 2018.
38. Xinhua News Network. Xinhua's first English AI anchor makes debut, 2018; https://www.youtube.com/watch?v=GAfiATTQufk
39. Yang, F., Wang, J., Shechtman, E., Bourdev, L., and Metaxas, D. 2011. Expression flow for 3D-aware face component transfer. *ACM Trans. Graph. 30*, 4, Art. 60 (July 2011); https://doi.org/10.1145/2010324.1964955

Ohad Fried (ohad@stanford.edu) is a postdoctoral research scholar at Stanford University, Stanford, CA, USA.

Jennifer Jacobs (jmjacobs@ucsb.edu) is an assistant professor of media arts and technology and director of the Expressive Computation Lab at the University of California at Santa Barbara, CA, USA.

Adam Finkelstein (af@cs.princeton.edu) is a professor of computer science at Princeton University, Princeton, NJ, USA.

Maneesh Agrawala (maneesh@cs.stanford.edu) is the Forest Baskett Professor of Computer Science and director of the Brown Institute for Media Innovation at Stanford University, Stanford, CA, USA.

DOI:10.1145/3371906

Exploring the vision of a model-based framework that may enable broader engagement with and informed decision making about sustainability issues.

Toward Model-Driven Sustainability Evaluation

JÖRG KIENZLE
McGill University

GUNTER MUSSBACHER
McGill University

BENOIT COMBEMALE
University of Toulouse and Inria

LUCY BASTIN
Aston University

NELLY BENCOMO
Aston University

JEAN-MICHEL BRUEL
University of Toulouse

CHRISTOPH BECKER
University of Toronto

STEFANIE BETZ
Furtwangen University

RUZANNA CHITCHYAN
University of Bristol

BETTY H.C. CHENG
Michigan State University

SONJA KLINGERT
University of Mannheim

RICHARD F. PAIGE
McMaster University

BIRGIT PENZENSTADLER
Chalmers University of Technology

NORBERT SEYFF
FHNW and University of Zurich

EUGENE SYRIANI
Universite de Montreal

COLIN C. VENTERS
University of Huddersfield

SUSTAINABILITY—THE CAPACITY TO endure—has emerged as a concern of central relevance for society. However, the nature of sustainability is distinct from other concerns addressed by computing research, such as automation, self-adaptation, or intelligent systems. It demands the consideration of environmental resources, economic prosperity, individual well being, social welfare, and the evolvability of technical systems.[7] Thus, it requires a focus not just on productivity, effectiveness, and efficiency, but also the consideration of longer-term, cumulative, and systemic effects of technology interventions, as well as lateral side effects not foreseen at the time of implementation. Furthermore, sustainability includes normative elements and encompasses multi-disciplinary aspects and potentially diverging views. As a wicked problem

IMAGE BY RANGIZZZ/SHUTTERSTOCK

(see the sidebar "Wicked Problems"), it challenges business-as-usual in many areas of engineering and computing research.

The complexity of these integrated techno-socioeconomic systems and their interactions with the natural environment is driving attention in several areas. These areas include means for understanding the emergent dynamics of these interactions and supporting better decision making through predictive simulation and system adaptation. At the heart of this is the notion of a model, an abstraction created for a purpose. Models are used throughout sustainability research (for example, for hydrology or pollution analysis) as well as software engineering (for example, for automated code generation). Models have a long history in research related to sustainability. The Global Modeling (GM) initiatives that started in 1960s and 1970s developed and used large mathematical dynamic global models to simulate large portions of the entire world.[13] GM in general was applied to human decision-making in domains such as economics, policy, defense, minimization of poverty, and climate change. The goal of GM is to offer a prediction of the future state of the world, or parts of it, using (perhaps heavily) mathematical equations and assumptions. Mathematical models offer a framework of stability that is useful in domains such as climate modeling, but it may not be the same in the case of social sciences domains.

In GM, several models can be seen as "modules" of a larger one, where outputs of one model are inputs for other(s) model(s). This vision of modularity was perhaps very advanced for its time. The idea of building models of complex systems based on simpler models has progressed enormously in the engineering domains, software engineering included. However, in the areas of social and natural sciences it is not the case.[12] The intention of initiatives related to GM, for example, International Futures[a] or the GLOBIOM model,[b] have common qualities shared by our proposal. However, GM

[a] http://pardee.du.edu
[b] http://www.globiom.org

» key insights

- **Recent progress in scientific modeling, model-driven engineering, and data curation can be used to support decision-making about sustainability issues.**
- **The conception of a sustainability evaluation and decision-support system capable of running what-if sustainability scenarios requires the collaboration of disciplines in CS and beyond.**
- **Emerging challenges include dealing with open-world contributions, uncertainty, and conflicting worldviews.**

did not present software engineering practices as a relevant aspect, partly due to the state of software engineering in those years.

Model-driven engineering (MDE) advocates the use of models that are successively refined and help analyze system properties. This article addresses a question at the intersection of MDE and sustainability research: How can we better support and automate sustainability by bringing together models, data, visualization, and self-adaptive systems to facilitate better engagement, exploration, and understanding of the effects that individuals' and organizational choices have on sustainability? We addressed this question with members from the MDE, sustainability design, and sustainability modeling communities, building on earlier contributions.[17]

The article conducts a focused review of converging research in MDE, data integration, digital curation (see the sidebar "Digital Curation"), public engagement, and self-adaptive systems with the perspective of sustainability as a driving motivation. We draw upon a vision of a highly capable integrated environment that facilitates integration of models and data from multiple diverse sources and visual exploration of what-if and how-to scenarios for multiple constituencies. This lens is especially effective for such a review due to its central relevance and urgency, but also because of the massively heterogeneous nature of data required to understand sustainability. We note the limitations of existing approaches and the common assumptions around reductionist modeling perspectives, quantification of uncertainty, and resolution of conflicts and contradictions. These issues are leveraged to identify and characterize emerging research challenges.

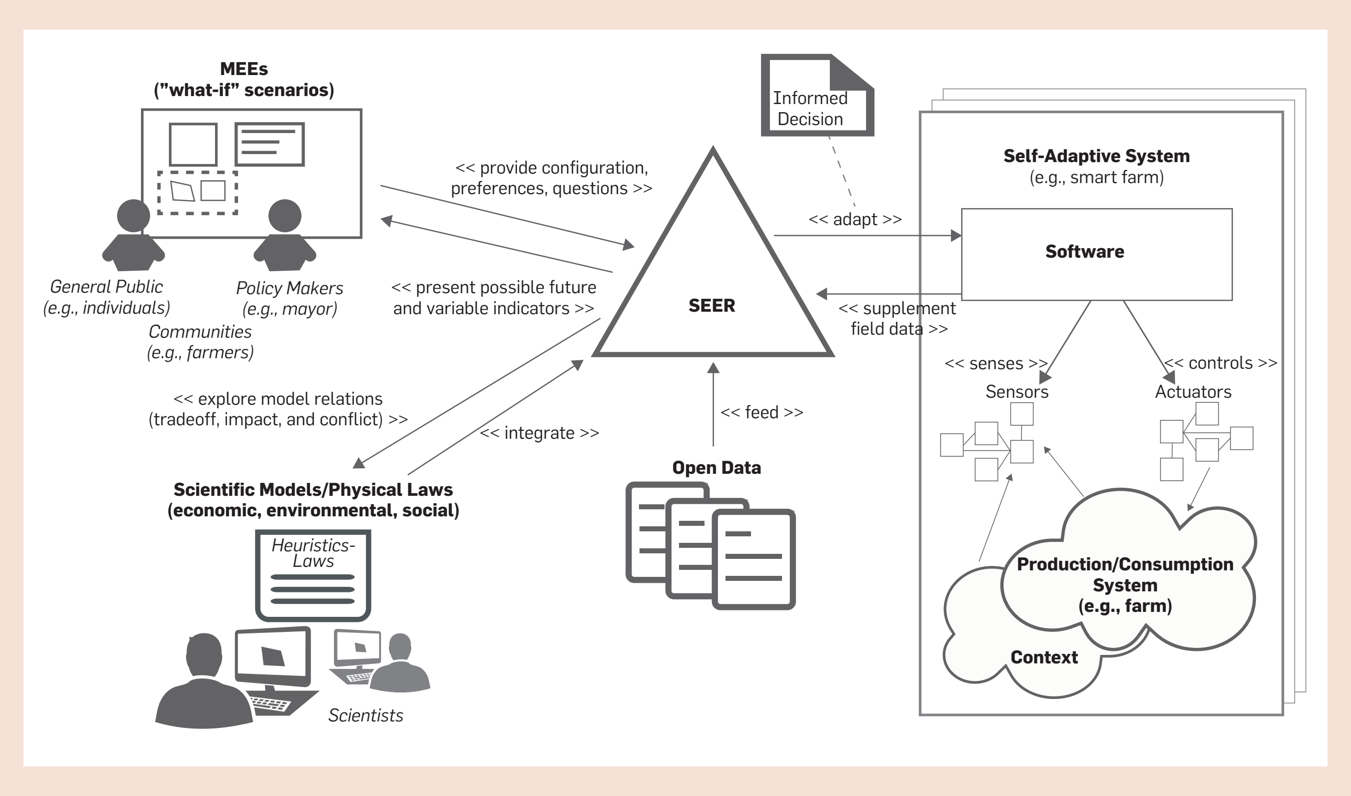
Sustainability Modeling

Modeling has been the essential mechanism to cope with complexity. While in science, models are used to describe existing real-world phenomena, in engineering, models are mostly used to describe a system that is to be developed. Thus, engineering models are typically constructive, while scientific models, for example, mathematical models and stochastic models, are typically used to predict real world aspects.

Modeling underpins many activities related to sustainability. As such, research in MDE can provide a framework for conceptualizing and reasoning about sustainability challenges. One key challenge is how to support decision-making and trade-off analysis to guide behavior of (self-adaptive) systems used for addressing sustainability issues. For this purpose, we present an idealized vision of a conceptual model-based framework, termed *Sustainability Evaluation Experience R* (SEER) as depicted in the accompanying figure. This system enables broader engagement of the community (for example, scientists, policymakers, and the general public), facilitates more informed decision-making through what-if scenarios, and directly uses these decisions to drive the automatic and dynamic adaptation of self-adaptive systems (SAS).[14] We elaborate this vision not as a design for a system to be implemented, but as a framework that enables us to distill the main nine capabilities needed to tackle this multidisciplinary challenge. Since we argue that MDE is one of the main enablers for a system like the SEER, we contemplate the challenges for MDE research that lie ahead.

Vision. This article introduces the SEER, a conceptual entity that brings

Figure 1. The Sustainability Evaluation Experience R.



together sustainability scientists and decision makers, whose output can be used to guide dynamic adaptation of an SAS. As such, the SEER focuses on enabling scientists to integrate and then test their heterogeneous models with an existing knowledge base; enabling individuals and policymakers to explore economic, social, and environmental impact of decisions, investigate trade-offs and alternatives, and express preferences; automating the acquisition of contextual data and enactment of decisions by directly feeding into the knowledge that guides the adaptation of an SAS. The SEER will give the context to introduce the nine capabilities.

Model integration. Scientists must be able to continuously integrate new knowledge into the SEER in the form of models or data. For example, an agronomist can contribute a biomass growth model corresponding to a newly discovered cultivation technique, or a city can decide to openly disclose urban data. Scientists can further connect this contributed material to available and relevant open data. Furthermore, they could investigate the consistency and validity of their models by testing them in combination with other existing domain models. This would help scientists to reach a common view or to highlight important divergences for discussion. To this end, the SEER must provide facilities for flexible data and model integration (C1), model curation (C2), as well as enable trustworthy open-world contributions (C3). The SEER should also support those scientists in investigating the consistency between heterogeneous models, accommodating different and possibly divergent world views (C4).

Model exploration and investigation. On the basis of this knowledge, individuals, communities, and policymakers would explore scenarios, evaluate trade-offs along the five sustainability dimensions[7] (technological, environmental, social, individual, economic) or planetary boundaries,[45] and explore direct, enabling, and structural effects (see the sidebar "Orders of Effects on Sustainability"). Hence, the SEER must enable use by the population at large (C8). For example, a farmer who is considering building a biowaste plant to become energy independent could investigate the consequences of this idea. This analysis must include basic information about the farmer's preferences and the current as-is situation, and to elicit any required information which is not available. To analyze this issue, the SEER needs data and model sources, such as an operational model of the farm or the heating system of the house. The SEER visualizes the analysis results to facilitate exploration. For example, economic analysis might suggest that heating with biowaste is more cost effective than oil. However, the user may doubt this assertion and wish to investigate the result, so the SEER should provide a transparent ratio-

Wicked Problems

The concept of 'wicked problems' was first described in the context of planning by Rittel and Webber.[44] The concept has been used in sustainability-related domains to conceptualize issues including climate change, controlling pandemics, and reducing social injustice. Often misunderstood within computing as simply difficult problems, the concept rather points to the inadequacy of problem solution pairs when used to identify and address complex issues in such situations. The crucial challenge in many situations lies instead in the multiplicity of legitimate and legitimately contradictory perspectives and worldviews about what the issues are. Those views cannot simply be reduced to a correct problem definition using logical operations, but require a discursive process to articulate a definition of the issues to address.[44]

Digital Curation

Data curation is a type of digital curation that involves the active management and preservation of digital resources for future use.[53] Digital resources extend far beyond what is commonly understood as data and include artifacts as diverse as scientific models, engineering models, and electronic records. Curation aims to ensure the quality of resources and provide a record of provenance to make resources discoverable and meaningful and instill trust in their authenticity. The ability to effectively create, share, and manage diverse assets for current and future use is critical for a sustainable society. Supporting trust provides a crucial objective for curation activities, but curation applies not only to resources that are assumed to be trusted a priori. As Rusbridge et al. point out, "long-term stewardship of digital assets is the responsibility of everyone in the digital information value chain."[46] Crucial curation activities may be carried out by actors that are not information professionals, such as citizen scientists annotating and releasing a dataset. This is especially pertinent in our scenario, where many activities of curating datasets and models take place "in the wild"[19] beyond the narrowly controlled confines of a rigorously defined data curation workflow.

