

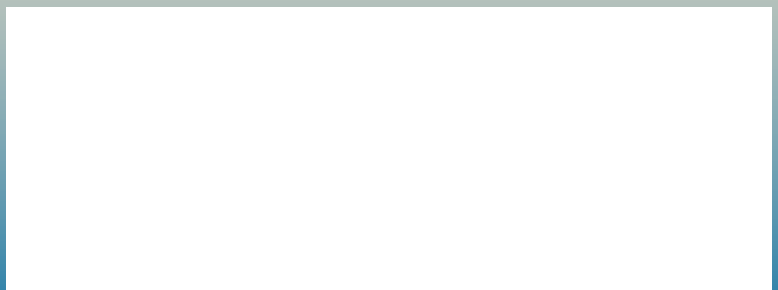
COMPUTING edge

- Software Engineering
- Cybersecurity
- Artificial Intelligence
- Ethics



APRIL 2022

www.computer.org



IEEE TRANSACTIONS ON

COMPUTERS

Call for Papers: IEEE Transactions on Computers

Publish your work in the IEEE Computer Society's flagship journal, *IEEE Transactions on Computers (TC)*. *TC* is a monthly publication with a wide distribution to researchers, industry professionals, and educators in the computing field.

TC seeks original research contributions on areas of current computing interest, including the following topics:

- Computer architecture
- Software systems
- Mobile and embedded systems
- Security and reliability
- Machine learning
- Quantum computing

All accepted manuscripts are automatically considered for the monthly featured paper and annual Best Paper Award.

Learn about calls for papers and submission details at www.computer.org/tc.



IEEE
COMPUTER
SOCIETY



STAFF

Editor

Cathy Martin

Publications Portfolio Managers

Carrie Clark, Kimberly Sperka

Senior Advertising Coordinator

Debbie Sims

Production & Design Artist

Carmen Flores-Garvey

Publisher

Robin Baldwin

Circulation: *ComputingEdge* (ISSN 2469-7087) is published monthly by the IEEE Computer Society, IEEE Headquarters, Three Park Avenue, 17th Floor, New York, NY 10016-5997; IEEE Computer Society Publications Office, 10662 Los Vaqueros Circle, Los Alamitos, CA 90720; voice +1 714 821 8380; fax +1 714 821 4010; IEEE Computer Society Headquarters, 2001 L Street NW, Suite 700, Washington, DC 20036.

Postmaster: Send address changes to *ComputingEdge*-IEEE Membership Processing Dept., 445 Hoes Lane, Piscataway, NJ 08855. Periodicals Postage Paid at New York, New York, and at additional mailing offices. Printed in USA.

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

Reuse Rights and Reprint Permissions: Educational or personal use of this material is permitted without fee, provided such use: 1) is not made for profit; 2) includes this notice and a full citation to the original work on the first page of the copy; and 3) does not imply IEEE endorsement of any third-party products or services. Authors and their companies are permitted to post the accepted version of IEEE-copyrighted material on their own Web servers without permission, provided that the IEEE copyright notice and a full citation to the original work appear on the first screen of the posted copy. An accepted manuscript is a version which has been revised by the author to incorporate review suggestions, but not the published version with copy-editing, proofreading, and formatting added by IEEE. For more information, please go to: http://www.ieee.org/publications_standards/publications

</rights/paperversionpolicy.html>. Permission to reprint/republish this material for commercial, advertising, or promotional purposes or for creating new collective works for resale or redistribution must be obtained from IEEE by writing to the IEEE Intellectual Property Rights Office, 445 Hoes Lane, Piscataway, NJ 08854-4141 or pubs-permissions@ieee.org. Copyright © 2022 IEEE. All rights reserved.

Abstracting and Library Use: Abstracting is permitted with credit to the source. Libraries are permitted to photocopy for private use of patrons, provided the per-copy fee indicated in the code at the bottom of the first page is paid through the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923.

Unsubscribe: If you no longer wish to receive this *ComputingEdge* mailing, please email IEEE Computer Society Customer Service at help@computer.org and type "unsubscribe *ComputingEdge*" in your subject line.

IEEE prohibits discrimination, harassment, and bullying. For more information, visit www.ieee.org/web/aboutus/whatis/policies/p9-26.html.

IEEE Computer Society Magazine Editors in Chief

Computer

Jeff Voas, *NIST*

Computing in Science & Engineering

Lorena A. Barba, *George Washington University*

IEEE Annals of the History of Computing

Gerardo Con Diaz, *University of California, Davis*

IEEE Computer Graphics and Applications

Torsten Möller, *Universität Wien*

IEEE Intelligent Systems

Longbing Cao, *University of Technology Sydney*

IEEE Internet Computing

George Pallis, *University of Cyprus*

IEEE Micro

Lizy Kurian John, *University of Texas at Austin*

IEEE MultiMedia

Shu-Ching Chen, *Florida International University*

IEEE Pervasive Computing

Marc Langheinrich, *Università della Svizzera italiana*

IEEE Security & Privacy

Sean Peisert, *Lawrence Berkeley National Laboratory and University of California, Davis*

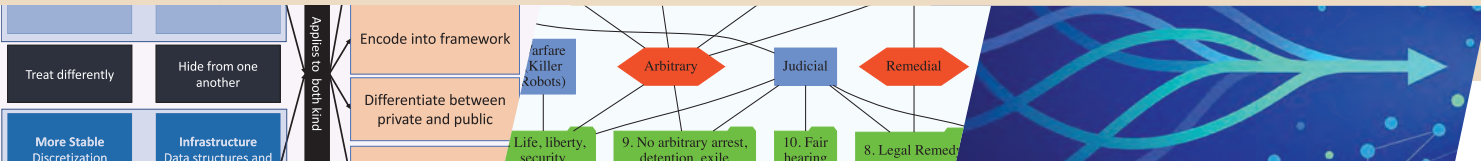
IEEE Software

Ipek Ozkaya, *Software Engineering Institute*

IT Professional

Irena Bojanova, *NIST*

COMPUTING edge



8

Insights From the Software Design of a Multiphysics Multicomponent Scientific Code

36

Artificial Intelligence and the Right to Explanation as a Human Right

48

Building an Accessible Digital World

Software Engineering

- 8 Insights From the Software Design of a Multiphysics Multicomponent Scientific Code

ANSHU DUBEY

- 12 The Behavioral Science of Software Engineering and Human-Machine Teaming

IPEK OZKAYA

Cybersecurity

- 16 The Challenges of Software Cybersecurity Certification

JOSÉ L. HERNÁNDEZ-RAMOS, SARA N. MATHEU, AND ANTONIO SKARMETA

- 21 Security Test

CHRISTOF EBERT, YOUSSEF REKIK, AND RAHUL KARADE

Artificial Intelligence

- 30 Formal Methods Boost Experimental Performance for Explainable AI

FREDERIK GOSSEN, TIZIANA MARGARIA, AND BERNHARD STEFFEN

- 36 Artificial Intelligence and the Right to Explanation as a Human Right

MICHAEL WINIKOFF AND JULIJA SARDELIĆ

Ethics

- 42 Pervasive Healthcare IRBs and Ethics Reviews in Research: Going Beyond the Paperwork

JINA HUH-YOO, REEMA KADRI, AND LORRAINE R. BUIS

- 48 Building an Accessible Digital World

SARAH HORTON

Departments

- 4 Magazine Roundup
7 Editor's Note: Engineering Interdisciplinary Software
55 Conference Calendar

Subscribe to *ComputingEdge* for free at www.computer.org/computingedge.



Magazine Roundup

The IEEE Computer Society's lineup of 12 peer-reviewed technical magazines covers cutting-edge topics ranging from software design and computer graphics to Internet computing and security, from scientific applications and machine intelligence to visualization and microchip design. Here are highlights from recent issues.

Computer

The ConnectionRoom: A New Analogy for Understanding the Ethical Dimensions of Social Media

The authors of this article from the January 2022 issue of *Computer* propose an analogy to help us understand that social media providers are brokers who connect us for a price. They challenge social media stakeholders to strike a deal that is transparent, equitable, and wise.

Computing

On Preserving Scientific Integrity for Climate Model Data in the HPC Era

Over the past 30 years, the Computational Science Graduate Fellowship (CSGF) program has played an integral role in preparing a large and diverse community of computational scientists to push the limits of high-performance computing (HPC). To celebrate the CSGF program's enduring influence, the author of this article from the November/December 2021 issue of *Computing in*

Science & Engineering shares perspective from the climate modeling community, which has used HPC to better understand Earth's climate system. While the benefits of HPC in climate science have been enormous, rapid computing advances have brought new challenges. One difficulty is quality assurance—ensuring that large and complex codes running on multiple platforms are correct. A second is mitigating the increasingly large data volumes.

IEEE Annals

of the History of Computing

Computing the Cubicle: Design for the High-Tech Office, 1970–1990

This article from the July–September 2021 issue of *IEEE Annals of the History of Computing* examines the influence of personal computing on the interior design of American offices and office furniture from 1970 to 1990. The author argues that the cubicle, with its modular system of work surfaces, powered partitions, and ergonomic accessories, served as an intermediary among the computer, the organization, the architecture, and the worker adapting

the computer to the office environment, and adapting the office environment to the computer.