Orders of Effects on Sustainability

Any given (software) system exercises three types of effects on sustainability of its situated environment:[7]

***Immediate effects* occur due to the production and immediate use of the system, for example, direct environmental impact of the SEER includes the amount of energy and effort spent on the development, and the reduction of energy consumed by using SEER to find (and set up) a low energy boiler.**

***Enabling effects* come from the ongoing use of a system, for example, as SEER promotes more energy-efficient choices to all its users, the system would enable reduction of energy use for heating, lighting, transportation, and so on, which cuts down the amount of energy resources needed. Yet, it also increases use of broadband for model evaluation, possibly require additional servers to process the vast number of models that the open participation of users from several domains requires, thus increasing energy and broadband consumption.**

***Structural effects* arise due to the long-term reaction of the dynamic socioeconomic system to the presence and use of the system, including lifestyle and economic/structural changes. For instance, if many farmers look for an energy-efficient way of selling their produce, with SEER recommending e-trade, the trade may move from physical markets to e-shops and e-markets, thus changing the selling and shopping behaviors as well as markets in a given community.**

nale and quantification of uncertainty (C6), as well as expose the underlying data. In addition to generating what-if scenarios (C5), the SEER should be capable of generating suggestions (C7) of how to reach user specified goals including quantifiable impacts.

Model automation. Strategic choices typically require a set of well-defined steps to implement them, a process that can benefit significantly from automation. This is especially pertinent when those steps are controlled by an SAS, for example, smart cities or smart buildings. In such cases, decisions are used directly to drive the runtime adaptation of the SAS. For example, when a farmer chooses to grow a specific crop, the SEER could continuously adjust the irrigation system to deliver the appropriate amount of water to the fields. Thus, the SEER must perform sustainability evaluation to determine adaptation needs (C9) to enable broader engagement from the various sustainability stakeholders and would hence serve as an adaptation trigger for an SAS.

Model-driven engineering aims to raise the level of abstraction at which a software system is designed, implemented, and evolved to improve the management of intrinsic complexity.

MDE For SEER

Here, we revisit the capabilities introduced previously and discuss how techniques from the MDE community and other associated communities can support them.

MDE aims to raise the level of abstraction at which a software system is designed, implemented, and evolved to improve the management of intrinsic complexity.[25] In MDE, a model describes an aspect of a system and is typically created for specific development purposes. Separation of concerns is supported through the use of different modeling languages, each providing constructs based on abstractions that are specific to an aspect of a system. But systems like the SEER also require, as a central function, a set of abilities to curate diverse collections of data and manage them throughout a long-lasting lifecycle to address concerns such as authenticity, archiving, transformation, reuse, appraisal, and preservation. In this context, data monitoring involves the continuous, automated acquisition of new datasets.

Accommodate flexible data and model integration (C1). The MDE community has been investigating how to integrate engineering models for various purposes (for example, analyses, code generation, simulation, execution). In addition to comparison operators such as those that can be defined in the Epsilon Comparison Language,[c] the community has developed various composition operators for model refinement/decomposition,[42] model consistency or impact analyses,[26] and model merging and weaving.[9] While these composition operators have been extensively studied for homogeneous and structural models,[15] recent efforts are also considering behavioral and heterogeneous models.[33]

In the software and systems modeling community, research on domain-specific modeling languages (DSMLs) is investigating technologies for developing languages and tools that enable domain experts to develop solutions efficiently. Unfortunately, the current lack of support for explicitly relating concepts expressed in different DSMLs makes it difficult for domain experts to reason about information distributed across models describing different system views. Supporting coordinated use of DSMLs led to the grand challenge of the *globalization of modeling languages*[16] and the GEMOC initiative. Beyond the current investigations that focus on relating languages of similar foundations, sustainability issues will impose additional research challenges relating to multiscale, uncertainty, and approximation or discontinuity.

An alternative to integrating DSMLs is to integrate models by co-simulation or model translation. For example, the functional mock-up interface (FMI) is a tool independent standard to support both model exchange and co-simulation of dynamic models using a combination of xml files and compiled C-code.[d] FMI is currently supported by over 130 modeling and simulation tools. Model translation approaches construct model transformation algorithms that integrate models by mapping them into a common modeling formalism. For example, the work described in Castro[12] transforms system dynamics models into discrete event system specification (DEVS) models,

[c] https://www.eclipse.org/emf/compare/

[d] The Functional Mockup Interface Standard, https://fmi-standard.org

which can then be integrated further with other discrete modeling formalisms, for example, state automata.

Unlike software-intensive systems, the SEER requires integration of numerous scientific models, regulations, preferences, and so on, when making predictions and in order to consider the many trade-offs when looking for potential solutions. The challenges for integrating models within a SEER are due to the following factors:

Different foundations. In traditional MDE, foundational notions, for example, hierarchy/containment or references, are used in constructing models; different notions are used in other modeling spaces (for example, derived attributes in MetaDepth[22]). The integration process must acknowledge and align these different notions.

Different technological spaces. Models may be constructed using mechanisms from different technological spaces (for example, databases, formulae such as ODEs), with varying assumptions about the basic building blocks of modeling; how those building blocks can be composed; how the well-formedness of models can be established; and how well-formed models can be manipulated.

Different levels and degrees of abstraction. Integrating models involves more than just establishing a consistent vocabulary: disparate models will use different abstractions (for example, patterns specific to the type of model), different layers and layering structures (for example, networking layers versus atmospheric chemistry model layers), and different forms of granularity (a grid of contemporaneous rainfall observations over a large area versus a time series of measurements of cumulative water flow at one location in a river).

Different scales. To integrate models at different scales, a model integration approach would have to clearly distinguish: Which models belong to which layers of abstraction (for example, given a predictive model of evapotranspiration that can be constructed for Earth as a whole, for a continent or a watershed, which one is relevant when integrating this with a model of crop production at a country level?); Which specific model out of a set of alternatives to use when there is no evidence demonstrating superiority of one model over another (for example, with insufficient ground truth to distinguish between two multispectral classifications, what characteristics of the classifiers would help the system to choose an option?); and How conflicts or inconsistencies between models and/or data are resolved (for example, given a set of decision trees that risk being overfitted to their training data, is it necessary to employ an ensemble method such as random forest?).

Different domains. In order to meaningfully integrate data from a variety of domains, it needs to be carefully described with metadata. This should include descriptions of units, phenomena measured, and other conceptual aspects, which are vital for communication when data is released "into the wild."

Composability. A crucial capability for the SEER is to automatically identify which data can and cannot logically be combined. For example, a user might be interested in assessing the economic value of a national park by overlaying its bounds on maps of ecosystem services. Such maps might be calculated in different ways, leading to conflicting results. For example, carbon capture per hectare may be computed for specific land covers by methods which rely on different assumptions about underlying physical processes. Should results derived from such datasets be averaged, or be shown as alternatives? A robust approach to this automated matching requires semantics to describe the underlying worldview implicit in each estimate.

Curate and evolve models (C2). The SEER must facilitate continuous management of models to ensure the generation of valid what-if and how-to scenarios. Model management involves supporting updates to models and to model integration. Key activities include model import and creation (for example, scientific model creation out of datasets), enhancing model quality, and representation of different views.

There are two approaches to scientific model creation: either start with a skeletal model with a few initial data points and incrementally collect relevant data while refining the model relationships; or build a model based on analysis of all accessible data.

From the perspective of the robust management of MDE products over time, version control is essential to reflect the state of the model at the time when a dataset was imported. When this initial dataset does not conform to later, updated versions of the model, maintenance challenges arise for the datasets. Conceptual approaches for version control in MDE have been developed, based on techniques for comparing and differencing models[23] as well as merging models. More recently, tools such as EMF Store[e] and CDO[f] have been developed, which are closely aligned with version control systems such as git. Conflicts are common with such approaches, and hence support for their detection and resolution are critical. Such tools typically are combined with those for comparison and differencing (for detection), and merging (resolution).

From the perspective of digital curation, larger concerns around provenance, authenticity, and stewardship become paramount. The provenance of data has been a central concern in fields such as databases and e-science.[11,41,47] Provenance modeling initiatives have focused on conceptual frameworks for representing generally applicable elements that capture provenance information in standardized ways.[g] Concepts such as research objects capture more than the dataset to support the flexible reuse of various products in research workflows and in particular, model-based scientific workflow software such as Kepler and Taverna.[6] Again, data provenance is a central concern and raises new challenges, as we will discuss.

Enable trustworthy open-world contribution (C3). To enable trustworthy open-world contributions, everyone should be allowed to contribute to the SEER, regardless of their social background, domains of expertise, or technical qualifications. A simple example of the utility of such contributions are the citizen-science projects. For instance, the U.K's Spring Watch program enlists radio listeners to report, via text and/or photographs, the observations of native wild life species, which can be a cost-free tool for observing,

[e] https://www.eclipse.org/emfstore/

[f] http://www.eclipse.org/cdo/

[g] https://www.w3.org/TR/prov-overview/

recording, and where necessary taking actions for preserving biodiversity. Contributions to the SEER would consist not only of data or models, but also of new mappings or relationships for integrating data and models.

To foster trust toward and use of the contributions, their provenance must be publicly availed. This is essential[47] in order to assure potential data users of the quality of the given data (providing answers to such questions as: what is the data source, were the derivation methods of the current data sound?); support the owners and users with the audit trial (Who is using the data? Are there any errors in the data generation?); provide recopies for replicating data derivation in order to maintain currency of the data, as well as to maintain clear derivation recipes; support attribution of data for both copyright and liability assignment purposes; and provide information about the data context, and for data discovery.

Currently data curation is being tackled by open-world contributions that have little provenance, so the quality of that data and the collection processes are questionable. For example, in the CARMEN bioinformatics project,[h] researchers can submit data and the metadata that describes it. However, provenance information is limited to the identity of the source. Yet, it is widely acknowledged that, in order to provide credible provenance for scientific workflow, one needs to report provenance not only of the provided data (for example, its sources and their views, including interests, purpose, concepts, principles, knowledge[29]) but also the process through which the data is derived (for example, used methodology, and technologies for data collection).[29,47]

In MDE's few open repositories for models, for example, ReMoDD[i] or the ATL Metamodel Zoo,[j] the situation is even worse, as little information is kept on the provenance or quality of the models, despite the long-established specification of provenance requirements for e-science systems.[40]

The challenges for trustworthy open-world contributions pertain to the following:

Subject of provenance,[47] or the provenance of data and its workflow: It is not clear at what level of detail provenance information needs to be gathered (for example, what granularity should the data be collected, rainfall per cm^2 or km^2?). Which sources are acceptable, for what purposes?[29] When pulling together several datasets, or starting analysis for a given purpose, are the used data collection methods and technologies compatible/appropriate for the said purpose? Who must take responsibility for errors in data collection or derivation? Eventually, how do the sources, their properties and the workflow affect the data quality, and how can the quality be separated from the notion of provenance itself?

Provenance representation.[47] Should data be annotated directly with the provenance details (for example, many scientific workflow tools, such as Taverna, record the provenance data implicitly in event logs[21]), or should provenance be derived at each workflow stage from the previous one? What syntax and semantics should be used to represent it? Can these be applicable across all kinds of domains, as the SEER has to integrate environmental, economic, technical, societal, individual, policy, and cultural aspects of life?

Storing provenance.[47] What are the costs of collecting and storing the provenance data at various granularity? Clearly, the richer the provenance data, the more it will affect the scalability of data collection and storage.

Integration. If the system accommodates import of new concepts of all kinds, we face integration challenges, for example, to find the best, that is, most reasonable, or most flexible open interfaces and common description language. Furthermore, the research community must let the ontology evolve iteratively, by adding new parts.

Trust. How do you foster trust, or calculate trust into the given model's output? How do we build trust models? How can we apply theoretical research models in the real world while large scale empirical evidence is still missing?

Relationship between risk and trust. How to deal with the inherent relationship between risk and trust? What are the risks involved in trusting a given model/data/process, and how to quantify these? Contrary to public perception, high trust does not mean low risk.

Currently research is ongoing on ways for handling many of the challenges mentioned earlier for controlled environments, such as for scientific work flows within tools like Taverna[k] and Kepler[l] (here, datasets and workflows are provided only by scientists or models by research groups who stake their professional reputation against the quality of their contributions). When the controls for contributions are removed, however, these challenges redouble and multiply.

Accommodate different world views (C4). The breadth of the impact of sustainability across five dimensions and multiple time scales, from human to global, inevitably brings with it differing and irreconcilable worldviews, and separates stakeholders socially and temporally.

To avoid bias, the SEER should provide all possible futures accommodating multiple and potentially divergent worldviews to the user given the available data and models. Therefore, the SEER must acknowledge that a model is constructed with its own (often implicit) worldview.[37] Model integration requires combination of the views, which can be challenging or even impossible if they contradict.

The modeling community deals with situations where worldviews are assumed to be consistent across stakeholders if they share the same modeling background.[36] In most engineering environments this is acceptable, since even large-scale systems have an ultimately "bounded" set of stakeholders. In these scenarios, any necessary negotiation of conflicting worldviews is a question of social organization and not addressed in modeling.

Traditional MDE normally resolves contradictions under model integration using constraints and transformations. This is feasible because even when the worldview is not fully shared, there should be overlap arising from agreement on a metamodeling stack (for example, three-tiered) and technology (the Eclipse Modeling

[h] http://www.carmen.org.uk/

[i] https://www.cs.colostate.edu/remodd/

[j] https://web.imt-atlantique.fr/x-info/atlanmod/index.php?title=Zoos

[k] https://taverna.incubator.apache.org/

[l] https://kepler-project.org/

Framework, EMF). This cannot be assumed in modeling for sustainability, where the social structure is so disconnected that the common assumption of consistent worldviews in MDE cannot hold. Different modeling schools must be integrated and multiple contradictory worldviews need to be made explicit and embraced.

The worldview has to become an explicit part of the modeling infrastructure, and several possible scenarios arise as noted in the following:

Matching worldviews. In some cases, worldviews can be reconciled. However, there may be no "actual" user/modeler who possesses this integrated view. How can this integrated view be derived/validated?

Incommensurable worldviews and models. Considering the fundamentally distinct nature of the types of concerns of interest for the stakeholders in sustainability, perspectives on what seems to be a common concern will not only disagree on the weighting of importance of particular aspects, such as "individual agency," but also on what this concern means, and how to evaluate it.

Contradictory worldviews. It should not be assumed that reconciliation of contradicting worldviews is always desirable and appropriate. Sometimes it may be desirable and useful to keep track of contradictions between models. To discuss this, we provide here a few examples for worldviews that disagree at least partially:

Incommensurable. In California, environmental sustainability can be regarded as fundamentally different in the problem context of preserving existing wetlands versus restoring an urban landscaping back toward the natural desert environment it was taken from.

Contradictory. In many developing cultures, big families still form the heart of the community. In many developed cultures, family structures have been overshadowed by career paths requiring mobility. One consequence is that two-income families struggle with local support systems for their kids while grandparents live far away and struggle with lonely old age. Neither worldview is wrong, but they cannot be consolidated completely.

The research challenge arising from this is not an unrealistic attempt at consolidating all existing worldviews. Instead, what we need are modeling concepts and mechanisms that allow us to contrast different worldviews to illustrate and explore conflicts between the assumptions and implications of two or more worldviews.[37] One option would be to use system dynamics to reach a group consensus and enhance systems thinking.[50]

To avoid bias, the SEER should provide all possible futures accommodating multiple and potentially divergent worldviews to the user given the available data and models.

However, system dynamics on its own is arguably incapable of securing consensus.[30] Because it lacks the awareness of social theory required to distinguish consensus from coercion, it must be positioned within a critically aware systems thinking framework that reflects upon its own selectivity, aims to emancipate marginalized perspectives and worldviews, and allows for pluralism in methods and theories.[39]

A useful starting direction in tackling these issues could be provided by model-documenting guidelines (for example, the ODD protocol[28]) that help to systematize and disambiguate categorizations of heterogeneous models, though full resolution of integration of such models is an open challenge.

Generate what-if scenarios (C5). The system should support the generation of what-if scenarios based on multiple types of models to project the scenarios' effects with regard to the five sustainability dimensions. Interactive exploration of the scenario as well as the involved data and models should be possible. Here, it is important the user of the SEER gets a feeling about how a possible future scenario may look and what effects the anticipated changes will have on the different sustainability dimensions. For example, what would a world look like that no longer used fossil fuel? To help SEER users understand the what-if scenarios and make the experience even more tangible, visualization techniques going beyond the presentation of numbers are needed.

What-if scenarios require query formulation, which is supported through query languages. These languages have been investigated by the MDE community with an intensive focus on automatic model management (for example, constraints, views, transformation). MDE provides languages for expressing structural queries based on first order logic (OCL[m]), use of optimization and search techniques combined with models,[24] as well as for behavioral queries based on temporal logic.[38] These languages rely on the modeling language specification for expressing queries related to the corresponding concepts or their associated behavioral semantics. The concept of model experiencing environments (MEEs)[43] has been introduced as an approach to support complex model and data integration, while offering customizable interfaces to access model analysis results and their visualizations.

The need for broad engagement with diverse communities and decision makers requires an ability to process questions articulated within the mental models and terminologies used by communities, and support cross-domain compatibility and mapping across various domains. Different impacts must be presented back to the user (using different kinds of visualizations), in such a form that the indicators and their underlying assumptions can be deeply and interactively analyzed for a better understanding. Current practices must be adapted to support the what-if scenario capability. This requires a bridging of the gap between the indicators and the modeling concepts manipulated by the SEER. The user must be able to express the indicators of interest, and the specific views to be used for representing them.

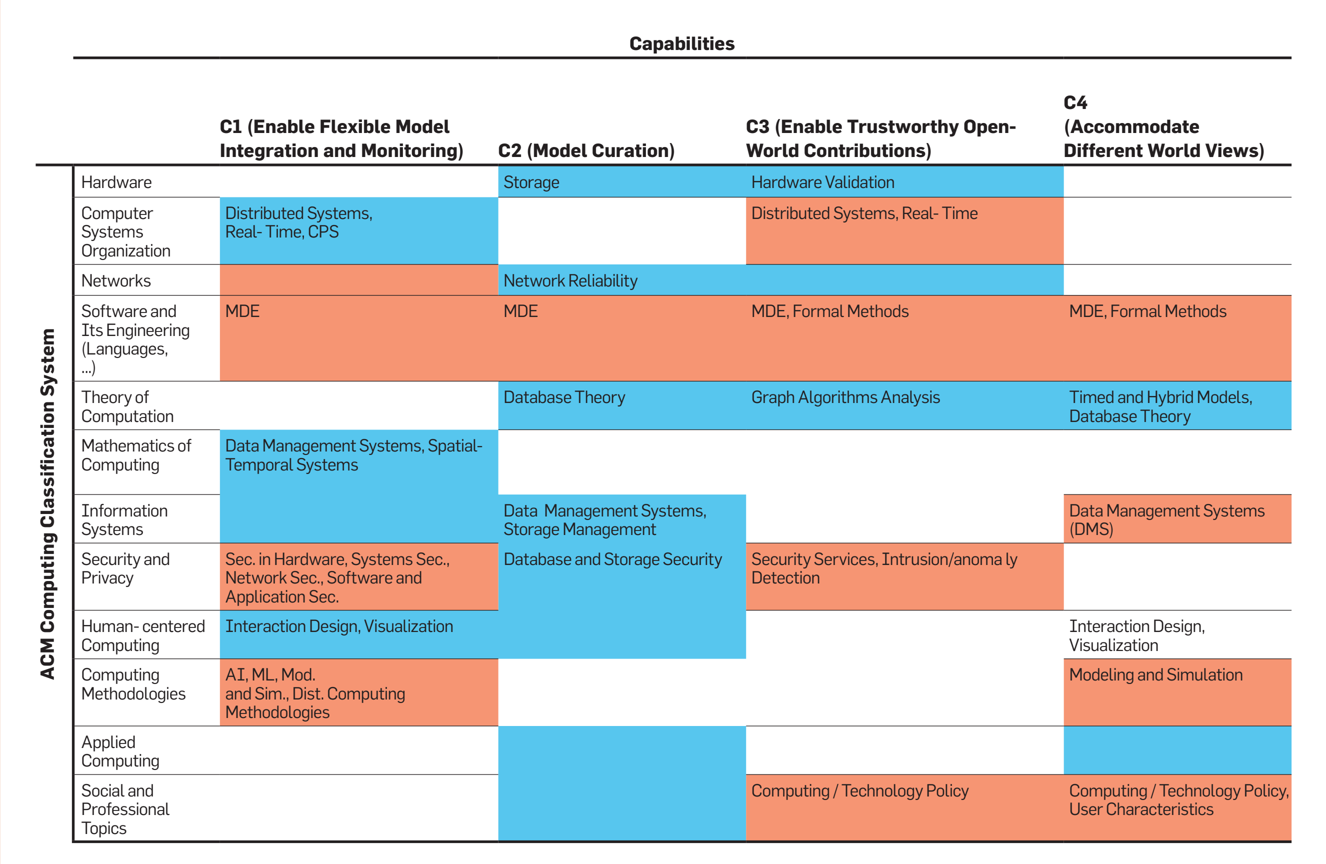
Provide transparent reasoning and quantification of uncertainty (C6). If users do not feel they understand what is happening in a system and why, they are less likely to trust it. Therefore, trustworthiness can only be established if the reasoning provided by the SEER is transparent, meaning users can understand where data comes from, to what degree it is reliable, and how it is combined in order to generate predictions.

Intra-model relationships have been a general focus of interest in the MDE community. User-defined mappings between MDE models are supported via model management tools such as the Atlas Model Weaver, EMF Compare, or the Epsilon Comparison Language (ECL). These approaches enable users to describe mappings between models and model elements and attach semantics to the relationships that are produced. Such models are usually within a single technological space (for example, EMF). There are also software component interface definitions, such as OpenMI and Taverna, which provide APIs that allow models to be configured to exchange data at run-time within workflows. While such technology is meant to be model agnostic, it supports connection of models from within a technological space. Additionally, such frameworks effectively focus on mappings between data, where the models are used to enable the construction of such mappings.

There has been limited research in the MDE community on dynamic model selection from a large set of models or on run-time conflict reso-

[m] https://www.omg.org/spec/OCL/

Figure 2. CS disciplines contributing toward realizing the SEER capabilities.



	Capabilities			
ACM Computing Classification System	C1 (Enable Flexible Model Integration and Monitoring)	C2 (Model Curation)	C3 (Enable Trustworthy Open-World Contributions)	C4 (Accommodate Different World Views)
Hardware		Storage	Hardware Validation	
Computer Systems Organization	Distributed Systems, Real- Time, CPS		Distributed Systems, Real- Time	
Networks		Network Reliability		
Software and Its Engineering (Languages, ...)	MDE	MDE	MDE, Formal Methods	MDE, Formal Methods
Theory of Computation		Database Theory	Graph Algorithms Analysis	Timed and Hybrid Models, Database Theory
Mathematics of Computing	Data Management Systems, Spatial-Temporal Systems			
Information Systems		Data Management Systems, Storage Management		Data Management Systems (DMS)
Security and Privacy	Sec. in Hardware, Systems Sec., Network Sec., Software and Application Sec.	Database and Storage Security	Security Services, Intrusion/anoma ly Detection	
Human- centered Computing	Interaction Design, Visualization			Interaction Design, Visualization
Computing Methodologies	AI, ML, Mod. and Sim., Dist. Computing Methodologies			Modeling and Simulation
Applied Computing				
Social and Professional Topics			Computing / Technology Policy	Computing / Technology Policy, User Characteristics

lution between models from disconnected domains and disciplines (most conflict resolution has focused on resolution between models from single or related domains). Current work on justifying model integration reasoning is centered around such topics as edit-aware modeling tools that keep track of the steps that the modelers take in modifying the model (for example, Altmannager et al.[2]) and tool support that allows one to keep track of all the versions of a model (Sparx Time Aware Modeling, Magic Draw Comparer, EMFCompare).

In goal modeling, the impact of alternative solutions on stakeholders' objectives is modeled to allow reasoning about trade-offs. Based on such models, explanations may be given of what influences what. It is still challenging to generate clear explanations of scenarios built on top of widely different types of models, each requiring different argumentation and concepts. For example, when analyzing a chart with a Pareto front to make an allocation decision, the farmer might see a cut off on one dimension. She might ask "but couldn't I do this?" for example, increase output beyond *x*? The SEER would need to be able to explain the Pareto front does not only take into consideration physical possibilities, but also considers legal constraints.

Within the domain of environmental modeling, there has been some consideration of integration challenges,[52] particularly in relation to the propagation of uncertainty through a series of chained models and its communication in a usable form at the end of the analysis.[3] 'Models' in that context, however complex, are concrete mathematical transformations that represent physical processes such as soil erosion, or non-physical processes such as market fluctuations. As such they are materializations of the more abstract class of models with which the SEER must work, and form just part of the set of components of which it must be composed.

However, many of the insights from this research also apply to an integrated system such as SEER: for example, the importance of semantics and controlled vocabularies in describing requirements, constraints or phenomena, and the fact that physical models may also be matched and merged as appropriate.

The uncertainty of available data and information hinders the precise specification of certain models and their parameters. Uncertainty may be, for example, epistemic, linguistic, or randomized[27] and can derive from many sources including measurement, data transformation, inaccurate definition of the phenomenon of interest, or generalizations made to ensure tractable computation. As such, uncertainty analysis (UA) and sensitivity analysis (SA) are prerequisites for model building.[18] While UA aims to quantify the overall uncertainty associated with the model response as a result of uncertainties in the model input, SA can be used to quantify the impact of parameter uncertainty on the overall simulation/prediction uncertainty. This makes it possible to distinguish between high

		Capabilities				
		C5 (Generation of what-if scenarios)	C6 (Transparent Reasoning/Uncertainty)	C7 (Generate suggestions)	C8 (Accessible to the population at large)	C9 (Sustainability Evaluation-based Adaptation)
ACM Computing Classification System	Hardware		🟦 Robustness		🟧	
	Computer Systems Organization		🟧 Real-Time, Dependable Systems			🟦 Distributed Systems, Real- Time, CPS
	Networks		🟦 Network Reliability			🟧
	Software and Its Engineering (Languages, ...)	🟧 MDE	🟧 MDE	🟧 MDE	🟧 MDE, Context- Specific Languages	🟧 MDE
	Theory of Computation	🟧	🟦 Approximation Algorithms Analysis, Separation Logic	🟧		🟧
	Mathematics of Computing	🟧	🟧	🟧		
	Information Systems	🟧 DMS, Data Mining, Decision Support Systems	🟧 Data Management Systems	🟦 Data Management Systems	🟦 Users and Interactive Retrieval	
	Security and Privacy		🟦 Security Services, Intrusion/ anoma ly Detection		🟦 Software and Application Sec., Human and Societal Aspects of Sec. and Privacy	🟧 Security in Hardware
	Human- centered Computing		🟦 Interaction Design, Visualization		🟧	
	Computing Methodologies	🟧 Modeling and Simulation	🟦 Modeling and Simulation	🟧 Modeling and Simulation	🟦 Computer Graphics, Modeling and Simulation	🟦 Distributed Computing Methodologies
	Applied Computing	🟦				🟦
	Social and Professional Topics	🟦 Computing and Business		🟦 Computing and Business	🟧 Comp. Educ., Comp. /Techn. Policy, User Characteristics	

leverage variables, whose values have a significant impact on the system behavior, and low-leverage variables, whose values have minimal impact on the system.[31,54] Such approaches can be used for various purposes, including model validation, evaluating model behavior, estimating model uncertainties, decision-making using uncertain models, and determining potential areas of research[34] and a variety of SA techniques have been developed to achieve such purposes.[35] However, federating several models is likely to result in the potential problem of enlarging the parameter space, which will require the automated detection of hotspots in the parameter space using approaches such as the ones proposed by Danos et al.[20]

Nevertheless, not all sources of uncertainty are known, and many are difficult to quantify. Uncertainty which sources can be assessed statistically may be communicated, for example, using probabilities, which are easily combined across a wide variety of well-supported frameworks and languages, for example, UncertML.[51] Fuzzy sets are more complex to combine across domains but can still be represented in mathematical form. However, on many occasions a quality assessment is not easily mapped to a value scale, or a problem does not become apparent until a dataset or model is used or compared to better alternatives that were not originally available. This is a clearly recognized challenge in citizen science, where a number of initiatives aim to harmonize metadata standards,[4] to adapt existing data formats to the citizen science context,[48] to develop robust ontologies to capture heterogeneous data collection protocols and to allow flexible annotation by contributors and expert evaluators alike.[n,5] Only through such concerted efforts can a potential user assess whether the reliability of a contributed resource matches their criteria, making it fit-for-purpose.