IEEE Computer Graphics

AND APPLICATIONS

Visualization Design Sprints for Online and On- Campus Courses

The authors of this article from the November/December 2021 issue of *IEEE Computer Graphics and Applications* present how to integrate design sprints and project-based learning into introductory visualization courses. A design sprint is a unique process based on rapid prototyping and user testing to define goals and validate ideas before starting costly development. The well-defined, interactive, and time-constrained design cycle makes design sprints a promising option for teaching project-based and active learning-centered courses to increase student engagement and hands-on experience. Over the past five years, the authors have adjusted the design sprint methodology for teaching a range of visualization courses. They present a detailed guide on incorporating design sprints into large undergraduate and small professional



development courses in both online and on-campus settings.

IEEE Intelligent Systems

Combining Sentiment Lexicons and Content-Based Features for Depression Detection

Numerous studies on depression have found that tweets posted by users with major depressive disorder could be utilized for depression detection. The potential of sentiment analysis for detecting depression through an analysis of social media messages has brought increasing attention to this field. In this article from the November/December 2021 issue of *IEEE Intelligent Systems*, the authors propose 90 unique features as input to a machine-learning classifier framework for detecting depression using social media texts. Derived from a combination of feature extraction approaches using sentiment lexicons and textual contents, these features can provide impressive results.

IEEE Internet Computing

Multipath Deadline-Aware Transport Proxy for Space Network

One way to maintain the communication between the space

station and the ground station is to use several relay satellites. However, those relay links suffer from high loss rate, limited bandwidth, and long round-trip time. Meanwhile applications running over these links, such as real-time communication, usually have deadline requirements for their data transfer. QUIC version 1 was released as RFC9000 by Iyengar and Thomson in 2021; it provides a great opportunity to improve transport services. Based on QUIC, the authors of this article from the November/December 2021 issue of *IEEE Internet Computing* develop multipath deadline-aware transport proxy to provide the deadline-aware transmission service for those applications. To avoid the loss caused by congestion control of each application, the proxy aggregates the data transmission of applications and transmits them along multiple paths.

IEEE micro

Evolution of the Graphics Processing Unit (GPU)

Graphics processing units (GPUs) power today's fastest supercomputers, are the dominant platform for deep learning, and provide the intelligence for devices ranging from self-driving cars to robots and smart cameras. They

also generate compelling photo-realistic images at real-time frame rates. GPUs have evolved by adding features to support new use cases. NVIDIA's GeForce 256, the first GPU, was a dedicated processor for real-time graphics, an application that demands large amounts of floating-point arithmetic for vertex and fragment shading computations and high memory bandwidth. As real-time graphics advanced, GPUs became programmable. The combination of programmability and floating-point performance made GPUs attractive for running scientific applications. Scientists found ways to use early programmable GPUs by casting their calculations as vertex and fragment shaders. Read more in this article from the November/December 2021 issue of *IEEE Micro*.

IEEE MultiMedia

On the User-Centric Comparative Remote Evaluation of Interactive Video Search Systems

In the research of video retrieval systems, comparative assessments during dedicated retrieval competitions provide priceless insights into the performance of individual systems. The scope and depth of such evaluations are unfortunately hard to improve,

due to the limitations by the setup costs, logistics, and organization complexity of large events. The authors of this article from the October–December 2021 issue of *IEEE MultiMedia* show that this easily impairs the statistical significance of the collected results, and the reproducibility of the competition outcomes. They present a methodology for remote comparative evaluations of content-based video retrieval systems and demonstrate that such evaluations scale up to sizes that reliably produce statistically robust results.



Assessing the Impact of Commuting on Workplace Performance Using Mobile Sensing

Commuting to and from work presents daily stressors for most workers. It is typically demanding in terms of time and cost, and it can impact people's mental health, job performance, and personal life. The authors of this article from the October–December 2021 issue of *IEEE Pervasive Computing* use mobile phones and wearable sensing to capture location-related context, physiology, and behavioral patterns of N=275 information workers while they commute, mainly by driving, between home and work locations spread across the United States for a one-year period. They assess the impact of commuting on participants' workplace

performance, showing that they can predict self-reported workplace performance metrics based on passively collected mobile-sensing features captured during commute periods.



Toward Cybersecurity Personalization in Smart Homes

Security personalization has become a critical need for smart homes in recent years. The current approaches cannot fully satisfy this requirement of user-centered security. The authors of this article from the January/February 2022 issue of *IEEE Security & Privacy* propose a user-friendly approach for the automatic configuration of home security solutions through policy-based management, minimizing human interventions, and improving security usability.



Scaling Open Source Software Communities: Challenges and Practices of Decentralization

To satisfy the growing needs of modern society, open source software is becoming increasingly large and complex, with a large number of code patches continually flowing in. For smooth scaling up, the authors of this article from the January/February 2022 issue of *IEEE Software* explore the challenges and best practices of decentralization.



Toward Trustworthy Urban IT Systems: The Bright and Dark Sides of Smart City Development

In smart cities built on information and communication technology, citizens and various IT systems interoperate in harmony. Cloud computing and Internet-of-Things technologies are making modern cities smarter. Smart cities can have a positive impact on citizens, but they can also make cities dangerous. Today, with the emerging reality of smart cities, this article from the November/December 2021 issue of *IT Professional* looks at both the bright and dark sides and provides a foundation for supporting work-related tasks of IT professionals, as well as non-IT experts involved in urban design and development. 🌍

Join the IEEE
Computer
Society

computer.org/join





Editor's Note

Engineering Interdisciplinary Software

Software programs can be designed for a specific purpose, or they can have multiple uses in multiple fields. Interdisciplinary software is more likely to succeed if it is customizable, flexible, easy to use, and trustworthy. This *ComputingEdge* issue showcases two articles that describe the development of interdisciplinary software.

"Insights From the Software Design of a Multiphysics Multi-component Scientific Code," from *Computing in Science & Engineering*, analyzes how one program became useful in multiple domains and has adapted to the world of high-performance computing. "The Behavioral Science of Software Engineering and Human-Machine Teaming," from *IEEE Software*, argues for

incorporating the study of people into the process of designing sociotechnical systems.

Good software is secure software. The authors of *IEEE Security & Privacy's* "The Challenges of Software Cybersecurity Certification" aim to increase adoption of the European Union's cybersecurity certification framework for software components. The authors of *IEEE Software's* "Security Test" advocate for risk-oriented security testing for embedded systems to facilitate the transparent hardening of critical systems.

Good software is also explainable, which is especially important in artificially intelligent systems. In *IT Professional's* "Formal Methods Boost Experimental Performance for Explainable AI," the authors illustrate how formal methods

can improve AI explainability. In *IEEE Internet Computing's* "Artificial Intelligence and the Right to Explanation as a Human Right," the authors reflect on which situations call for AI explanation as a human rights consideration.

Finally, this *ComputingEdge* issue features two articles about ethical concerns in computing. *IEEE Pervasive Computing's* "Pervasive Healthcare IRBs and Ethics Reviews in Research: Going Beyond the Paperwork" discusses the processes of understanding and communicating the risks of technology research that involves human subjects. *Computer's* "Building an Accessible Digital World" emphasizes the importance of creating technology that is accessible to people with disabilities. 🌈

DEPARTMENT: SCIENTIFIC PROGRAMMING

Insights From the Software Design of a Multiphysics Multicomponent Scientific Code

Anshu Dubey , Argonne National Laboratory, Lemont, IL, 60439, USA

Using simulations for scientific discovery requires that the software used in the simulations undergoes a rigorous design and development process similar to that of the lab instruments in the experimental sciences. To devise a good design methodology, it is critical to understand the requirements, constraints, and challenges. This article describes insights from the long-term stewardship of a multiphysics multicomponent software, FLASH, that was designed more than 20 years ago for astrophysics, now serves multiple communities, and has been successful in adapting to the changing world of high-performance computing.

Over the last two decades, the advances in computing hardware and computational mathematics have transformed the methods of scientific discovery. Computational advances in engineering have been instrumental in reducing, and sometimes eliminating, the cost of experiments in product design. While the benefits of computing are appreciated, the equivalence between scientific software and laboratory instruments has not been fully realized. Experimental scientists understand that the quality of their science depends upon the quality of their laboratory instruments. A similar understanding about software quality has not emerged in the computational sciences. Therefore, it is still difficult to persuade domain scientists that upfront investment in robust software design is in their best interest, and that it is an investment whose rewards are reaped for years.¹ The return on the investment is tremendously valuable not only in terms of the quality of science they produce, but also in terms of the scientific productivity of their team members.

In the world of scientific discovery, software can be developed for many different purposes. Best practices

for designing software may differ depending upon their use target. Here, I am going to focus on a design methodology that I found to be very useful in developing a multiphysics multicomponent software, FLASH, meant to be used as a community tool in astrophysics,^{2,3} that then went on to become a community tool for several other domains because its design enabled relatively easy customization for these other domains.

DESIGN CHALLENGES

Before going into the details of the design methodology, it would be good to explain what is involved in science through simulations and the challenges involved. The process of science through simulation roughly follows the flow shown in Figure 1. The phenomena of interest are captured in mathematical models, which are then discretized so that numerical methods can be applied to obtain their solution. **Verification** is the process of ensuring that the model is implemented correctly. This is achieved through testing the software for correctness, stability, and convergence. The process of **validation** is meant to ensure that the devised model adequately captures the phenomenon of interest by comparing it with experiments and/or observations. Various feedback loops in the figure represent steps in the development process where it is important to do sanity checks and ensure that the progress is consistent with the objectives of the science being done.

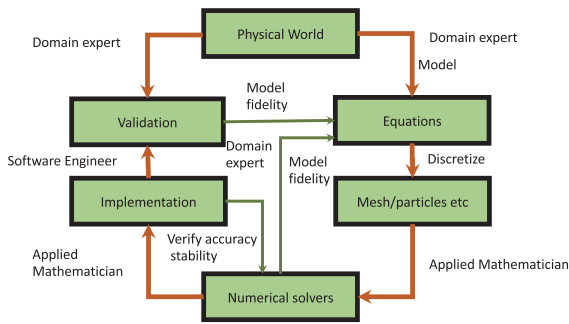


FIGURE 1. Schematic of how simulations are typically used in scientific discovery.

Although Figure 1 captures the stages of development, it says little about the challenges at each stage. The entire design process is a balancing act between competing concerns. Well-known and understood principles for software design can be at cross purposes with the demands that are placed on the control flow by the requirements dictated by nature. For instance, the most basic good software design principles are modularity and encapsulation. However, real world is messy, and the model capturing its behavior may not lend itself to easy modularization. Similarly, one would like to design data layouts that minimize rearrangements in memory and maximize spatial and temporal locality for good performance. However, often different solvers have different optimal data layouts, and one is forced to consider trade-offs between the cost of rearranging data versus the slowdown caused by suboptimal layouts.

METHODOLOGY

The map of expertise needed as shown in Figure 1 brings out another challenge; the development team members are likely to come from diverse fields with different expertise. The design needs to be cognizant of the technical necessities for an interdisciplinary team to be productive. One key principle is enabling people to focus on what they know best without having to learn all aspects of the software; or in other words, **Separation of Concerns**. The term implies that software is developed in a way that different aspects of software do not interfere too much with one another. For example, in writing a parallel code for distributed-memory machines, a good practice was to separate all the local calculations from those that needed communication between processors. If the code is organized this way, for example, the applied mathematicians can do their algorithm development

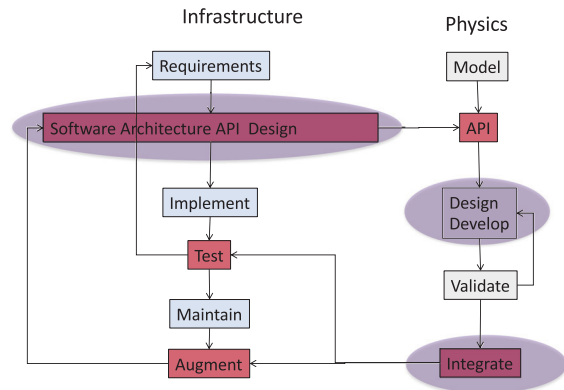


FIGURE 2. Design approach for achieving separation of concerns.

largely independently of the parallelization, while performance engineers can focus on optimizing scaling without having to know all the math.

THE KEY TO ACHIEVING SEPARATION OF CONCERNS RELATES TO THE NEED FOR MODULARIZATION AND ENCAPSULATION.

This principle is well known and has been followed by many projects that have had success in their respective science communities.⁴ The key to achieving separation of concerns relates to the need for modularization and encapsulation. And it brings back the question of lateral interactions between modules dictated by nature that can make encapsulation difficult. One way to achieve a semblance of encapsulation is to modularize on the basis of similar well-defined functionalities and provide explicit interfaces for lateral coupling if needed. Interfaces should be designed to achieve a good balance between adequate functionality and flexibility without unnecessary bloat. Figure 2 shows a workflow for achieving separation of concerns. The figure has two branches of development, one that pertains to the infrastructure and book-keeping part of the code, and the other is the part that implements the arithmetic of the computation. These two branches interact at a few points through interfaces where the first branch provides services needed by the second branch. Sometimes components in the second branch may undergo changes that need to be communicated to the first branch, hence the provision for *augmenting* the first branch.

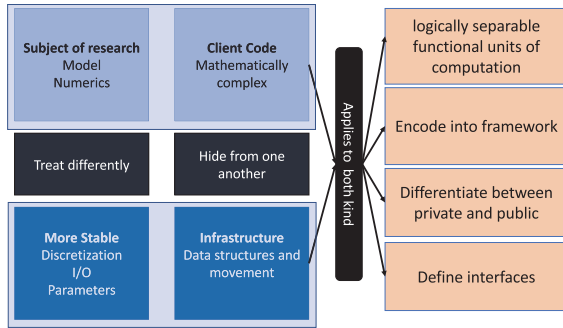


FIGURE 3. Overview of software architecture that accommodates slow-evolving and fast-evolving components.

The other fundamental design methodology that is critical for long-lived nimble scientific code is implicit in the discussion above. It is based on recognizing that the design of the two branches ought to be treated differently. The infrastructure, or the backbone of the code is the key to the robustness, performance characteristics, and extensibility of the code, and therefore its longevity. Being the service provider, this part of the code needs a thorough understanding of the design constraints imposed by the algorithms used in the science models. A careful exploration of the design space with prototyping and evaluation requires a nontrivial amount of upfront investment, but it is this investment that ultimately pays off. It is inevitable that at certain cadence even the infrastructure undergoes deep refactoring because both the hardware and the numerical methods constantly evolve. But, if designed well, it should typically not need major overhaul through several generations of computing platforms.

SINCE SCIENTIFIC SOFTWARE IS DEVELOPED FOR THE PURPOSE OF EXPLORATION, IT IS RARELY USED IN EXACTLY THE SAME WAY MORE THAN ONCE.

The other part of the code, that implements the model, should be treated as the client code with as close to a plug-and-play design as feasible. The evolution of science domains usually goes hand-in-hand with advances in model fidelity and the methods that implement the models. Because science campaigns

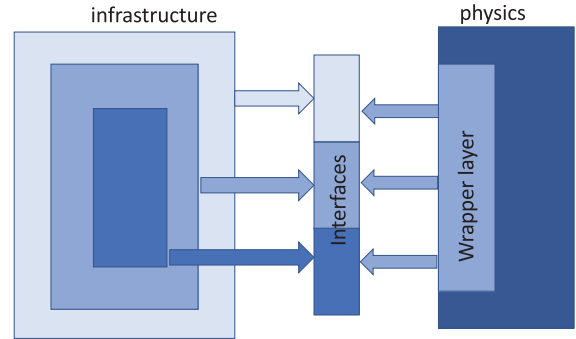


FIGURE 4. Interface design for layered access to framework features.

typically take the code in uncharted territories for exploration, this part of the code is subject to continuous and rapid changes and should be designed such that these advances can be quickly assimilated. Ideally these advances and any customizations needed should either not involve the infrastructure at all or at best require small tweaks to the infrastructure. Figure 3 captures the methodology for devising a software architecture that accommodates robust slow-evolving infrastructure with flexible and nimble science solvers coexisting in the software.

Since scientific software is developed for the purpose of exploration, it is rarely used in exactly the same way more than once. Every new science project using it is likely to tweak, modify, and customize it in some way. Often whole new capabilities may be added for a new project. Therefore, extensibility and customizability are critical, but nontrivial, to achieve because it is difficult to anticipate the direction in which it may be needed in future. An added new challenge is the increasing heterogeneity in the computing platforms. Not even the mid-size clusters these days are without some form of accelerators that provide the bulk of computing power. And these accelerators differ from one vendor to another, and from one generation to the next by the same vendor. This necessitates making the separation of concerns more concrete, and interfaces more flexible. One way to balance these somewhat conflicting requirements is to design for hierarchical access to the infrastructure as show in Figure 4. Here, the details of the numerical method are known only to the fully encapsulated part of the physics in the figure. The wrapper layer chooses the functionality that may be exposed to other physics components, and permit lateral coupling between them as needed. The infrastructure in turn exposes its functionalities at different levels of granularity and hierarchy to the physics components. This kind of

layering allows a range of transparency to the developers of physics modules. A less demanding physics module can opt for more transparency and interact with the infrastructure at a superficial level. However, should the developer of a physics module wish for greater control over the use of resources, and exercise the advanced features of the infrastructure, they can opt to interact with the infrastructure at a deeper level. The tradeoff is between the extent of knowledge and understanding of the infrastructure and possibility of better performance. Such a design has the added advantage that it does not eliminate the possibility of including physics that demands deep interaction with the infrastructure to be computed. A critical design principle, therefore, is that whenever there is choice between transparency to the end-user or flexibility, the software architect should opt for flexibility.