Generate suggestions (C7). The system should be capable of generating suggestions of how to achieve the user's specified goals. This generation of suggestions is based on the capability to create what-if scenarios (C3), as those are needed to build a knowledge base for a recommender system. Based on such a what-if scenario knowledge base, a recommender system can generate how-to scenarios by using model inference. Inferred models can be compared to current ones and criteria applied to select the most appropriate candidate solutions, for example, the closest to the current situation. Therefore, the SEER must calculate different alternatives to minimize negative impact on the different sustainability dimensions. To do so, the system must be informed what a user may and may not change, for example, they cannot change the weather. Furthermore, the SEER needs to know user preferences in order to make adequate individual suggestions. Such user preferences include the modeling view of the system under consideration, the agency over individual elements, and the scale at which they can be changed. The preferences could even be changed at run-time and the model recalculated based on the updated constraints.[49]

The uncertainty of available data and information hinders the precise specification of certain models and their parameters.

Enable use by the population at large (C8). Since the SEER is to be used by the population at large, careful consideration must be given to human factors and ergonomics in system design. Some example issues to be addressed here include simple ways to establish and update preferences and goals (for example, via graphical or voice-based interfaces); results interpretation (via visualization or voice feedback explaining the results' implications); and customization of interactions for different user groups (domain-specific model customization support for specialist users). The quality of the users' experience[10] should also be considered, accounting for the users' emotional and physiological states, the situational characteristics of the experience, and the experience of model use itself.[1]

Evaluating adaptation for sustainability (C9). Based on the sustainability evaluation performed by the SEER, adaptation triggers may be generated to guide the self-adaptation of an SAS. In the original framework proposed by Kephart and Chess,[32] an SAS has four key stages (MAPE-K loop): Monitoring environment and system conditions, Analysis to determine whether

[n] https://www.w3.org/TR/vocab-dqv/

the system needs to self-reconfigure, planning for how to adapt the system safely to satisfy new requirements/needs, and Execution of the adaptation plan. All four stages make use of a Knowledge resource. While the original intent for Knowledge was for static information (for example, sensor properties, policies, and constraints), for our purposes, we realize the Knowledge resource with the SEER. As such, the SEER becomes a dynamic source of sustainability-evaluation knowledge that incorporates input from the stakeholders, scientific models and their integration, open data, results of what-if scenario exploration, or user needs to guide the self-adaptation of an SAS. The entire MAPE-K loop is hence open for human assessment and feedback to derive a recommendation that can either be realized by an automated adaptation or realized by human intervention. For example, Bruel et al.[8] present a smart farming system including an irrigation system that determines and delivers the right amount of water every day in order to maximize produced biomass, based on current water stress, the climate series, biomass models, and the farmer's input.

Conclusion

In this article, we detailed each capability needed by the SEER and reported on how MDE has already contributed toward that capability. However, most of the disciplines in computer science (CS) must come together to realize the SEER vision outlined here. Therefore, we used the ACM Computing Classification System[o] to assess the CS disciplines and create a simplified heat map (see Figure 2) where we indicate for each top-level category whether or not we, that is, the 16 authors, believe it is not relevant (white), relevant (blue), or highly relevant (red) to realize the SEER. Whenever we feel that some subcategories are notably more important than others, they are mentioned explicitly in the appropriate cells of the heat map. The heat map represents the biased view of the authors, and as a result, the importance of some categories might have been misjudged. In general, it can be supposed that expertise in CS is needed across all capabilities, that each of the CS categories is highly relevant for at least one of the capabilities, and finally that MDE is highly relevant across all capabilities.

[o] http://dl.acm.org/ccs/ccs.cfm

References

1. Abrahao, S. et al. User experience for model-driven engineering: Challenges and future directions. *Model Driven Engineering Languages and Systems*, 2017, 229–236.
2. Altmanninger, K. et al. Why model versioning research is needed! An experience report. *MoDSE-MCCM Workshop at MoDELS*, 2009, 1–12.
3. Bastin, L. et al. Managing uncertainty in integrated environmental modelling: The uncertweb framework. *Environmental Modelling and Software 39*, 2013. Elsevier, 116–134.
4. Bastin, L. et al. *Good Practices for Data Management.* Chapt. 11, 2017.
5. Bastin, L. et al. *Volunteered Metadata, and Metadata on VGI: Challenges and Current Practices.* Springer, 2017.
6. Bechhofer, S. et al. Why linked data is not enough for scientists. In *Proceedings of 2010 IEEE 6th Intern. Conf. e-Science*, Dec. 2010, 300–307.
7. Becker, C. et al. Requirements: The key to sustainability. *IEEE Software 33*, 1 (Jan. 2016), 56–65.
8. Bruel, J.M. et al. MDE in practice for computational science. In *Proc. of Intern. Conf. on Computational Science*, June 2015.
9. Brunet, G. et al. A manifesto for model merging. In *Proc. of 2006 Intern. Workshop on Global Integrated Model Management*, 2006, 5–12.
10. Bui, M. and Kemp, E. E–tail emotion regulation: Examining online hedonic product purchases. *Int. J. Retail and Distribution Management 41*, 2013, 155–170.
11. Buneman, P., Khanna, S., and Wang-Chiew, T. Why and where: A characterization of data provenance. *Database Theory ICDT 2001, LNCS.* Springer, 2001, 316–330.
12. Castro, R. Open research problems: Systems dynamics, complex systems. *Theory of Modeling and Simulation (3rd Edition)*, chapt. 24. Academic Press, 2019.
13. Castro, R. and Jacovkis, P. Computer-based global models: From early experiences to complex systems. *J. Artificial Societies and Social Simulation 18*, 1 (2015), 1–13.
14. Cheng, B.H.C. et al. Software engineering for self-adaptive systems: A research roadmap. *Software Engineering for SAS*, 2009, 1–26.
15. Clavreul, M. et al. Integrating legacy systems with mde. In *Proc. of Intern. Conf. Software Engineering*, 2010, 69–78.
16. Combemale, B. et al. Globalizing modeling languages. *Computer*, (June 2014), 68–71.
17. Combemale, B. et al. Modeling for sustainability. *Modeling in Software Engineering*, 2016.
18. Crosetto, M., Tarantola, S., and Saltelli, A. Sensitivity and uncertainty analysis in spatial modelling based on GIS. *Agriculture, Ecosystems & Environment 81*, 1 (2000), 71–79.
19. Dallas, C. Digital curation beyond the wild frontier: A pragmatic approach. *Archival Science 16*, 4 (2016), 421–457.
20. Danos, A., Braun, W., Fritzson, P., Pop, A., Scolnik, H., and Castro, R. Towards an open Modelica-based sensitivity analysis platform including optimization-driven strategies. In *Proc. of EOOLT '17*, 2017. ACM, 87–93.
21. Davidson, S.B. and Freire, J. Provenance and scientific workflows: Challenges and opportunities. In *Proc. of Intern. Conf. Management of Data*, 2008. ACM, 1345–1350.
22. de Lara, J. and Guerra, E. Deep meta-modelling with metadepth. In *Proc. Of the 48th Intern. Conf. Objects, Models, Components, Patterns*, 2010, 1–20.
23. Dimitrios, S. et al. Different models for model matching: An analysis of approaches to support model differencing. In *Proc. of Workshop on Comparison and Versioning of Software Models*, 2009.
24. Faunes, M. et al. Automatically searching for metamodel well-formedness rules in examples and counter-examples. *Model Driven Engineering Languages and Systems, LNCS*, 2013, 187–202.
25. France, R.B. and Rumpe, B. Model-driven development of complex software: A research roadmap. In *Proc. of Workshop on the Future of Software Engineering*, 2007, 37–54.
26. Galvao, I. and Goknil, A. Survey of traceability approaches in model-driven engineering. In *Proc. of EDOC 2007*, Oct. 2007, 313–313.
27. Giese, H. et al. *Living with Uncertainty in the Age of Runtime Models*, 2014, 47–100.
28. Grimm, V., Polhill, G., and Touza, J. Documenting social simulation models: The ODD protocol as a standard. *Simulating Social Complexity*, Springer, 2017, 349–365.
29. Huang, J. From big data to knowledge: Issues of provenance, trust, and scientific computing integrity. *Big Data 2018*, 2197–2205.
30. Jackson, M.C. *Systems Thinking: Creative Holism for Managers.* Wiley Chichester, 2003.
31. Jorgensen, S.E. and Fath, B.D. 2—Concepts of modelling. *Fundamentals of Ecological Modelling* vol. 23, *Developments in Environmental Modelling*. Elsevier, 2011, 19–93.
32. Kephart, J.O. and Chess, D.M. The vision of autonomic computing. *Computer 36* (Jan 2003), 41–50.
33. Larsen, V. et al. A behavioral coordination operator language (BCOoL). *MODELS 2015*, Aug. 2015.
34. Lehr, W., Calhoun, D., Jones, R., Lewandowski, A., and Overstreet, R. Model sensitivity analysis in environmental emergency management: A case study in oil spill modeling. In *Proc. of Winter Simulation Conf.* Dec. 1994, 1198–1205.
35. Hamby, D.M. A review of techniques for parameter sensitivity analysis of environmental models. *Environmental Monitoring and Assessment*, 32:135–154, 09 1994.
36. Meadows, D., Richardson, J., and Bruckmann, G. *Groping in the Dark: The First Decade of Global Modelling.* John Wiley & Sons, 1982.
37. Meadows, D. H. and Robinson, J.M. The electronic oracle: computer models and social decisions. *System Dynamics Review 18*, 2 (2002), 271–308.
38. Meyers, B. et al. Promobox: A framework for generating domain-specific property languages. *Software Language Engineering*, 2014, 1–20.
39. Midgley, G. What is this thing called CST? *Critical Systems Thinking.* Springer, Boston, MA, 1996, 11–24.
40. Miles, S., Groth, P., Branco, M., and Moreau, L. The requirements of using provenance in e-science experiments. *J. Grid Comp. 5*, 1 (2007), 1–25.
41. Moreau, L. et al. The provenance of electronic data. *Commun. ACM 51*, 4 (Apr. 2008), 52–58.
42. Mussbacher, G. et al. Assessing composition in modeling approaches. In *Proc. of CMA Workshop*, 2012, 1:1–1:26.
43. Mussbacher, G. et al. The relevance of model-driven engineering 30 years from now. *MODELS 2014, LNCS 8767*, 183–200.
44. Rittel, H.W. and Webber, M.M. Dilemmas in a general theory of planning. *Policy Sciences 4*, 2 (1973), 155–169.
45. Rockstrom, J. et al. A safe operating space for humanity. *Nature 461* (Sept. 2009), 472–475.
46. Rusbridge, C. et al. The digital curation centre: A vision for digital curation. In *Proc. of Intern. Symp. Mass Storage Systems and Technology*, 2005, 31–41.
47. Simmhan, Y.L., Plale, B., and Gannon, D. A survey of data provenance in e-science. *SIGMOD Rec. 34*, 3 (Sept. 2005), 31–36.
48. Simonis, I. et al. Sensor Web Enablement (SWE) for citizen science. In *Proc. of the IEEE Int. Geoscience and Remote Sensing Symposium*, 2016.
49. Tikhonova, U. et al. Constraint-based run-time state migration for live modeling. *Software Language Engineering*, 2018.
50. Vennix, J. A. M. Building consensus in strategic decision making: System dynamics as a group support system. *Group Decision and Negotiation 4*, 4 (July 1995), 335–355.
51. Williams, M. et al. Uncertml: An XML schema for exchanging uncertainty. In *Proc. of the 16th Conf. GISRUK 2008*, 275–279.
52. Wirtz, D. and Nowak, W. The rocky road to extended simulation frameworks covering uncertainty, inversion, optimization and control. *Environmental Modelling and Software 93* (2017), 180–192.
53. Yakel, E. Digital curation. *OCLC Systems & Services: Intern. Digital Library Perspectives 23*, 4 (2007), 335–340; https://doi.org/10.1108/10650750710831466
54. Zheng, Y., Han, F., Tian, Y., Wu, B., and Lin, Z. Chapter 5: Addressing the uncertainty in modeling watershed nonpoint source pollution. *Developments in Environmental Modelling, Ecological Modelling and Engineering of Lakes and Wetlands.* Elsevier, 2014, 113–159.

Jörg Kienzle (Joerg.Kienzle@mcgill.ca) is the corresponding author for this article.

DOI:10.1145/3378931

To view the accompanying paper, visit doi.acm.org/10.1145/3378933 rh

Technical Perspective
A Perspective on Pivot Tracing

By Rebecca Isaacs

DISTRIBUTED SYSTEMS ARE difficult to manage at the best of times: diagnosis, debugging, capacity planning, and configuration of runtime properties like timeouts or service-level objective (SLO) thresholds, are made more challenging by the extra complexity that arises from distribution. Throw into the mix that a single request will be serviced by multiple independent microservices, and these challenges compose and multiply.

Yet this type of serving environment is completely normal—just about every online service uses a collection of distinct, communicating functions to fulfil each user request. For example, the sale of a single item on a shopping site might involve an authentication service, a bot detection service, an inventory management service, and a payments service, each of which will be sharded *N* ways and likely using a caching layer in front of (distributed) persistent storage. With distribution comes scale: requests consisting of hundreds, or even thousands, of nested RPCs are not unusual in Web services. Standard mechanisms for batching, pipelining, concurrency, fault tolerance, hedging of requests, load balancing, and other such in-band, dynamic, control systems further exacerbate the difficulties of understanding system behavior.

In such an environment, how do service providers debug their systems? If the shopping cart checkout request latency exceeds the 99th percentile, how can we identify which microservice was responsible and why? One important tool for tackling this kind of problem, on par with, and complementary to logs, counters, and metrics, is tracing.

A typical end-to-end request tracing system relies on the RPC subsystem to propagate a unique request identifier between microservices, and thus tie together causally related service invocations. A trace will also capture metadata about the request, collected at each hop along the way—details like the URI string or a client identifier, the name and IP address of each host, and often performance metrics such as how much CPU the request consumed at each component.

This design is simple but has an inherent tension between generality and cost. Most tracing systems pick a point in this trade-off space in which every service instance generates records locally and transmits them directly to a collector that groups records from across the system by trace identifier. With this approach, the set of queries that can be run (offline) against the traces is unconstrained, but the system can produce vast amounts of data with correspondingly high cost. As a result, requests are traced at a low sampling rate (.01% or lower is typical in production), which means rare events, often critical for detection and diagnosis of problems, may be missed.

Pivot Tracing, the system described in the following paper, chooses a different trade-off. Instead of eagerly handing trace records off to a collector for long-term storage and future processing, it installs continuous queries, on demand, inside the distributed system itself, and dynamically enables instrumentation at *tracepoints* to record exactly the information needed to answer the currently active queries. By this design, Pivot Tracing favors specificity in return for low cost, while removing the need to down-sample requests. A particularly appealing aspect is the query language consists of familiar relational operators over streams of tuples, extended to specify joins between causally related events (with the happened-before operator), which enables some neat "pivot table" styles of analysis across data generated by different types of components and at different points in a request's lifetime.