CONCLUSIONS

A very important question posed by the added heterogeneity in platforms is how much should the design process change to cope with it. Based on our experience in developing Flash-X (the new exascale code derived from FLASH), I have found that the basic design principles do not change, but the details get more complex. The boxes circled in Figure 2, combined with the layered interface design shown in Figure 4 was found to be sufficient for our purposes. The infrastructure has become a lot more complex,⁵ but the basic principles of separation of concerns, modularity, flexibility, and well-thought-out interfaces still hold. 🍌

ACKNOWLEDGMENTS

This work was supported by the U.S. Department of Energy, Office of Science, under Contract DE-AC02-06CH11357.

REFERENCES

1. A. Dubey, P. Tzeferacos, and D. Q. Lamb, "The dividends of investing in computational software design: A case study," *Int. J. High Perform. Comput. Appl.* vol. 33, no. 2, pp. 322–331, 2019, doi: 10.1177/1094342017747692.
2. A. Dubey *et al.*, "Extensible component-based architecture for FLASH, a massively parallel, multiphysics simulation code," *Parallel Comput.*, vol. 35, no. 10–11, pp. 512–522, 2009, doi: 10.1016/j.parco.2009.08.001.
3. A. Dubey *et al.*, "Evolution of FLASH, a multi-physics scientific simulation code for high-performance computing," *Int. J. High Perform. Comput. Appl.*, vol. 28, no. 2, pp. 225–237, 2014, doi: 10.1177/1094342013505656.
4. A. Grannan, K. Sood, B. Norris, and A. Dubey, "Understanding the landscape of scientific software used on high-performance computing platforms," *Int. J. High Perform. Comput. Appl.*, vol. 34, no. 4, pp. 465–477, 2020, doi: 10.1177/1094342019899451.
5. A. Dubey, J. O'Neal, K. Weide, and S. Chawdhary, "Distillation of best practices from refactoring flash for exascale," *SN Comput. Sci.*, vol. 1, no. 4, pp. 1–9, 2020, doi: 10.1007/s42979-020-0077-x.

ANSHU DUBEY is currently a computational scientist with the Mathematics and Computer Science Division, Argonne National Laboratory, Lemont, IL, USA. She received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, New Delhi, India, in 1985, and the Ph.D. degree in computer science from Old Dominion University, Norfolk, VA, USA, in 1993. Her research interests include the area of high-performance scientific computing and software sustainability. Contact her at adubey@anl.gov.

DEPARTMENT: FROM THE EDITOR

The Behavioral Science of Software Engineering and Human–Machine Teaming

Ipek Ozkaya

Designing and sustaining sociotechnical systems where relationships among humans, machines, and environmental aspects are intertwined is not new to software engineering. Emery and Trist¹ coined the term *sociotechnical systems* in 1960 to draw attention to the need for people, machines, and context to all be considered when developing and sustaining these systems. Interactions and dependencies in sociotechnical systems get complex quickly as the interdisciplinary nature of such sys-

THE BEHAVIORAL SCIENCE OF SOFTWARE ENGINEERING FOCUSES ON THE COGNITIVE, SOCIAL, AND BEHAVIORAL IMPLICATIONS OF DEVELOPING SOFTWARE SYSTEMS.

tems drive different design priorities and information flow mechanisms: sociologists see social systems, psychologists observe them as cognitive systems, computer scientists approach them as information systems, and engineers see the hardware systems.² All of these perspectives are not only valid but also are essential elements of sociotechnical systems.

The behavioral science of software engineering focuses on the cognitive, social, and behavioral implications of developing software systems.³ In a recent publication, Storey and colleagues⁴ examined 151 software engineering papers published in two premium

software engineering venues, the International Conference on Software Engineering and *Empirical Software Engineering Journal*. They observed that, while the findings cited in the papers claimed to focus on people as part of their research, they often did not include explicit consideration of human aspects. These findings demonstrate that while software engineers recognize that software systems are part of the sociotechnical systems in which humans and their behavior are part of the system design, we still lack a clear emphasis on incorporating the study of humans into the process of design.

The sociotechnical systems of the future without doubt will also include artificial intelligence (AI) components. Smith⁵ emphasizes that designing trustworthy AI systems and human–machine teaming has to start from an explicit and consciously designed inclusion of human aspects. Understanding human reasoning and cognition has always been crucial in software engineering to better design for the complex interactions between users and systems. However, we are entering a new era where the behavioral science of software and system engineering must increasingly both guide design principles of sociotechnical systems and focus explicitly on human–machine teaming. How technologies will interface with humans to establish effective human–machine teaming requires an understanding of how various engineers, developers, and end users behave as well as an understanding of the uncertainty involved in the behavior of AI-enabled systems.

WHAT IS HUMAN–MACHINE TEAMING?

The term *human–machine teaming* refers to the efficient and effective integration of humans with

complex machines. While it is easy to assume that any user interaction with any user system is human-machine teaming, our emphasis should be on teaming rather than just human-machine interactions. Effective teaming implies having a shared awareness of the task, team, and context as well as some shared commonality and understanding of the end goal to be achieved. In a recent report, 605 U.S. workers were asked to identify an intelligent technology they use on a regular basis and classify the interaction with that technology as a teammate or a tool. In this study, 68% of the respondents classified the intelligent technologies they employed, ranging from autonomous cars, service robots, industrial robots, robotic assistants, and navigation aids to small home intelligent devices, as tools rather than teammates. The lack of decision authority and communication richness was among the top reasons why participants viewed the technology as a tool instead of a teammate.⁶

CHANGING INTERACTIONS AND THE MENTAL MODEL OF USERS

How user interaction models will need to evolve when considering human-machine teaming is currently insufficiently studied in behavioral science of software engineering. A top priority concern in designing effective human-machine teaming is trust: whether humans will and should trust the systems to make decisions on their behalf or collaboratively. The interaction models of humans with computers will and should change. Improving our understanding of what effective and trustworthy human-machine teaming looks like will shape the design of interactions. Researchers will also need to better understand how human-machine interactions will deviate from current design models and consequently develop new models.

Software systems influence human cognition and task flows; how those task flows should be modified is not always predetermined or even understood despite all the contextual design focus when constructing systems.⁷ A software system as a tool creates new task flows. The ultimate goal of any software system is to improve the effectiveness of its users in completing their tasks. Successful systems are those that augment human behavior in more

efficient ways or sometimes define a completely new way for people to achieve their tasks. An example of this phenomenon was observed when CAD tools became available to engineers and designers after their first introduction as a concept with Sketchpad in 1963 by Ivan Sutherland.⁸ CAD tools work with the mental model of repetition, reuse, and scaling to the rest of the system that is being designed.

DESIGNING TRUSTWORTHY AI SYSTEMS AND HUMAN-MACHINE TEAMING HAS TO START FROM AN EXPLICIT AND CONSCIOUSLY DESIGNED INCLUSION OF HUMAN ASPECTS.

CAD broke the barrier between the act of designing and that of creating the blueprint artifact. However, the engineers and designers who are the target users for these tools had to learn to approach their task differently. They needed to recognize the reused elements of their designs so that they could create once and propagate as needed. CAD tools influenced design capture and, in a way, eliminated barriers, allowing quicker iterations and approaching design as an activity where repetitive elements are proactively recognized.⁹

CAD tools enabled new interaction flows to be accepted by end users by focusing on their goals: to iterate on designs at ease and create the artifact along the way without an added burden. Consequently, as the users became familiar with these tools, they were able to allocate more time to the design activity. We will likely observe similar task shifts as we gain more experiences in human-machine teaming through the development and use of AI-enabled systems, in particular autonomous systems. For example, how quickly should a human react to a potentially wrong recommendation from the system, which recommendations are more essential to react to, how should users redirect their attention, and how can systems be designed to best support their human counterparts for effective teaming? We are yet to understand the limitations and horizons of humans in this new mode of human-machine teaming.

THE BEHAVIORAL SCIENCE OF SOFTWARE ENGINEERING: IMPLICATIONS FOR HUMAN–MACHINE TEAMING

There are a number of implications for those studying the behavioral science of software engineering as well as those developing systems that will need to incorporate human–machine teaming. Software developers, software engineering researchers, data scientists, and engineers will need to do the following:

DEFINING THE BOUNDARIES OF TRUST WHEN HUMAN–MACHINE TEAMING IS INVOLVED WILL IMPROVE THE CAPABILITIES OF THE SYSTEMS DEVELOPED.

- › Consider human aspects explicitly, with a focus on how their task flows may evolve and whether such changes are acceptable and within the goals of the outcomes expected from the systems.
- › Start with a clear understanding of trust within the context of the system, from the perspective of end users, and design to that level of verifiable trust. Humans have different tolerance levels of trust depending on the system they are using. Defining the boundaries of trust when human–machine teaming is involved will improve the capabilities of the systems developed.
- › Recognize that the systems developed may imply new interaction models where people may need to be retrained or the systems may need to be redesigned to improve the task flows for most effective human–machine interaction.
- › Recognize that human–machine teaming goes beyond human–machine interaction and expectations such as trust, ethics, privacy, and control not only take priority as part of the behavioral science of software engineering but should also drive the system design.