Pivot Tracing favors specificity in return for low cost; removing the need to down-sample requests.

The paper also proposes another interesting twist to the conventional approach with the notion of *baggage*. With continuously executing queries running in-situ, where does the input data come from? How does information generated at one component (say, the name of the client application) reach a query running on a different component (say, a file system node that will join this name with a count of bytes read)? Baggage is a container for propagating the causally related "stream of tuples" in-band, along with the request itself. This is an intriguing design choice because the propagation and query processing costs are borne by the live system itself, and thus have to be managed carefully, but in return we have a flexible and powerful tool for interactive debugging of a complex, distributed system.

Although tracing systems have been around over 20 years, their use in production has only become mainstream in the last few years. Tracing support is now offered by most cloud providers to their customers, and there is an active open source community, defining standards such as OpenTracing (with baggage now part of the specification), OpenCensus, and OpenZipkin. Nevertheless, there is still a great deal of unrealized potential in tracing for sophisticated debugging and rich analytical insights to help manage complex distributed systems, and this thought-provoking paper makes a timely contribution to the conversation. C

Rebecca Isaacs is a software engineer at Twitter in San Francisco, CA, USA.

The views here are the author's own, and do not reflect the views of Twitter.

DOI:10.1145/3378933

Pivot Tracing: Dynamic Causal Monitoring for Distributed Systems

By Jonathan Mace, Ryan Roelke, and Rodrigo Fonseca

Abstract

Monitoring and troubleshooting distributed systems are notoriously difficult; potential problems are complex, varied, and unpredictable. The monitoring and diagnosis tools commonly used today—logs, counters, and metrics—have two important limitations: what gets recorded is defined a priori, and the information is recorded in a component- or machine-centric way, making it extremely hard to correlate events that cross these boundaries. This paper presents Pivot Tracing, a monitoring framework for distributed systems that addresses both limitations by combining dynamic instrumentation with a novel relational operator: the happened-before join. Pivot Tracing gives users, at runtime, the ability to define arbitrary metrics at one point of the system, while being able to select, filter, and group by events meaningful at other parts of the system, even when crossing component or machine boundaries. Pivot Tracing does not correlate cross-component events using expensive global aggregations, nor does it perform offline analysis. Instead, Pivot Tracing directly correlates events as they happen by piggybacking metadata alongside requests as they execute. This gives Pivot Tracing low runtime overhead—less than 1% for many cross-component monitoring queries.

1. INTRODUCTION

Monitoring and troubleshooting distributed systems are hard. The potential problems are myriad: hardware and software failures, misconfigurations, hot spots, aggressive tenants, or even simply unrealistic user expectations. Despite the complex and unpredictable nature of these problems, most of the monitoring and diagnosis tools commonly used today—logs, counters, and metrics—have at least two fundamental limitations: what gets recorded is defined a priori, at development or deployment time, and the information is captured in a component- or machine-centric way, making it extremely difficult to correlate events that cross these boundaries.

While there has been great progress in using machine learning techniques and static analysis to improve the quality of logs and their use in troubleshooting,[16] logs carry an inherent tradeoff between recall and overhead, as what gets logged must be defined a priori.

Addressing this limitation, dynamic instrumentation systems such as Fay[7] and DTrace[4] enable the diagnosis of unanticipated performance problems in production systems[3] by providing the ability to select, at runtime, which of a large number of tracepoints to activate. Dynamic instrumentation, however, is still limited when it comes to correlating events that cross address-space or OS-instance boundaries. This limitation is fundamental, as neither Fay nor DTrace can affect the monitored system to propagate the monitoring context across these boundaries.

In this paper, we present Pivot Tracing, a monitoring framework that combines dynamic instrumentation with causal tracing techniques[8, 23] to fundamentally increase the power and applicability of either technique. Pivot Tracing gives operators and users, at runtime, the ability to obtain an almost arbitrary metric at one point of the system, while selecting, filtering, and grouping by causally preceding events from other parts of the system, even when crossing component or machine boundaries. Pivot Tracing exposes these features by modeling system events as the tuples of a streaming, distributed data set. Users can write relational queries about system events using Pivot Tracing's LINQ-like query language. Pivot Tracing compiles queries into efficient instrumentation code and dynamically installs the code at the sources of events specified in the query, returning a streaming data set of results to the user.

The key contribution of Pivot Tracing is the "happened-before join" operator, $\overset{\rightarrow}{\bowtie}$, that enables queries to be contextualized by Lamport's happened-before relation, →.[15] Using $\overset{\rightarrow}{\bowtie}$, queries can group and filter events based on properties of any events that causally precede them in an execution.

To track the happened-before relation between events, Pivot Tracing borrows from causal tracing techniques, and utilizes a generic metadata propagation mechanism for passing partial query execution state along the execution path of each request. This enables inline evaluation of joins during request execution, drastically mitigating query overhead and avoiding the scalability issues of global evaluation.

We have implemented and open-sourced a prototype of Pivot Tracing for Java-based systems, and instrumented a variety of distributed systems including HDFS, HBase, MapReduce, Tez, YARN, and Spark. In our full evaluation,[16] we show that Pivot Tracing can effectively identify a diverse range of root causes such as software bugs, misconfiguration, and limping hardware. We show that Pivot Tracing is dynamic, extensible to new kinds of analysis, and enables cross-tier analysis between inter-operating applications with low execution overhead.

The original version of this paper was published in *Proceedings of the 25th Symposium on Operating Systems Principles* (2015), ACM. New York, NY, 378–393.

2. MOTIVATION

2.1. Pivot Tracing in action

In this section, we motivate Pivot Tracing with a monitoring task on the Hadoop stack. Our goal here is to demonstrate some of what Pivot Tracing can do, and we leave details of its design and implementation to Sections 3 and 4, respectively.

Suppose we are managing a cluster of eight machines and want to know how disk bandwidth is being used across the cluster. On these machines, we are simultaneously running clients with workloads in HBase, HDFS, and MapReduce. It suffices to know that HBase is a distributed database that accesses data through HDFS, a distributed file system. MapReduce, in addition to accessing data through HDFS, also accesses the disk directly to perform external sorts and to shuffle data between tasks. Figure 1 depicts this scenario along with the following client applications:

FSread4M	Random closed-loop 4MB HDFS reads
FSread64M	Random closed-loop 64MB HDFS reads
Hget	10kB row lookups in a large HBase table
Hscan	4MB table scans of a large HBase table
MRsort10G	MapReduce sort job on 10GB of input data
MRsort100G	MapReduce sort job on 100GB of input data

By default, the systems expose a few metrics for disk consumption, such as disk read throughput aggregated by each HDFS DataNode. To reproduce this metric with Pivot Tracing, we define a *tracepoint* for the DataNodeMetrics class, in HDFS, to intercept the incrBytesRead(int delta) method. A tracepoint is a location in the application source code where instrumentation can run, cf. Section 3. We then run the following query, in Pivot Tracing's LINQ-like query language[17]:

```
Q1 : From incr In DataNodeMetrics.incrBytesRead
     GroupBy incr.host
     Select incr.host, SUM(incr.delta)
```

This query causes each machine to aggregate the delta argument each time incrBytesRead is invoked, grouping by the host name. Each machine reports its local aggregate every second, from which we produce the time series in Figure 2a.

Things get more interesting, though, if we wish to measure the HDFS usage of each of our client applications. HDFS only has visibility of its direct clients, and thus an aggregate view of all HBase and all MapReduce clients. At best, applications must estimate throughput client side. With Pivot Tracing, we define tracepoints for the client protocols of HDFS (DataTransferProtocol), HBase (ClientService), and MapReduce (ApplicationClientProtocol), and use the name of the client process as the group by key for the query. Figure 2b shows the global HDFS read throughput of each client application, produced by the following query:

```
Q2 : From incr In DataNodeMetrics.incrBytesRead
     Join cl In First(ClientProtocols) On cl -> incr
     GroupBy cl.procName
     Select cl.procName, SUM(incr.delta)
```

The -> symbol indicates a happened-before join. Pivot Tracing's implementation will record the process name the first time the request passes through any client protocol method and propagate it along the execution. Then, whenever the execution reaches incrBytesRead on a DataNode, Pivot Tracing will emit the bytes read or written, grouped by the recorded name. This query exposes information about client disk throughput that cannot currently be exposed by HDFS.

Figure 2c demonstrates the ability for Pivot Tracing to group metrics along arbitrary dimensions. It is generated

Figure 1. Six client workloads access the disks on eight cluster machines indirectly via HBase, a distributed database; HDFS, a distributed file system; and MapReduce, a data processing framework.

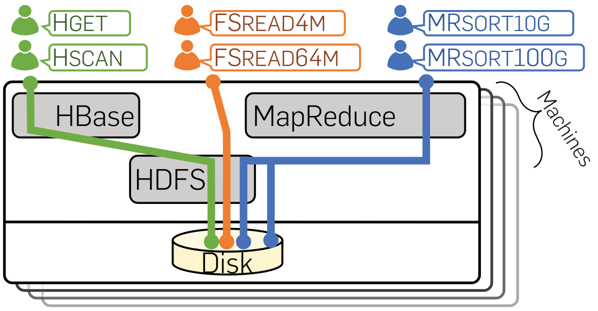
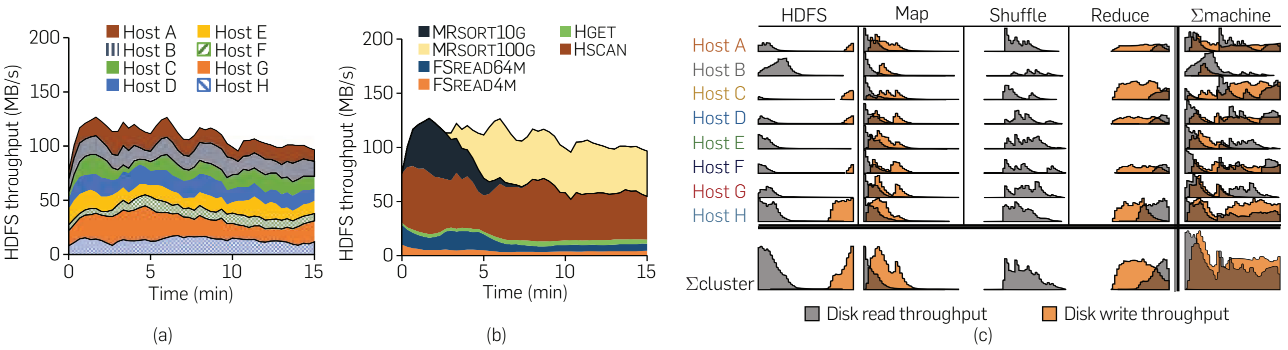


Figure 2. In this example, Pivot Tracing exposes a low-level HDFS metric grouped by client identifiers from other applications. Pivot Tracing can expose arbitrary metrics at one point of the system, while being able to select, filter, and group by events meaningful at other parts of the system, even when crossing component or machine boundaries. (a) HDFS DataNode throughput per machine from instrumented DataNodeMetrics. (b) HDFS DataNode throughput grouped by high-level client application. (c) Pivot table showing disk read and write sparklines for MRsort10g. Rows group by host machine; columns group by source process. Bottom row and right column show totals, and bottom-right corner shows grand total.



by two queries similar to Q2 that instrument Java's FileInputStream and FileOutputStream, still joining with the client process name. We show the per-machine, per-application disk read and write throughput of MRSORT10G from the same experiment. This figure resembles a pivot table, where summing across rows yields per-machine totals, summing across columns yields per-system totals, and the bottom right corner shows the global totals. In this example, the client application presents a further dimension along which we could present statistics.

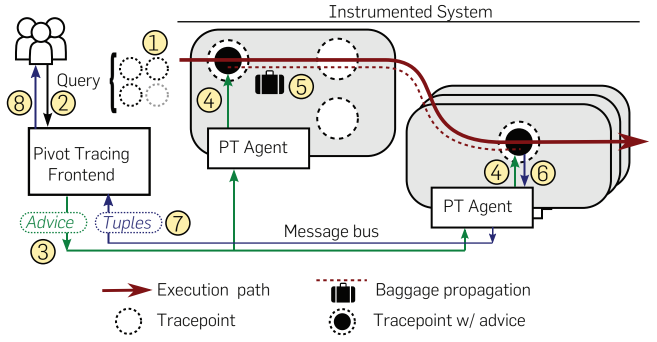
Query Q1 above is processed locally, while query Q2 requires the propagation of information from client processes to the data access points. Pivot Tracing's query optimizer installs dynamic instrumentation where needed, and determines when such propagation must occur to process a query. The out-of-the box metrics provided by HDFS, HBase, and MapReduce cannot provide analyses like those presented here. Simple correlations—such as determining *which* HDFS datanodes were read from by a high-level client application—are not typically possible. Metrics are ad hoc between systems; HDFS sums IO bytes, while HBase exposes operations per second. There is very limited support for cross-tier analysis: MapReduce simply counts global HDFS input and output bytes; HBase does not explicitly relate HDFS metrics to HBase operations.

2.2. Pivot Tracing overview

Figure 3 presents a high-level overview of how Pivot Tracing enables queries such as Q2. We refer to the numbers in the figure (e.g., ①) in our description. Full support for Pivot Tracing in a system requires two basic mechanisms: dynamic code injection and causal metadata propagation.

Queries in Pivot Tracing refer to variables exposed by one or more *tracepoints*—places in the system where Pivot Tracing can insert instrumentation. Tracepoint definitions are not part of the system code, but are rather instructions on where and how to change the system to obtain the exported identifiers. Tracepoints in Pivot Tracing are similar to pointcuts from aspect-oriented programming,[14] and can refer to arbitrary interface/method signature combinations. Tracepoints are defined by someone with knowledge of the system, maybe a developer or expert operator, and define the vocabulary for queries (①). They can be defined and installed at any point in time, and can be shared and disseminated.

Figure 3. Pivot Tracing overview (Section 2.2).



Pivot Tracing models system events as tuples of a streaming, distributed dataset. Users submit relational queries over this dataset (②), which get compiled to an intermediate representation called *advice* (③). Advice uses a small instruction set to process queries, and maps directly to code that local Pivot Tracing agents install dynamically at relevant tracepoints (④). Later, requests executing in the system invoke the installed advice each time their execution reaches the tracepoint.

We distinguish Pivot Tracing from prior work by supporting *joins* between events that occur within and across process, machine, and application boundaries. The efficient implementation of the happened before join requires advice in one tracepoint to send information along the execution path to advice in subsequent tracepoints. This is done through a new *baggage* abstraction, which uses causal metadata propagation (⑤). In query Q2, for example, cl.procName is packed in the first invocation of the ClientProtocols tracepoint, to be accessed when processing the incrBytesRead tracepoint.

Advice in some tracepoints also emit tuples (⑥), which get aggregated locally and then finally streamed to the client over a message bus (⑦ and ⑥).

2.3. Monitoring and troubleshooting challenges

Pivot Tracing addresses two main challenges in monitoring and troubleshooting. First, when the choice of what to record about an execution is made a priori, there is an inherent tradeoff between recall and overhead. Second, to diagnose many important problems one needs to correlate and integrate data that crosses component, system, and machine boundaries.

One size does not fit all. Problems in distributed systems are complex, varied, and unpredictable. By default, the information required to diagnose an issue may not be reported by the system or contained in system logs. Current approaches tie logging and statistics mechanisms into the development path of products, where there is a mismatch between the expectations and incentives of the developer and the needs of operators and users. Panelists at SLAML[2] discussed the important need to "close the loop of operations back to developers." According to Yuan et al.,[25] regarding diagnosing failures, "(. . .) existing log messages contain too little information. Despite their widespread use in failure diagnosis, it is still rare that log messages are systematically designed to support this function."

This mismatch can be observed in the many issues raised by users on Apache's issue trackers[16] requesting new metrics, changes to aggregation methods, or new breakdowns of existing metrics. Many issues remain unresolved due to developer pushback or inertia.

Eventually, applications may be updated to record more information, but this has effects both in performance and information overload. Users must pay the performance overheads of any systems that are enabled by default, regardless of their utility. For example, HBase SchemaMetrics were introduced to aid developers, but all users of HBase pay the 10% performance overhead they incur.[10] The HBase user guide carries the following warning for users wishing to integrate with Ganglia: "By default, HBase emits a large number of metrics per region server. Ganglia may have difficulty processing all these metrics. Consider increasing the capacity of the Ganglia server or reducing the number of metrics emitted by HBase."

The glut of recorded information presents a "needle-in-a-haystack" problem to users[21]; while a system may expose information relevant to a problem, for example, in a log, extracting this information requires system familiarity developed over a long period of time. For example, Mesos cluster state is exposed via a single JSON endpoint and can become massive, even if a client only wants information for a subset of the state.[16]

Dynamic instrumentation frameworks such as Fay,[7] DTrace,[4] and SystemTap[20] address these limitations, by allowing almost arbitrary instrumentation to be installed dynamically at runtime, and have proven extremely useful in the diagnosis of complex and subtle system problems.[3] Because of their side-effect-free nature, however, they are limited in the extent to which probes may share information with each other. In Fay, only probes in the same address space can share information, while in DTrace the scope is limited to a single operating system instance.

Crossing boundaries. This brings us to the second challenge Pivot Tracing addresses. In multi-tenant, multi-application stacks, the root cause and symptoms of an issue may appear in different processes, machines, and application tiers, and may be visible to different users. A user of one application may need to relate information from some other dependent application in order to diagnose problems that span multiple systems. For example, HBASE-4145[9] outlines how MapReduce lacks the ability to access HBase metrics on a per-task basis, and that the framework only returns aggregates across all tasks. MESOS-1949[18] outlines how the executors for a task do not propagate failure information, so diagnosis can be difficult if an executor fails. In discussion the developers note: "The actually interesting/useful information is hidden in one of four or five different places, potentially spread across as many different machines. This leads to unpleasant and repetitive searching through logs looking for a clue to what went wrong. (...) There's a lot of information, that is, hidden in log files and is very hard to correlate."

Prior research has presented mechanisms to observe or infer the relationship between events and studies of logging practices conclude that end-to-end tracing would be helpful in navigating the logging issues they outline.[16]

A variety of these mechanisms have also materialized in production systems, for example, Google's Dapper,[23] Apache's HTrace,[1] and Twitter's Zipkin.[24] These approaches can obtain richer information about particular executions than component-centric logs or metrics alone, and have found uses in troubleshooting, debugging, performance analysis and anomaly detection, for example. However, most of these systems record or reconstruct traces of execution for offline analysis, and thus share the problems above with the first challenge, concerning what to record.

3. DESIGN

We now detail the fundamental concepts and mechanisms behind Pivot Tracing. Pivot Tracing is a dynamic monitoring and tracing framework for distributed systems. At a high level, it aims to enable flexible runtime monitoring by correlating metrics and events from arbitrary points in the system. The challenges outlined in Section 2 motivate the following high-level design goals:

1. Dynamically configure and install monitoring at runtime.
2. Low system overhead to enable "always on" monitoring.
3. Capture causality between events from multiple processes and applications.

Tracepoints. Tracepoints provide the system-level entry point for Pivot Tracing queries. A tracepoint typically corresponds to some event: a user submits a request, a low-level IO operation completes, an external RPC is invoked, etc. A tracepoint identifies one or more locations in the system code where Pivot Tracing can install and run instrumentation, such as the name of a method. Since Pivot Tracing uses dynamic instrumentation to install queries, tracepoints do not need to be defined a priori, nor do they require a priori modification of system code; they are simply references to locations in the source code. A tracepoint is only materialized once a query is installed that references it. Tracepoints export named variables that can be accessed by instrumentation, such as method arguments or local variables, as well as several default variables: host, timestamp, process id, process name, and the tracepoint definition.

Whenever execution of the system reaches a tracepoint, any instrumentation configured for that tracepoint will be invoked, generating a tuple with its exported variables. These are then accessible to any instrumentation code installed at the tracepoint.

Query language. Pivot Tracing enables users to express high-level queries about the variables exported by one or more tracepoints. We abstract tracepoint invocations as streaming datasets of tuples; Pivot Tracing queries are therefore relational queries across the tuples of several such datasets.

To express queries, Pivot Tracing provides a parser for LINQ-like text queries such as those outlined in Section 2. Table 1 outlines the query operations supported by Pivot Tracing. Pivot Tracing supports several typical operations including projection (Π), selection (σ), grouping (G), and aggregation (A). Pivot Tracing aggregators include Count, Sum, Max, Min, and Average. Pivot Tracing also defines the temporal filters MostRecent, MostRecentN, First, and FirstN, to take the 1 or N most or least recent events. Finally, Pivot Tracing introduces the *happened-before join* query operator ($\overrightarrow{\bowtie}$).

Happened-before joins. A key contribution of Pivot Tracing is the happened-before join query operator. Happened-before join enables the tuples from two Pivot Tracing queries to be joined based on Lamport's happened before relation, $\rightarrow$.[15] For events a and b occurring anywhere in the system, we say that a happened before b and write $a \rightarrow b$ if the occurrence of event a causally preceded the occurrence of event b and they occurred as part of the execution of the same request.[a] If a

[a] This definition does not capture all possible causality, including when events in the processing of one request could influence another, but could be extended if necessary.

Table 1. Operations supported by the Pivot Tracing query language.

Operation	Description	Example
From	Use input tuples from a set of tracepoints	**From** e **In** RPCs
Union (∪)	Union events from multiple tracepoints	**From** e **In** DataRPCs, ControlRPCs
Selection (σ)	Filter only tuples that match a predicate	**Where** e.Size < 10
Projection (Π)	Restrict tuples to a subset of fields	**Select** e.User, e.Host
Aggregation (A)	Aggregate tuples	**Select** **SUM**(e.Cost)
GroupBy (G)	Group tuples based on one or more fields	**GroupBy** e.User
GroupBy aggregation (GA)	Aggregate tuples of a group	**Select** e.User, **SUM**(e.Cost)
Happened-before join ($\overrightarrow{\bowtie}$)	Happened-before join tuples from another query	**Join** d **In** Disk **On** d **->** e
	Happened-before join a subset of tuples	**Join** d **In** **MostRecent**(Disk) **On** d **->** e

and b are not part of the same execution, then $a \nrightarrow b$ if the occurrence of a did not lead to the occurrence of b, then $a \nrightarrow b$ (e.g., they occur in two parallel threads of execution that do not communicate); and if $a \rightarrow b$ then $b \nrightarrow a$.

For any two queries Q_1 and Q_2, the happened-before join $Q_1 \overrightarrow{\bowtie} Q_2$ produces tuples t_1t_2 for all $t_1 \in Q_1$ and $t_2 \in Q_2$ such that $t_1 \rightarrow t_2$. That is, Q_1 produced t_1 before Q_2 produced tuple t_2 in the execution of the same request. Figure 4 shows an example execution triggering tracepoints A, B, and C several times, and outlines the tuples that would be produced for this execution by different queries.

Query Q2 in Section 2 demonstrates the use of happened-before join. In the query, tuples generated by the disk IO tracepoint DataNodeMetrics.incrBytesRead are joined to the first tuple generated by the ClientProtocols tracepoint.

Happened-before join substantially improves our ability to perform root cause analysis by giving us visibility into the relationships *between* events in the system. The happened-before relationship is fundamental to a number of prior approaches in root cause analysis.[16] Pivot Tracing is designed to efficiently support happened-before joins, but does not optimize more general joins such as equijoins (⋈).

Advice. Pivot Tracing queries compile to an intermediate representation called *advice*. Advice specifies the operations to perform at each tracepoint used in a query, and eventually materializes as monitoring code installed at those tracepoints (Section 4). Advice has several operations for manipulating tuples through the tracepoint-exported variables, and evaluating $\overrightarrow{\bowtie}$ on tuples produced by other advice at prior tracepoints in the execution.

Table 2 outlines the advice API. Observe creates a tuple from exported tracepoint variables. Unpack retrieves tuples generated by other advice at other tracepoints prior in the execution. Unpacked tuples can be joined to the observed tuple, that is, if t_o is observed and t_{u1} and t_{u2} are unpacked, then the resulting tuples are t_ot_{u1} and t_ot_{u2}. Tuples created by this advice can be discarded (Filter), made available to advice at other tracepoints later in the execution (Pack), or output for global aggregation (Emit). Both Pack and Emit can group tuples based on matching fields, and perform simple aggregations such as SUM and COUNT. Pack also has the following special cases: FIRST packs the first tuple encountered and ignores subsequent tuples; RECENT packs only the most recent tuple, overwriting existing tuples. FIRSTN and RECENTN generalize this to N tuples. The advice API is expressive but restricted enough to provide some safety guarantees. In particular, advice code has no jumps or recursion, and is guaranteed to terminate.

Figure 4. An example execution that triggers tracepoints A, B, and C several times. We show several Pivot Tracing queries and the tuples that would result for each.

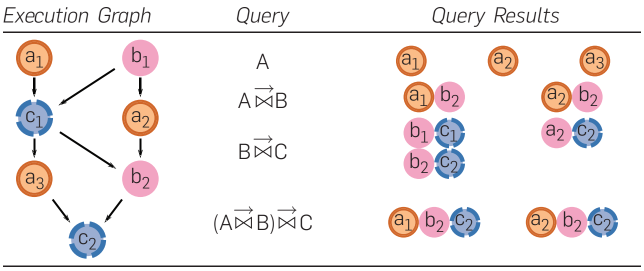

Execution Graph
Query
Query Results
a_1
b_1
c_1
a_2
a_3
b_2
c_2
A
$A \overrightarrow{\bowtie} B$
$B \overrightarrow{\bowtie} C$
$(A \overrightarrow{\bowtie} B) \overrightarrow{\bowtie} C$
a_1 a_2 a_3
a_1 b_2 a_2 b_2
b_1 c_1 a_2 c_2
b_2 c_2
a_1 b_2 c_2 a_2 b_2 c_2

Table 2. Primitive operations supported by Pivot Tracing advice for generating and aggregating tuples as defined in Section 3.

Operation	Description
Observe	Construct a tuple from variables exported by a tracepoint
Unpack	Retrieve one or more tuples from prior advice
Filter	Evaluate a predicate on all tuples
Pack	Make tuples available for use by later advice
Emit	Output a tuple for global aggregation

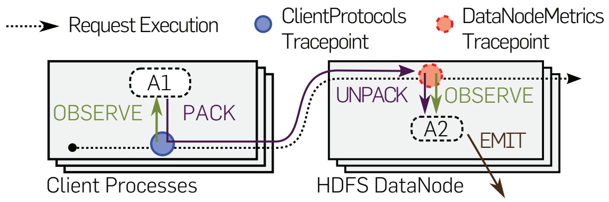
Query Q2 in Section 2 compiles to advice A1 and A2 for ClientProtocols and DataNodeMetrics, respectively:

```
A1 : OBSERVE procName          A2 : OBSERVE delta
     PACK-FIRST procName            UNPACK procName
                                    EMIT procName, SUM(delta)
```

Figure 5 shows how this advice and the tracepoints interact with the execution of requests in the system. First, when a request's execution reaches ClientProtocols, A1 is invoked, which observes and packs a single valued tuple containing the process name. Then, when execution reaches DataNodeMetrics, A2 is invoked, which unpacks the process name, observes the value of delta, then emits a joined tuple.

To compile a query to advice, we instantiate one advice specification for a From clause and add an Observe operation for the tracepoint variables used in the query. For each Join clause, we add an Unpack operation for the variables that originate from the joined query. We recursively generate

Figure 5. Advice generated for Q2 from Section 2: A1 observes and packs procName; A2 unpacks procName, observes delta, and emits (procName, SUM(delta)).



advice for the joined query, and append a Pack operation at the end of its advice for the variables that we unpacked. Where directly translates to a Filter operation. We add an Emit operation for the output variables of the query, restricted according to any Select clause. Aggregate, GroupBy, and GroupByAggregate are all handled by Emit and Pack.

Baggage. Pivot Tracing enables inexpensive happened-before joins by providing the *baggage* abstraction. Baggage is a per-request container for tuples, that is, propagated alongside a request as it traverses thread, application, and machine boundaries. Pack and Unpack store and retrieve tuples from the current request's baggage. Tuples follow the request's execution path and therefore explicitly capture the happened-before relationship.

Baggage is a generalization of end-to-end metadata propagation techniques outlined in prior work such as X-Trace[8] and Dapper.[23] Using baggage, Pivot Tracing efficiently evaluates happened-before joins in situ during the execution of a request.