Smith¹⁰ shares an initial human–machine teaming framework checklist and agreement for teams who

are designing for human–machine teaming. These can serve as a good starting point for both software engineers and behavioral scientists. 🤖

REFERENCES

1. F. E. Emery, E. L. Trist, C. W. Churchman, and M. Verhulst, “Socio-technical systems,” in *Management Science: Models and Techniques*, vol. 2, C. W. Churchman and M. Verhulst, Eds. Oxford, UK: Pergamon, 1960, pp. 83–97.
2. G. D. Baxter and I. Sommerville, “Socio-technical systems: From design methods to systems engineering,” *Interact. Comput.*, vol. 23, no. 1, pp. 4–17, 2011. doi: 10.1016/j.intcom.2010.07.003.
3. M. Petre, J. Buckley, L. Church, M.-A. Storey, and T. Zimmermann, “Behavioral science of software engineering,” *Software*, vol. 37, no. 6, pp. 21–25. doi: 10.1109/MS.2020.3014413.
4. M. Storey, N. A. Ernst, C. Williams, and E. Kalliamvaku, “The who, what, how of software engineering research: A socio-technical framework,” *Empir. Softw. Eng.*, vol. 25, no. 5, pp. 4097–4129, 2020. doi: 10.1007/s10664-020-09858-z.
5. C. J. Smith, “*Designing trustworthy AI: A human-machine teaming framework to guide development*,” 2019. [Online]. Available: <http://arXiv:CoRRabs/1910.03515>
6. J. Lyons, K. Wynne, S. Mahoney, and M. Roebke, “Trust and human-machine teaming: A qualitative study,” in *Artificial Intelligence for the Internet of Everything*. Amsterdam, The Netherlands: Elsevier, 2019, pp. 101–116.
7. K. Holtzblatt and H. Beyer, *Contextual Design: Defining Customer-Centered Systems (Interactive Technologies)*. San Mateo, CA: Morgan Kaufmann, 1997.
8. I. Sutherland, “*Sketchpad: A man-machine graphical communication system*,” Univ. Cambridge, Tech. Rep. UCAM-CL-TR-574, Sept. w2003.
9. S. K. Bhavnani and B. E. John, “Exploring the unrealized potential of computer-aided drafting,” in *Proc. SIGCHI Conf. Human Factors Computing. Syst. (CHI’96)*, 1996, pp. 332–339. doi: 10.1145/238386.238538.
10. C. J. Smith, “*Designing ethical AI Experiences: Checklist and agreement*,” Carnegie Mellon University, Software Engineering Institute, Pittsburgh, fact sheet, Dec. 2019. Accessed: Sept. 2020. [Online]. Available: <https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=636620>



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

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

COMPUTER SOCIETY WEBSITE: www.computer.org

OMBUDSMAN: Direct unresolved complaints to ombudsman@computer.org.

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

AVAILABLE INFORMATION: To check membership status, report an address change, or obtain more information on any of the following, email Customer Service at help@computer.org or call +1 714 821 8380 (international) or our toll-free number, +1 800 272 6657 (US):

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

PUBLICATIONS AND ACTIVITIES

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

Periodicals: The society publishes 12 magazines and 17 journals. Refer to membership application or request information as noted above.

Conference Proceedings & Books: Conference Publishing Services publishes more than 275 titles every year.

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

Technical Committees: TCs provide professional interaction in more than 30 technical areas and directly influence computer engineering conferences and publications.

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

Certifications: The society offers three software developer credentials. For more information, visit www.computer.org/certification.

BOARD OF GOVERNORS MEETING

TBD

EXECUTIVE COMMITTEE

President: William D. Gropp

President-Elect: Nita Patel

Past President: Forrest Shull

First VP: Riccardo Mariani; **Second VP:** David S. Ebert

Secretary: Jyotika Athavale; **Treasurer:** Michela Taufer

VP, Membership & Geographic Activities: Andre Oboler

VP, Professional & Educational Activities: Hironori Washizaki

VP, Publications: David S. Ebert

VP, Standards Activities: Annette Reilly

VP, Technical & Conference Activities: Grace Lewis

2021–2022 IEEE Division VIII Director: Christina M. Schober

2022–2023 IEEE Division V Director: Cecilia Metra

2022 IEEE Division VIII Director-Elect: Leila De Florian

BOARD OF GOVERNORS

Term Expiring 2022: Nils Aschenbruck,

Ernesto Cuadros-Vargas, David S. Ebert, Grace Lewis,

Hironori Washizaki, Stefano Zanero

Term Expiring 2023: Jyotika Athavale, Terry Benzel,

Takako Hashimoto, Irene Pazos Viana, Annette Reilly,

Deborah Silver

Term Expiring 2024: Saurabh Bagchi, Charles (Chuck) Hansen,

Carlos E. Jimenez-Gomez, Daniel S. Katz, Shixia Liu,

Cyril Onwubiko

EXECUTIVE STAFF

Executive Director: Melissa A. Russell

Director, Governance & Associate Executive Director:

Anne Marie Kelly

Director, Conference Operations: Silvia Ceballos

Director, Information Technology & Services: Sumit Kacker

Director, Marketing & Sales: Michelle Tubb

Director, Membership & Education: Eric Berkowitz

Director, Periodicals & Special Projects: Robin Baldwin

COMPUTER SOCIETY OFFICES

Washington, D.C.: 2001 L St., Ste. 700, Washington, D.C.

20036-4928; **Phone:** +1 202 371 0101; **Fax:** +1 202 728 9614; **Email:**

help@computer.org

Los Alamitos: 10662 Los Vaqueros Cir., Los Alamitos, CA 90720;

Phone: +1 714 821 8380; **Email:** help@computer.org

MEMBERSHIP & PUBLICATION ORDERS

Phone: +1 800 678 4333; Fax: +1 714 821 4641;

Email: help@computer.org

IEEE BOARD OF DIRECTORS

K. J. Ray Liu, *President & CEO*

Saifur Rahman, *President-Elect*

John W. Walz, *Director & Secretary*

Mary Ellen Randall, *Director & Treasurer*

Susan "Kathy" Land, *Past President*

Stephen M. Phillips, *Director & Vice President, Educational Activities*

Lawrence O. Hall, *Director & Vice President, Publication Services and Products*

David A. Koehler, *Director & Vice President, Member and Geographic Activities*

James E. Matthews, *Director & President, Standards Association*

Bruno Meyer, *Director & Vice President, Technical Activities*

Deborah M. Cooper, *Director & President, IEEE-US*

DEPARTMENT: BUILDING SECURITY IN

The Challenges of Software Cybersecurity Certification

José L. Hernández-Ramos, *European Commission, Joint Research Centre*

Sara N. Matheu and Antonio Skarmeta, *University of Murcia*

In 2019, the new European Union (EU) cybersecurity regulation “Cybersecurity Act” (“CSA”)¹ entered into force to create a common framework for the certification of any information and communication technology (ICT) system, including products, services, and processes. The main purpose of this framework is to reduce the current fragmentation of cybersecurity certification schemes² as well as to increase end users’ trust in a hyperconnected society³ by fostering a mutual recognition of certified ICT components in any EU country.*

Despite the expected benefits of cybersecurity certification in terms of transparency for end users and the use of best practices, software providers still consider cybersecurity certification to be a costly and complex process. Indeed, certification could cause delays in the launch of new systems, with a significant economic impact.⁴ So, from the industry’s perspective, why should companies invest time and money in certifying ICT components and systems? This is not an easy question to answer, as security and privacy are not yet highly demanded features, due to a lack of awareness.⁵ The consequence is a vicious circle in which the lack of demand (or awareness) and the required effort cause software providers to oppose applying certification processes that, in turn, would increase user awareness.

In this context, we believe that the realization of the cybersecurity certification framework promoted by the CSA is key to fostering transparency

and trust and, consequently, awareness of ICT systems’ cybersecurity. However, it requires the joint effort of certification bodies, manufacturers, and software providers so that an ICT system is certified according to the cybersecurity of its software components. This aspect could be addressed through the inclusion of cybersecurity requirements in the development, maintenance, and operation of software components in certification schemes, as mentioned by a recent report from the EU Agency for Cybersecurity (ENISA).⁶ However, other challenges also need to be addressed. Thus, our main goal is to increase the awareness of the challenges of cybersecurity certification so that the accreditation’s benefits can be leveraged by end users through a more trustworthy digital ecosystem. Based on recent reports provided by ENISA^{4,6} and according to our own experience in this area,⁷ some of these aspects include the following:

- › Definition and certification under different assurance levels; those levels are defined by the CSA and need to be considered by certification bodies and manufacturers when certifying their systems.
- › Software composability and software updates, which impact the certification of a whole system and its components during their lifecycle; these aspects are of interest to manufacturers and software providers as well as cybersecurity certification practitioners for defining the

* This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see <https://creativecommons.org/licenses/by/4.0/deed.ast>

relationship between different certification schemes.

- › Development of coordinated vulnerability disclosure (CVD) procedures, which must be followed by vulnerability providers (e.g., a certain company or cybersecurity researcher) to maintain software providers' control of their systems.