Tuple aggregation and query optimization. To reduce the volume of emitted tuples, Pivot Tracing performs intermediate aggregation for queries containing Aggregate or GroupByAggregate. Pivot Tracing aggregates the emitted tuples within each process and reports results globally at a regular interval, for example, once per second. Process-level aggregation substantially reduces traffic for emitted tuples; Q2 from Section 2 is reduced from approximately 600 to 6 tuples per second from each DataNode. Pivot Tracing also rewrites queries to minimize the number of tuples that are packed during a request's execution, using the same query rewriting rules described by Fay[7] that push projection, selection, and aggregation terms as close as possible to source tracepoints. We extend these query rewriting rules[16] to add further optimizations for happened-before joins.

4. IMPLEMENTATION

We have implemented a Pivot Tracing prototype in Java and applied Pivot Tracing to several open-source systems from the Hadoop ecosystem. Pivot Tracing source code and the instrumented systems are publicly available from the Pivot Tracing project website.[b]

Agent. A Pivot Tracing agent thread runs in every Pivot Tracing-enabled process and awaits instruction via central pub/sub server to weave advice to tracepoints. Tuples emitted by advice are accumulated by the local Pivot Tracing agent, which performs partial aggregation of tuples according to their source query. Agents publish partial query results back to the user at a configurable interval—by default, 1 s.

Dynamic instrumentation. Our prototype weaves advice at runtime, providing dynamic instrumentation similar to that of DTrace[4] and Fay.[7] Java version 1.5 onwards supports dynamic method body rewriting via the java.lang.instrument package. The Pivot Tracing agent pro-grammatically rewrites and reloads class bytecode from within the process using Javassist.[5] To weave advice, we rewrite method bodies to add advice invocations at the locations defined by the tracepoint. Our prototype supports tracepoints at the entry, exit, or exceptional return of any method. Tracepoints can also be inserted at specific line numbers.

To define a tracepoint, users specify a class name, method name, method signature, and weave location. Pivot Tracing also supports pattern matching, for example, all methods of an interface on a class. This feature is modeled after *pointcuts* from AspectJ.[13] Pivot Tracing supports instrumenting privileged classes (e.g., FileInputStream in Section 2) by providing an optional agent that can be placed on Java's boot classpath.

Pivot Tracing only makes system modifications when advice is woven into a tracepoint, so inactive tracepoints incur no overhead. Executions that do not trigger the tracepoint are unaffected by Pivot Tracing. Pivot Tracing has a zero-probe effect: methods are unmodified by default, so trace-points impose truly zero overhead until advice is woven into them.

Baggage. Our implementation of baggage uses thread-local variables for storing per-request baggage instances. At the beginning of a request, we instantiate empty baggage in the thread-local variable; at the end of the request, we clear the baggage from the thread-local variable. The baggage API can get or set tuples for a query and at any point in time baggage can be retrieved for propagation to another thread or serialization onto the network. To support multiple queries simultaneously, queries are assigned unique IDs and tuples are packed and unpacked based on this ID.

Hadoop instrumentation. Pivot Tracing relies on developers to implement Baggage propagation when a request crosses thread, process, or asynchronous execution boundaries. We have implemented this propagation in several open-source systems that are widely used in production today: HDFS, HBase, MapReduce, Tez, YARN, and Spark. To propagate baggage across remote procedure calls, we manually extended the protocol definitions of the systems. To propagate baggage across execution boundaries within individual processes we implemented AspectJ[13] instrumentation to automatically modify common interfaces (Thread, Runnable, Callable, and Queue). Each system required between 50 and 200 lines of manual code modification. Once modified, these systems could support arbitrary Pivot Tracing queries without further modification.

5. EVALUATION

In this section, we evaluate Pivot Tracing with a case study in the context of the Hadoop Distributed FileSystem[22] (HDFS).[c] HDFS is a distributed file system comprising a central NameNode process that manages filesystem metadata, and

[b] http://pivottracing.io.

[c] We refer the reader to the full evaluation[16] for other case studies and evaluation of Pivot Tracing overheads.

multiple DataNode processes running across a cluster that store replicated file blocks. We describe our discovery of a replica selection bug in HDFS that resulted in uneven distribution of load to replicas. After identifying the bug, we found that it had been recently reported and subsequently fixed in an upcoming HDFS version.[11]

HDFS provides file redundancy by decomposing files into blocks and replicating each block onto several machines (typically 3). A client can read any replica of a block and does so by first contacting the NameNode to find replica hosts (invoking GetBlockLocations), then selecting the closest replica as follows: (1) read a local replica, (2) read a rack-local replica, and (3) select a replica at random. We discovered a bug whereby rack-local replica selection always follows a global static ordering due to two conflicting behaviors: the HDFS client does not randomly select between replicas; and the HDFS NameNode does not randomize rack-local replicas returned to the client. The bug results in heavy load on some hosts and near zero load on others.

In this scenario, we ran 96 stress test clients on an HDFS cluster of eight DataNodes and one NameNode. Each machine has identical hardware specifications; 8 cores, 16GB RAM, and a 1Gbit network interface. On each host, we ran a process called StressTest that used an HDFS client to perform closed-loop random 8kB reads from a dataset of 10,000 128MB files with a replication factor of 3. Our queries use tracepoints from both client and server RPC protocol implementations of the HDFS DataNode DataTransferProtocol and NameNode GetBlockLocations client protocol.

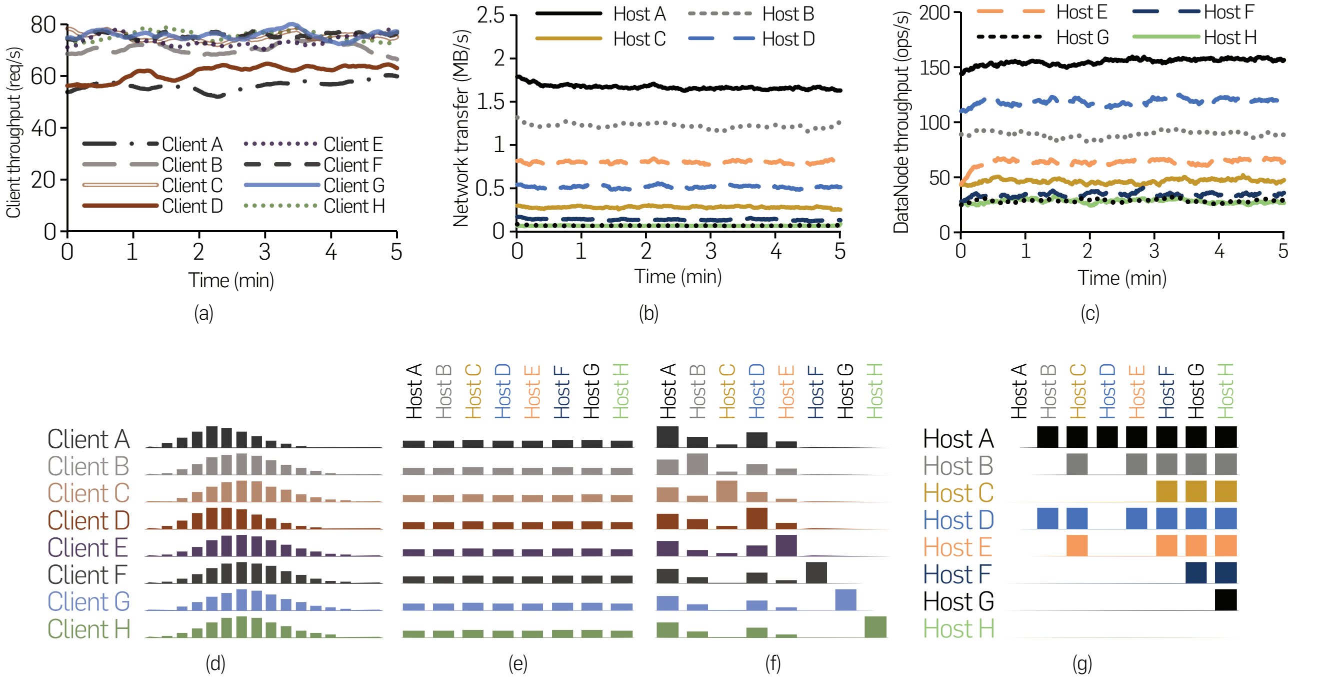
Our investigation of the bug began when we noticed that the stress test clients on hosts A and D had consistently lower request throughput than clients on other hosts, shown in Figure 6a, despite identical machine specifications and setup. We first checked machine level resource utilization on each host, which indicated substantial variation in the network throughput (Figure 6b). We began our diagnosis with Pivot Tracing by first checking to see whether an imbalance in HDFS load was causing the variation in network throughput. The following query installs advice at a DataNode tracepoint, that is, invoked by each incoming RPC:

```
Q3 : From dnop In DN.DataTransferProtocol
     GroupBy dnop.host
     Select dnop.host, COUNT
```

Figure 6c plots the results of this query, showing the HDFS request throughput on each DataNode. It shows that DataNodes on hosts A and D in particular have substantially higher request throughput than others—host A has on average 150 ops/s, while host H has only 25 ops/s. This behavior was unexpected given that our stress test clients are supposedly reading files uniformly at random. Our next query installs advice in the stress test clients and on the HDFS NameNode, to correlate each read request with the client that issued it:

```
Q4 : From getloc In NN.GetBlockLocations
     Join st In StressTest.DoNextOp On st -> getloc
     GroupBy st.host, getloc.src
     Select st.host, getloc.src, COUNT
```

Figure 6. Pivot Tracing query results leading to our discovery of HDFS-6268.[11] Faulty replica selection logic led clients to prioritize the replicas hosted by particular DataNodes: host A was always preferred over other hosts if it held a replica; host D was always preferred, except if host A held a replica; etc. The increased load to host A DataNode reduced the throughput of co-located client A. (a) Clients on Hosts A and D experience reduced workload throughput. (b) Network transfer is skewed across machines. (c) HDFS DataNode throughput is skewed across machines. (d) Observed HDFS file read distribution (row) per client (col). (e) Frequency each client (row) sees each DataNode (col) as a replica location. (f) Frequency each client (row) subsequently selects each DataNode (col). (g) Observed frequency of choosing one replica host (row) over another (col).



This query counts the number of times each client reads each file. In Figure 6d, we plot the distribution of counts over a 5-min period for clients from each host. The distributions all fit a normal distribution and indicate that all of the clients are reading files uniformly at random. The distribution of reads from clients on A and D are skewed left, consistent with their overall lower read throughput.

Having confirmed the expected behavior of our stress test clients, we next checked to see whether the skewed datanode throughput was simply a result of skewed block placement across datanodes:

```
Q5 : From getloc In NN.GetBlockLocations
       Join st In StressTest.DoNextOp On st -> getloc
       GroupBy st.host, getloc.replicas
       Select st.host, getloc.replicas, COUNT
```

This query measures the frequency that each DataNode is hosting a replica for files being read. Figure 6e shows that, for each client, replicas are near-uniformly distributed across DataNodes in the cluster. These results indicate that clients have an equal opportunity to read replicas from each DataNode, yet, our measurements in Figure 6c clearly show that they do not. To gain more insight into this inconsistency, our next query relates the results from Figure 6e to those from Figure 6c:

```
Q6 : From DNop In DN.DataTransferProtocol
       Join st In StressTest.DoNextOp On st -> DNop
       GroupBy st.host, DNop.host
       Select st.host, DNop.host, COUNT
```

This query measures the frequency that each client selects each DataNode for reading a replica. We plot the results in Figure 6f and see that the clients are clearly favoring particular DataNodes. The strong diagonal is consistent with HDFS client preference for locally hosted replicas (39% of the time in this case). However, the expected behavior when there is not a local replica is to select a rack-local replica uniformly at random; clearly these results suggest that this was not happening.

Our final diagnosis steps were as follows. First, we checked to see *which* replica was selected by HDFS clients from the locations returned by the NameNode. We found that clients always selected the first location returned by the NameNode. Second, we measured the conditional probabilities that DataNodes precede each other in the locations returned by the NameNode. We issued the following query for the latter:

```
Q7 : From DNop In DN.DataTransferProtocol
       Join getloc In NN.GetBlockLocations
                        On getloc -> DNop
       Join st In StressTest.DoNextOp On st -> getloc
       Where st.host != DNop.host
       GroupBy DNop.host, getloc.replicas
       Select DNop.host, getloc.replicas, COUNT
```

This query correlates the DataNode, that is, selected with the other DataNodes also hosting a replica. We remove the interference from locally hosted replicas by *filtering* only the requests that do a non-local read. Figure 6g shows that host A was *always* selected when it hosted a replica; host D was always selected except if host A was also a replica, and so on. This should not have been the case; due to random replica selection, no host should have been preferred over any other host.

At this point in our analysis, we concluded that this behavior was quite likely to be a bug in HDFS. HDFS clients did not randomly select between replicas, and the HDFS NameNode did not randomize the rack-local replicas. We checked Apache's issue tracker and found that the bug had been recently reported and fixed in an upcoming version of HDFS.[11]

Application-level overhead. To estimate the impact of Pivot Tracing on application-level throughput and latency, we ran benchmarks from HiBench,[12] YCSB,[6] and HDFS DFSIO and NNBench benchmarks. Many of these benchmarks bottleneck on network or disk and we noticed no significant performance change with Pivot Tracing enabled.

To measure the effect of Pivot Tracing on CPU bound requests, we stress tested HDFS using requests derived from the HDFS NNBench benchmark: Read8k reads 8kB from a file; Open opens a file for reading; Create creates a file for writing; Rename renames an existing file. Read8kB is a DataNode operation and the others are NameNode operations. We compared the end-to-end latency of requests in unmodified HDFS to HDFS modified in the following ways: (1) with Pivot Tracing enabled, (2) propagating baggage containing one tuple but no advice installed, (3) propagating baggage containing 60 tuples (≈1kB) but no advice installed, and (4) with queries Q3—Q7 installed.

Table 3 shows that the application-level overhead with Pivot Tracing enabled is at most 0.3%. This overhead includes the costs of empty baggage propagation within HDFS, baggage serialization in RPC calls, and to run Java in debugging mode. The most noticeable overheads are incurred when propagating 60 tuples in the baggage, incurring 15.9% overhead for Open. Since this is a short CPU-bound request (involving a single read-only lookup), 16% is within reasonable expectations. RENAME does not trigger any advice for queries Q3–Q7, reflected by an overhead of just 0.3%.

6. DISCUSSION

Despite the advantages over logs and metrics for troubleshooting (Section 2), Pivot Tracing is not meant to replace all functions of logs, such as security auditing, forensics, or debugging.[19]

Pivot Tracing is designed to have similar per-query overheads to the metrics currently exposed by systems today. It is feasible for a system to have several Pivot Tracing queries on by default; these could be sensible defaults provided by developers, or custom queries installed by users to address their specific needs. We leave it to future work to explore the use of Pivot Tracing for automatic problem detection and exploration.

Table 3. Latency overheads for HDFS stress test with Pivot Tracing enabled, baggage propagation enabled, and queries enabled.

	Read8k (%)	Open (%)	Create (%)	Rename (%)
Unmodified	0	0	0	0
PivotTracing Enabled	0.3	0.3	<0.1	0.2
Baggage—1 Tuple	0.8	0.4	0.6	0.8
Baggage—60 Tuples	0.82	15.9	8.6	4.1
Queries Q3–Q7	1.5	4.0	6.0	0.3

While users are restricted to advice comprised of Pivot Tracing primitives, Pivot Tracing does not guarantee that its queries will be side-effect free, due to the way exported variables from tracepoints are currently defined. We can enforce that only trusted administrators define tracepoints and require that advice be signed for installation, but a comprehensive security analysis, including complete sanitization of tracepoint code is beyond the scope of this paper.

Even though we evaluated Pivot Tracing on an 8-node cluster in this paper, initial runs of the instrumented systems on a 200-node cluster with constant-size baggage being propagated showed negligible performance impact. It is ongoing work to evaluate the scalability of Pivot Tracing to larger clusters and more complex queries. Sampling at the advice level is a further method of reducing overhead that we plan to investigate.

We opted to implement Pivot Tracing in Java in order to easily instrument several popular open-source distributed systems written in this language. However, the components of Pivot Tracing generalize and are not restricted to Java—a query can span multiple systems written in different programming languages due to Pivot Tracing's platform-independent baggage format and restricted set of advice operations. In particular, it would be an interesting exercise to integrate the happened-before join with Fay or DTrace.

7. CONCLUSION

Pivot Tracing is the first monitoring system to combine dynamic instrumentation and causal tracing. Its novel happened-before join operator fundamentally increases the expressive power of dynamic instrumentation and the applicability of causal tracing. Pivot Tracing enables cross-tier analysis between any interoperating applications, with low execution overhead. Ultimately, its power lies in the uniform and ubiquitous way in which it integrates monitoring of a heterogeneous distributed system.

References

1. Apache HTrace. http://htrace.incubator.apache.org/. [Online; accessed March 2015]. (Section 2.3).
2. Bodik, P. Overview of the workshop of managing large-scale systems via the analysis of system logs and the application of machine learning techniques (SLAML'11). *SIGOPS Oper. Syst. Rev. 45*, 3 (2011), 20–22. (Section 2.3).
3. Cantrill, B. Hidden in plain sight. *ACM Queue 4*, 1 (Feb. 2006), 26–36. (Sections 1 and 2.3).
4. Cantrill, B., Shapiro, M.W., Leventhal, A.H. Dynamic instrumentation of production systems. In *USENIX Annual Technical Conference, General Track* (2004), pp. 15–28. (Sections 1, 2.3, and 4).
5. Chiba, S. Javassist: Java bytecode engineering made simple. *Java Developer's Journal 9*, 1 (2004). (Section 4).
6. Cooper, B.F., Silberstein, A., Tam, E., Ramakrishnan, R., Sears, R. Benchmarking cloud serving systems with ycsb. In *Proceedings of the 1st ACM Symposium on Cloud Computing* (2010). ACM, pp. 143–154. (Section 5).
7. Erlingsson, Ú., Peinado, M., Peter, S., Budiu, M., Mainar-Ruiz, G. Fay: Extensible distributed tracing from kernels to clusters. *ACM Trans. Comput. Syst. (TOCS) 30*, 4 (2012), 13. (Sections 1, 2.3, 3, and 4).
8. Fonseca, R., Porter, G., Katz, R.H., Shenker, S., Stoica, I. X-trace: A pervasive network tracing framework. In *Proceedings of the 4th USENIX Conference on Networked Systems Design & Implementation* (Berkeley, CA, USA, 2007), NSDI'07, USENIX Association. (Sections 1 and 3).
9. HBASE-4145 Provide metrics for HBASE client. https://issues.apache.org/jira/browse/HBASE-4145. [Online; accessed 25 February 2015]. (Section 2.3).
10. HBASE-8370 Report data block cache hit rates apart from aggregate cache hit rates. https://issues.apache.org/jira/browse/HBASE-8370. [Online; accessed 25 February 2015]. (Section 2.3).
11. HDFS-6268 Better sorting in NetworkTopology. pseudoSortByDistance when no local node is found. https://issues.apache.org/jira/browse/HDFS-6268. [Online; accessed 25 February 2015]. (Sections 1 and 3).
12. Huang, S., Huang, J., Dai, J., Xie, T., Huang, B. The hibench benchmark suite: Characterization of the mapreduce-based data analysis. In *New Frontiers in Information and Software as Services* (2010). IEEE, pp. 41–51. (Section 5).
13. Kiczales, G., Hilsdale, E., Hugunin, J., Kersten, M., Palm, J., Griswold, W.G. An Overview of AspectJ. In *European Conference on Object-Oriented Programming* (London, UK, 2001). Springer-Verlag, pp. 327–353. (Section 4).
14. Kiczales, G., Lamping, J., Mendhekar, A., Maeda, C., Lopes, C.V., Loingtier, J.-M., Irwin, J. Aspect-oriented programming. In *European Conference on Object-Oriented Programming*, LNCS 1241 (June 1997), Springer-Verlag. (Section 2.2).
15. Lamport, L. Time, clocks, and the ordering of events in a distributed system. *Commun. ACM 21*, 7 (1978), 558–565. (Sections 1 and 3).
16. Mace, J., Roelke, R., Fonseca, R. Pivot tracing: Dynamic causal monitoring for distributed systems. In *Proceedings of the 25th Symposium on Operating Systems Principles* (2015). ACM, pp. 378–393. (Sections 1, 2.5, and 3).
17. Meijer, E., Beckman, B., Bierman, G. Linq: Reconciling object, relations and xml in the.net framework. In *Proceedings of the 2006 ACM SIGMOD International Conference on Management of Data*, SIGMOD'06 (New York, NY, USA, 2006). ACM, pp. 706–706. (Section 2.1).
18. MESOS-1949 All log messages from master, slave, executor, etc. should be collected on a per-task basis. https://issues.apache.org/jira/browse/MESOS-1949. [Online; accessed 25 February 2015]. (Section 2.3).
19. Oliner, A., Ganapathi, A., Xu, W. Advances and challenges in log analysis. *Commun. ACM 55*, 2 (2012), 55–61. (Section 6).
20. Prasad, V., Cohen, W., Eigler, F.C., Hunt, M., Keniston, J., Chen, B. Locating system problems using dynamic instrumentation. In *2005 Ottawa Linux Symposium* (2005). (Section 2.3).
21. Rabkin, A., Katz, R.H. How hadoop clusters break. *IEEE Softw. 30*, 4 (2013), 88–94. (Section 2.3).
22. Shvachko, K., Kuang, H., Radia, S., Chansler, R. The Hadoop distributed file system. In *2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST)* (2010). IEEE, pp. 1–10. (Section 5).
23. Sigelman, B.H., Barroso, L.A., Burrows, M., Stephenson, P., Plakal, M., Beaver, D., Jaspan, S., Shanbhag, C. Dapper, a large-scale distributed systems tracing infrastructure. *Google Technical Report* (2010). (Sections 1, 2.3, and 3).
24. Twitter Zipkin. http://twitter.github.io/zipkin/. [Online; accessed March 2015]. (Section 2.3).
25. Yuan, D., Zheng, J., Park, S., Zhou, Y., Savage, S. Improving software diagnosability via log enhancement. *ACM Trans Comput Syst 30*, 1 (2012), 4. (Section 2.3).

Jonathan Mace, Ryan Roelke, and Rodrigo Fonseca, Brown University Department of Computer Science, Providence, RI, USA.

[CONTINUED FROM P. 104] counter-referendums for all laws. In which order should the majority party pass the 10 laws with the goal to maximize the number that become active?

Solution to Warm-Up: The majority party should pass the laws having a 0.999 to survive ones first. In such a case, there is a probability of $0.999^3 = 0.997$ that the minority party will lose the first three counter-referendums and then the majority party will be able to make all 10 laws active without further risk of counter-referendums, even for the laws that have only 0.01 chance of surviving a counter-referendum. Clearly the minority party should pick its battles better.

Warm-Up 2: If the minority party decides not to call for counter-referendums on all laws from the majority party, for which ones should it invoke counter-referendaums assuming the majority party first passes the five laws with a 0.999 of survival and then the five laws with a 0.01 chance of survival?

Solution: One possibility is for the minority to oppose only those laws with a 0.01 chance of surviving a counter-referendum. The minority has a 0.99^3 (approximately 97%) chance of stopping the election after the first three unpopular laws. Another interesting possibility is to contest some of the 0.999 laws, because if the minority wins on those, then there is the possibility of stopping more laws. However, that strategy runs the risk of losing three counter-referendums in a row. In either case, slightly more than five laws will be passed.

OK, I think you are ready now. The goal for the majority is to pass as many laws as possible and the goal for the minority is to prevent the majority from doing so.

Question: Suppose the majority passes laws with survival probabilities 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1 in that order. Which should the minority contest with counter-referendums to minimize the number of laws that become active? Recall the minority must decide which laws to contest as soon as they pass.

Solution (kind of): I do not have a good theory, so I wrote a program that explores the possibilities. Based on my program, the minority should ask for counter-referendums for laws starting with survival probability 0.6 and then the rest that have lower survival probabilities. In such a case, the expected number of laws that will pass is approximately 5.3.

Is there a natural law that says that if a country consists of mostly non-violent people and a small group willing to use force, then the brutes will win? Is there any way for the decent people to stop the brutes?

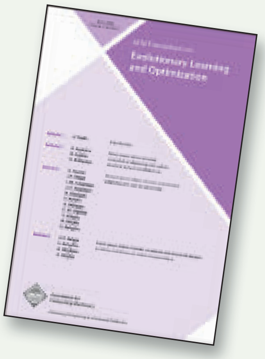
Question: Would the majority party be better off (in terms of total number of laws passed) if it permuted the order in which it passed these laws?

Solution: There may be several better permutations, but at least this one 0.3, 0.5, 0.8, 0.7, 0.6, 0.4, 0.9, 0.2, 0.1 does seem better, raising the expected number of laws to 5.4. In this case, again, the minority should contest every law that has a 0.6 probability of surviving or less.

Upstart: Given a set of potential laws that the majority wants to pass, each with a probability of surviving a counter-referendum, what is the best strategy for the majority side to use in order to pass as many laws as possible, no matter how clever the minority is? The majority can order the laws in any order and may drop some.

All are invited to submit their solutions to upstartpuzzles@cacm.acm.org; solutions to upstarts and discussion will be posted at http://cs.nyu.edu/cs/faculty/shasha/papers/cacmpuzzles.html

Dennis Shasha (dennisshasha@yahoo.com) is a professor of computer science in the Computer Science Department of the Courant Institute at New York University, New York, USA, as well as the chronicler of his good friend the omniheurist Dr. Ecco.



DOI:10.1145/3380444

Dennis Shasha

Upstart Puzzles
Stopping Tyranny

A compromise proposal toward a solution to making it impossible for a would-be tyrant to exceed reasonable authority.

If laws are passed with probabilities of surviving a counter-referendum of 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1 in that order, which should the minority contest?

IMAGE BY ALICIA KUBISTA/ANDRIJ BORYS ASSOCIATES

MOST PEOPLE WHO live in dictatorships are decent, so why are their leaders so bad? Is there a natural law stating if a country consists of mostly non-violent people and a small group willing to use force, then the brutes will win? Is there any way for the decent people to stop the brutes?

One answer is a robust representative democracy. Unfortunately, history is filled with examples (even current ones) in which a leader is democratically elected and then becomes a dictator over time. Representative democracy suffers from the loophole that one bad election can mess up everything.

Is there a way to make the world safe, to make it impossible for a would-be tyrant to exceed reasonable authority when the public starts to realize what has happened? The most straightforward way would be for all decisions to be made by referendum, but that is impractical, because governments make thousands of decisions per day and the public simply does not have the necessary information.

Here is a compromise proposal.

Make it possible for, say, 1/3, of all representatives to call for a "counter-referendum" on any law that has been passed by the legislature. A counter-referendum is a vote by all the people to decide whether to let a law become active or remove it from the books. Making an electronic referendum (or any vote) cryptographically secure is a topic of active research—but suppose that it were secure and enforceable.

To further ensure the majority does not simply re-pass a law that has been rejected by one or more counter-referendums, if three counter-referendums in a row go against one or more laws passed by the legislature, then the majority legislators requires 2/3 super-majorities to pass any further laws for one year.

On the other hand, voter fatigue is always an issue. So if three counter-referendums in a row fail (that is, a majority of counter-referendum voters support the original legislation for three successive counter-referendums), then there will be no possibility for more counter-referendums for a year.

The majority of legislators decides when to vote (and presumably pass) each law. Once a law passes in the legislature, the minority must decide right away whether to put it up for a counter-referendum, which happens one month after the original passage. In that case, the law becomes active only if it survives the counter-referendum.

For the simplifying purposes of this puzzle, suppose both sides know the probability each law has of surviving a counter-referendum and those probabilities are independent. Trust me, the puzzle is difficult enough even then.

Warm-Up: Suppose there are 10 laws the majority party can pass but each of five has a probability of 0.999 to survive a counter-referendum and each of five has only a 0.01 chance of surviving. Suppose further the minority party will invoke

[CONTINUED ON P. 103]

FODS-2020: ACM-IMS Foundations of Data Science Conference

October 18 - 20, 2020 | Seattle, Washington

The Association for Computing Machinery (ACM) and the Institute of Mathematical Statistics (IMS) have come together to launch a conference series on the Foundations of Data Science. Our inaugural event, the ACM-IMS Interdisciplinary Summit on the Foundations of Data Science, took place in San Francisco in 2019. Starting in 2020 we will have an annual conference with refereed conference proceedings. This interdisciplinary event will bring together researchers and practitioners to address foundational data science challenges in prediction, inference, fairness, ethics and the future of data science.

Key Dates

Submission: April 13, 2020
Notification: July 15, 2020
Camera-ready: August 1, 2020

https://fods.acm.org/

FODS-2020 Conference Co-chairs

Jeannette Wing
Columbia University

David Madigan
Columbia University

Contact:
fods2020@columbia.edu

acm ims