CYBERSECURITY CERTIFICATION ASSURANCE LEVELS

The first problem concerns what must be certified and how deeply. The evaluation of a software component should consider the system where the component will be deployed as well as the different levels of assurance for the certification process. These assurance levels are defined by the CSA regulation to indicate the rigor and depth of the certification process (self-assessment, basic, substantial, and high) to harmonize the different levels provided by existing certification schemes. For example, the well-known Common Criteria Scheme⁸ already defines its own evaluation assurance levels, with the same purpose. This way, a software component could be evaluated according to a certain assurance level, considering the context and the domain where it will be used. However, this could be unknown when a component is created, or the same component might be deployed in systems that are certified under different assurance levels and contexts.

Should the same software component be certified several times according to different assurance levels and contexts where it might be deployed?

This could make software providers more reluctant to use certification processes if lightweight and efficient approaches are not in place. Furthermore, the fulfillment of a certain assurance level should be measured according to agreed cybersecurity standards. However, there is a lack of standardized and widely used approaches to carry out these processes.⁹ This could reduce users' trust in the cybersecurity certification of software components. Indeed, end users could find it difficult to compare the cybersecurity level of various ICT systems that were certified with various schemes or based on different standards.

Consequently, the use of a harmonized set of standards for different assurance levels is a key factor for the cybersecurity certification process.

SOFTWARE COMPOSABILITY

A single ICT system could be made up of components and subsystems that have additional software modules. Therefore, the system's cybersecurity certification depends on the accreditation of each of its subsystems and software components. However, each of these components may have been certified by using different schemes and assurance levels. Therefore, the question becomes,

How should the different certifications of each component be assembled to compose a system's cybersecurity certification?

Furthermore, the development of a software component may not be linked to a specific product or system. Thus, the certification of a module in certain hardware and in a particular operating system may not be valid for the composition of a specific system. This aspect could hinder the potential reuse of previous certifications for the accreditation of a whole system. In this case, it is important to identify which information from the certification process could help to avoid (at least partially) the recertification of a component. If proper actions are not in place, a new certification process could be required, with additional effort and cost.

The relationship between the security level provided by each software component will also depend on how these modules are interconnected. Indeed, a certain vulnerability in a software library could be more or less exploited depending on the use of the library by the system. Additionally, in an increasingly interconnected world, the security of a certain software component could be influenced by the security level of a system with which the component is communicating. In fact, a system's security level may be reduced if it needs to communicate with a vulnerable system for its intended operation. Therefore, software composability aspects go beyond the usual intrasystem vision.

To address such issues, a key factor is to identify the relationship between software components and

certification schemes. For this purpose, there is an additional need to establish a common set of requirements and guidelines that foster an effective and efficient composability process, taking into account the context of use and CSA assurance levels. These aspects are crucial to deal with the cybersecurity certification of emerging scenarios, such as the ongoing development of contact tracing frameworks and mobile apps to restrain the spread of COVID-19. Indeed, such systems will be composed by several components, including mobile apps and back-end servers, which could be certified according to different schemes and varying requirements, depending on the country.

SOFTWARE UPDATES

According to the CSA, cybersecurity certification schemes must provide support throughout the lifecycle of an ICT system. This means that the cybersecurity level of a certain system could change during its lifecycle, and, consequently, the system could need to be recertified. In particular, during an ICT system's lifecycle, its software components will be updated to extend functionality or cope with a security issue. These updates could modify the interactions and communication with other components within the system and even with other systems. Beyond updating a component itself, a software module's operating environment can also be revised. Furthermore, the certification of systems' components could expire throughout their lifecycle.

How could software updates affect the cybersecurity certification of software components and the whole system?

Depending on the type of software update, the cybersecurity recertification of a component could be required, which, in turn, might necessitate the recertification of the system where the component is deployed. During the process, the software component (and even the whole system) may not be operational, and it may become vulnerable to attacks and threats. Therefore, the system should be put in a secure state based on stable software versions. This aspect could require a system to manage and track the different software versions associated with software

components and their relationships. Furthermore, due to the potential cost of the recertification process, manufacturers and software providers may be reluctant to produce regular updates for their systems, or they might update the systems without using a recertification process. To address this aspect, the use of lightweight, efficient, and automated testing techniques is paramount for the recertification process so that software providers can be encouraged to recertify their updated systems.

CVD

The current trend toward the interconnection of physical devices implies an increase of the attack surface that can be ubiquitously exploited. While mitigating such attacks and vulnerabilities requires suitable security mechanisms and protocols, efficient vulnerability disclosure and sharing is a key factor for cybersecurity certification. In fact, the CSA explicitly mentions the use of repositories that list vulnerabilities as a source of supplementary cybersecurity information for certified ICT systems. The main reason is that a repository of vulnerabilities could foster increasing trust in ICT systems and a growing awareness of cybersecurity risks, and it could help with the tracking of an ICT system's cybersecurity level throughout the system's lifecycle.

However, as described in a recent report by the Center for European Policy Studies,¹⁰ the realization of a CVD framework requires the cooperation and collaboration of different stakeholders at the EU level, including manufacturers and vulnerability finders. The CVD process embraces the discovery, reporting, publication, and remediation of vulnerabilities to minimize the associated risks as well as to increase transparency for end users. Therefore, CVD can help to bridge cybersecurity certification and the software industry. But

will software providers be willing to share information about their components' vulnerabilities?

To cope with this aspect, the vulnerability disclosure process should be also responsible in such a way that manufacturers and software providers are given a certain period of time to prepare patches and notify users in a timely and reliable manner before a

vulnerability is disclosed. Toward this end, we believe that the development of an EU platform for the vulnerability disclosure process must be fostered. As suggested by a recent ENISA report,⁶ this platform could be used to share additional cybersecurity information from an ICT system, including threat models, testing processes, software versions, and information about certification schemes. For the realization of such a program, the use of emerging technologies, such as blockchain, could be considered for building a transparent EU platform on which manufacturers, software providers, and end users share cybersecurity information about ICT systems.¹¹ This platform would serve to foster the alignment of software development activities with the cybersecurity certification process.

QUO VADIS?

Continuous technological advances will enable the development of new ICT systems, shaping innovative digital ecosystems for the benefit of society. As recognized by the CSA, this requires that certification schemes provide a high level of flexibility to adapt to a changing technological environment to avoid the risk of becoming outdated. Furthermore, the CSA regulation contemplates the publication of the Union Rolling Work Program (Article 47) that will be periodically updated to identify strategic priorities for future certification schemes based on criteria such as market demand.

One of the main current advances is the development of 5G technologies and systems that are intended to transform the next digital age. These systems will be enriched by software components whose cybersecurity will affect the deployment of 5G technology. So,

how can cybersecurity certification help to promote the deployment of 5G?

As described in the “Cybersecurity of 5G networks” recommendation,¹² the realization of a cybersecurity certification framework should promote consistent security levels and the creation of certification schemes adapted to 5G-related equipment and software. The use of cybersecurity certification schemes would foster a common understanding of the threats, assets, attacks, and risks of 5G systems, and it would

help to recognize the cybersecurity level of a certain 5G system across all EU member states.

In addition to 5G systems, the development of artificial intelligence systems and quantum computing techniques could be considered for cybersecurity certification in the next future. To be successful, cybersecurity certification must go hand in hand with the software development process to promote more secure ICT systems. 🤖

ACKNOWLEDGMENT

This work has been partially funded by the European Commission, through the H2020-830929 CyberSec4-Europe and H2020-952702 BIECO projects.

REFERENCES

1. European Parliament, “*Regulation (EU) 2019/881 of the European Parliament and of the Council of 17 April 2019 on ENISA (the European Union Agency for Cybersecurity) and on Information and Communications Technology Cybersecurity Certification (Cybersecurity Act)*,” 2019. Accessed: Oct. 23, 2020. [Online]. Available: <https://eur-lex.europa.eu/eli/reg/2019/881/oj>
2. “*State of the art syllabus: Overview of existing cybersecurity standards and certification schemes v2*,” European Cyber Security Organisation, Brussels, Belgium, 2017. [Online]. Available: <https://ecs-org.eu/documents/publications/5a31129ea8e97.pdf>
3. J. L. Hernandez-Ramos, D. Geneiatakis, I. Kounelis, G. Steri, and I. Nai Fovino, “Toward a data-driven society: A technological perspective on the development of cybersecurity and data-protection policies,” *IEEE Security Privacy*, vol. 18, no. 1, pp. 28–38, Jan. 2020. doi: 10.1109/MSEC.2019.2939728.
4. “*Considerations on ICT security certification in EU - Survey report*,” European Network and Information Security Agency, Athens, Greece, 2017. [Online]. Available: https://www.enisa.europa.eu/publications/certification_survey
5. K. Busse, J. Schäfer, and M. Smith, “Replication: No one can hack my mind revisiting a study on expert and non-expert security practices and advice,” in *Proc. 15th Symp. Usable Privacy Security (SOUPS)*, 2019, pp. 117–136.
6. “*Advancing software security in the EU. The role of the EU cybersecurity certification framework*,” European

- Network and Information Security Agency, Athens, Greece, 2019. [Online]. Available: https://www.enisa.europa.eu/publications/advancing-software-security-through-the-eu-certification-framework/at_download/fullReport
7. S. N. Matheu, J. L. Hernandez-Ramos, and A. F. Skarmeta, "Toward a cybersecurity certification framework for the Internet of Things," *IEEE Security Privacy*, vol. 17, no. 3, pp. 66–76, May 2019. doi: 10.1109/MSEC.2019.2904475.
 8. D. S. Herrmann, *Using the Common Criteria for IT Security Evaluation*. Boca Raton, FL: CRC Press, 2002.
 9. "Support of the cybersecurity certification - Recommendations for European standardisation in relation to the Cybersecurity Act," European Network and Information Security Agency, Athens, Greece, 2019. [Online]. Available: https://www.enisa.europa.eu/publications/recommendations-for-european-standardisation-in-relation-to-csa-i/at_download/fullReport
 10. L. Pupillo, A. Ferreira, and G. Varisco, *Software Vulnerability Disclosure in Europe: Technology, Policies and Legal Challenges: Report of a CEPS Task Force*, CEPS Task Force Reports, Brussels, Belgium: Centre for European Policy Studies, June 28, 2018. [Online]. Available: https://www.ceps.eu/download/publication/?id=10636&pdf=CEPS%20TFRonSVD%20with%20cover_0.pdf
 11. R. Neisse et al., "An interledger blockchain platform for cross-border management of cybersecurity information," *IEEE Internet Comput.*, vol. 24, no. 3, pp. 19–29, June 2020. doi: 10.1109/MIC.2020.3002423.
 12. European Commission, "Commission recommendation of 26.3.2019: Cybersecurity of 5G networks," 2019. Accessed: Oct. 23, 2020. [Online]. Available: <https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32019H0534&from=GA>

JOSÉ L. HERNÁNDEZ-RAMOS is a scientific project officer at the European Commission, Joint Research Centre, Ispra, Varese, 21027, Italy. His research interests include the application of security and privacy mechanisms to the Internet of Things and transport systems. Hernández-Ramos received a Ph.D. in computer science from the University of Murcia, Spain. He has served as a technical program committee member and chair for different international

conferences. Contact him at jose-luis.hernandez-ramos@ec.europa.eu.

SARA N. MATHEU is a postdoctoral researcher at the University of Murcia, Murcia, 30100, Spain. Her research interests are related to security certification for the Internet of Things. Matheu received a Ph.D. in computer science from the University of Murcia in 2020. She has participated in several projects, including ARMOUR, CyberSec4Europe, and BIECO. Contact her at saranieves.matheu@um.es.

ANTONIO SKARMETA is a full professor in the Department of Information and Communications Engineering, University of Murcia, Murcia, 30100, Spain. His research interests include the integration of security services, identity, the Internet of Things, and smart cities. Skarmeta received a Ph.D. in computer science from the University of Murcia. He has published more than 200 international papers and been a member of several program committees. Contact him at skarmeta@um.es.



Security Test

Christof Ebert, Youssef Rekik, and Rahul Karade

FROM THE EDITOR

The Internet of Things connects devices with each other and cloud services to create new user experiences. Connectivity, however, invites cyberattacks, which are growing with the use of more standardized equipment, such as Ethernet and Linux software stacks. Risk-oriented security testing through a mix of methods and tools facilitates the transparent hardening of critical systems. Youssef Rekik, Rahul Karade, and I provide an overview of industry-scale technologies for security testing. I look forward to hearing from readers and prospective authors about this column and the technologies you want to know more about. —*Christof Ebert*

Cybercriminals can break into any connected system. Traditionally, IT systems with their many open interfaces had been in the focus of attackers, while embedded systems were perceived to be too difficult to hack and not worth the time and energy required. But as systems have added Ethernet, WLAN, USB, Bluetooth, GPS, and other connectivity features, the number of attack surfaces has increased. The most popular hacking method involves attacking a diagnosis port, or otherwise open interface, which can give a malevolent party access to functions or, at least, the ability to corrupt data and prohibit performance such as denial-of-service attacks.

The pressure to deliver products as fast as possible, combined with increasingly open architectures and overwhelming complexity, has further weakened the quality and security of Internet of Things (IoT) systems across industries. Today, medical devices, such as insulin pumps and pacemakers, are at risk, as are cars, industry production facilities, and wide, distributed utility systems. The more our society depends on connectivity the more we are at risk of being hit by major attacks that have the potential not only to

damage single systems but entire cities and countries. Imagine a major breakdown of electric utilities. Such an event would immediately disable the water supply and, thus, threaten life in the impacted area.

*RISK-ORIENTED SECURITY WITH
DEDICATED TEST METHODS AND
APPROPRIATE TOOLS IS THE CALL OF
THE DAY.*

Risk-oriented security with dedicated test methods and appropriate tools is the call of the day. Security testing must start with static code analysis, proceed with unit tests, and further advance through dedicated methods, such as fuzzing and robustness evaluations up to the level of penetration testing (Pen-Test). Let us briefly introduce risk-oriented security

Digital Object Identifier 10.1109/MS.2019.2958354
Date of current version: 12 February 2020

engineering and delve into the appropriate test methods and tools. We will focus on novel gray-box penetration techniques that bridge threat analysis and more efficient and effective testing.

As with all verification and validation methods, cybersecurity testing requires deep experience and competence to select the best methods and test the end criteria as well as a lean yet effective regression strategy capable of continuous integration and deliveries. Often, we see companies that test components and their interfaces while overlooking security threats in networking and services. We have enriched this article with experiences from our security-consulting projects.

CONVERGENCE AND CYBERSECURITY

The convergence of IT and embedded systems has opened many doors for criminals, literally speaking.¹⁻³ Hacking tools are easily available, even in online shops. They are sold on the normal web. Software-defined radio technologies for man-in-the-middle attacks that mimic and intercept signals are available, as are code grabbers. Forums provide complete tutorials on breaking into utilities and stealing vehicles. For instance, automotive hacking tools and online support are available on websites including Omerta.cc, Nulled.to, ffffff.ru, and Chipadla.ru.

Converging IT and embedded systems and devices, such as cars, transport vehicles, medical equipment, industry automation, and utilities networks, might not easily be broken while running in a stable, disconnected mode. Yet the connectivity infrastructure can and will be used to attack any connected device, such as an automotive electronic control unit (ECU). By breaching the IT servers used for software updates, remote control, and maintenance, hackers can load malware and corrupt data. Attackers have even manipulated cellular networks through built-in subscriber-identification-module cards, which many IT companies use to connect with real-time information and update firmware.

The major problem with these attacks concerns not only data security and privacy but, in most cases, functional safety. Embedded devices that have control systems, such as cars and medical implants, are by definition *safety critical*. The basic

principle, therefore, is simple to understand yet difficult to achieve: There is no functional safety without security. Based on the specific challenges of automotive security, original equipment manufacturers and suppliers must realize effective protection against manipulations of the converging embedded and IT systems.

Let us look to automotive systems since they exhibit the meeting of IT and embedded systems most prominently. Cars demand functional safety. The engine, steering, and braking are influenced by numerous embedded computers. Assistance systems enhance drivers' capabilities through features such as advanced cruise control, platooning, and automatic parking. In the future, we will face fully autonomous cars that will depend even more on IT systems inside and outside. Intercepting their communications or, maybe accidentally, corrupting their data would mean that their initial safety case was no longer valid, which is reason enough to explore technologies for security testing.

RISK-ORIENTED CYBERSECURITY

Risk-oriented security helps to balance the growing threats against increasing complexity during the entire lifecycle. Unlike many previous attempts, our research and many practice projects indicate that while security by design is good, it is not good enough. Effective security must handle the entire lifecycle. To cover the major safety hazards that result through security misuse and abuse scenarios for automobiles, we combine safety and security engineering (Figure 1). While safety and security need their own respective methodologies, we suggest that organizational infrastructures and governance schemes at present should be combined for efficiency and effectiveness. We see this as an evolution toward a full (independent) cybersecurity organization in the lifecycle.^{4,5}

Cybersecurity testing in this space connects security requirements directly to design decisions, following the triple peak model of combined requirements analysis, solution modeling, and test-oriented requirements engineering.⁶ This ensures full traceability from the initial threat analysis and risk assessment (TARA) and definition of security requirements. From a compliance and governance perspective, we see this approach as helpful since it illustrates the

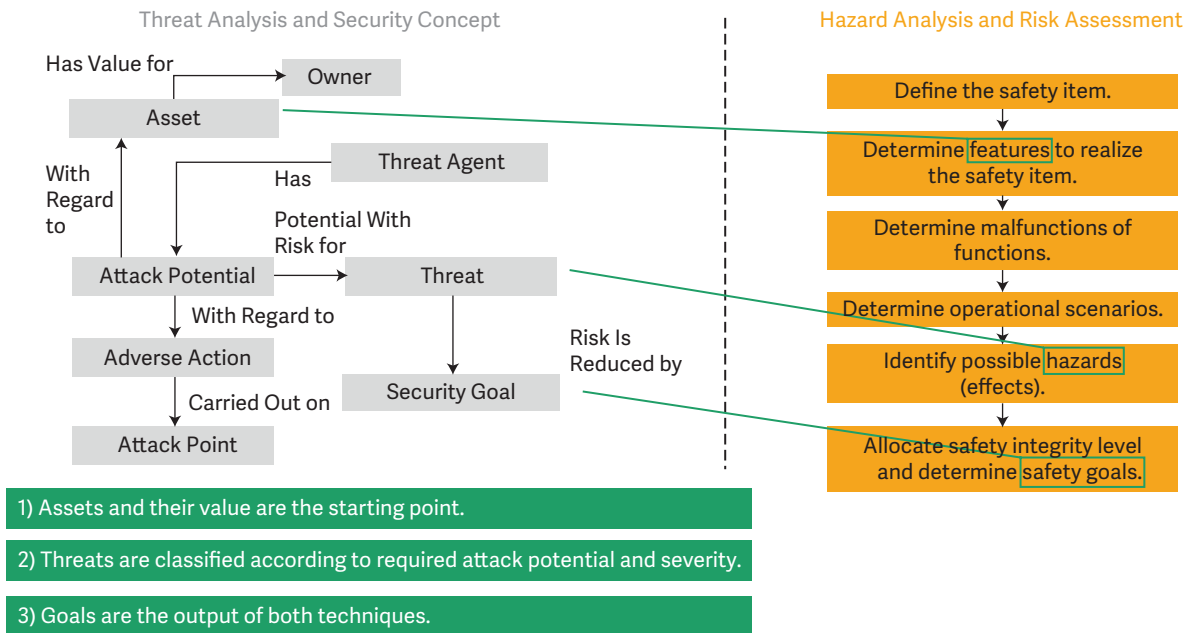


FIGURE 1. Efficiently connecting safety and security.

necessity to prove that security requirements and decisions have been adequately verified through each regression. For instance, at Vector Consulting, we introduced this risk-oriented cybersecurity methodology to a leading supplier in a highly safety-critical environment and ensured that the road-test cycle time could be dramatically reduced.

Security testing can never be complete. The different test strategies, from white-box static analysis to unit tests, fuzzing, and the PenTest, must balance the cost of not having enough security and being attacked, with all of the damaging consequences, versus the expense to implement appropriate security mechanisms and keep them updated through the lifecycle.

SECURITY-TEST TECHNOLOGIES

Security testing is divided into phases, each requiring different tools. Therefore, it is essential to know the tools' strengths and limitations to choose the best one per our requirements. Table 1 provides an overview of test tools that are in wide use. As is the tradition of this column, we examine established tools and newer ones that are gaining popularity. Some tools that are used in more than one industry are chosen to understand their collective impact. There are open-source and commercial tools, some that are

fully automated, and some that are offered in the form of software as a service. The tools were rated "essential" because they were effective and well supported. Specifically, we examined quality attributes such as usability, scalability, and update availability.

Usability

A tool should provide an interactive GUI and command line to better facilitate understanding and training. It should support the preparation of detailed reports and graphs to show the risks and exploits associated with each weakness found during the PenTest. Automating some common procedures should be possible. Frameworks such as the popular Metasploit are introducing all of the methodologies required to carry out each phase of the PenTest process, helping to make obsolete the tools dedicated to a single phase. The Nmap tool, which is exclusively used for reconnaissance scanning, is improving in its methodology to provide information about vulnerabilities that is as detailed as possible.

Scalability

It is essential for the penetration testing tools to support a variety of programming languages and network and application protocols. These tools must provide

TABLE 1. The security-testing tools.

Tool name	Application domain	Methodology	URL	Supplier	Portability			Usage scheme
					Windows	Linux	Mac	
Metasploit	IT, IoT, medical	Scanning and exploitation	https://www.metasploit.com/	Rapid7	✓	✓	✓	<ul style="list-style-type: none"> • Test and exploit vulnerabilities in operating systems and applications • Develop and execute exploit code against a remote target
CANoe	Automotive	Testing and analysis of controllers and networks, including dedicated security testing, such as fuzzing	https://www.vector.com/int/en/products/products-a-z/software/canoe/	Vector	✓			<ul style="list-style-type: none"> • ECU and network of ECUs development, simulation, and testing
BurpSuit Professional	IT	Web security and network scanning	https://portswigger.net/burp/pro	Portswigger	✓	✓		<ul style="list-style-type: none"> • Web security and network scanner
Scapy	IT, automotive	Scanning and exploitation	https://scapy.net/	Scapy	✓	✓	✓	<ul style="list-style-type: none"> • Scanning and exploitation of networked devices and automotive ECUs
BreakingPoint	IT, automotive, energy	Applications and network-security testing	https://www.ixiacom.com/products/network-security-testing-breakingpoint	Ixia				<ul style="list-style-type: none"> • Scanning and exploitation to validate network performance and security posture
AppScan	IT	Dynamic and static security testing	https://www.hcltechsw.com/wps/portal/products/appscan/home	HCL	✓			<ul style="list-style-type: none"> • Exploitation of web and mobile applications
Netsparker	IT	Dynamic analysis	https://www.netsparker.com/	Netsparker	✓			<ul style="list-style-type: none"> • Scanning of web interfaces
bESTORM	Aerospace, automotive, IT	Black-box and dynamic-security testing and validation.	https://beyondsecurity.com/solutions/bestorm.html	Beyond Security	✓	✓		<ul style="list-style-type: none"> • Fuzzing and validating network and embedded devices • New protocol definition using XML • Test applications and hardware, multiprotocol fuzzer
Defensics	IT, automotive	Comprehensive fuzzing and vulnerabilities-scanning framework	https://www.synopsys.com/software-integrity/security-testing/fuzz-testing.html	Synopsys	✓	✓		<ul style="list-style-type: none"> • Black-box fuzzing to test APIs and services, discovering and remediating unknown vulnerabilities in software and devices • Software, telecommunication, CAN, and IP protocols fuzzing

API: application programming interface; CAN: controller-area network.

APIs to extend penetration testing to multiple hardware and software targets. The portability of the software helps in reaching a greater number of researchers. The necessary argument one can make is that the system provided by a company should be compatible with other systems. This enhances the scope of the testing tools.

Availability

A tool’s license and cost are important aspects. Tools that continuously update their vulnerability and exploit databases are recommended. Those that contain an extensive threat library and multivector testing capabilities are better choices for security testing.

	Strengths	Limitations	License	Cost	Notable qualities
	<ul style="list-style-type: none"> Up-to-date database of known vulnerabilities and exploits Prewritten scripts Hardware bridge API: an IoT PenTest extension 	<ul style="list-style-type: none"> Limited capabilities for the free version Requires personnel training 	Open source, commercial	Low	Contemporary interface is intuitive and provides fast learning curve
	<ul style="list-style-type: none"> Its open design helps to easily connect to other tools. It can be used to sniff and analyze all automotive networks. It includes XCP (the standard protocol for ECU development). 	<ul style="list-style-type: none"> Currently limited to automotive security testing Complex test language 	Commercial	High	Comprehensive GUI improves the usability. Uses dedicated security test methods.
	<ul style="list-style-type: none"> Automated as well as manual testing The spidering feature scans the website end to end with automated and manual options. 	<ul style="list-style-type: none"> The tester needs to be trained. It is difficult to understand the tool on its own. 	Commercial	Medium	Ease of use and effective vulnerability scanning
	<ul style="list-style-type: none"> Supports multiple network protocols Interactive packet and result manipulation Fast packet designing 	<ul style="list-style-type: none"> Can't handle a large number of packets simultaneously Partial support for certain complex protocols 	Open source	Low	Modular and extensible to other protocols
	<ul style="list-style-type: none"> Large number of supported application protocols Continuous update of exploits, malware, botnet, and distributed denial-of-service attacks 	<ul style="list-style-type: none"> Using the recommended Ixia hardware makes it expensive. It needs to be installed on a hypervisor. 	Commercial	High	Efficient threat detection
	<ul style="list-style-type: none"> It highlights, with several grades of severity, the types of vulnerabilities. 	<ul style="list-style-type: none"> Scans may become slow on large websites. 	Commercial	Medium	Reliable with low number of false positives
	<ul style="list-style-type: none"> Identifies a wide area of vulnerabilities Reduces the scan time 	<ul style="list-style-type: none"> Expensive and restricts the number of websites Difficult to configure 	Commercial	High	Highly customizable and in-depth reporting
	<ul style="list-style-type: none"> Embedded device-protocol fuzzing The PenTester can easily set up new protocol test modules. An efficient combination can be selected to realize high-speed testing. 	<ul style="list-style-type: none"> High price Requires personal training and support 	Commercial	High	Adaptable and scalable to include a greater number of protocols, making it reliable to use across industries
	<ul style="list-style-type: none"> The generational fuzzer takes an intelligent, targeted approach to negative testing. Advanced file and protocol template fuzzers enable users to build their own test cases. Independent of operating system 	<ul style="list-style-type: none"> Customers cannot buy multiple licenses for a single protocol. 	Commercial	High	Comprehensive fuzzing and high vulnerabilities-detection efficiency

XCP: Universal Measurement and Calibration Protocol; CAPL: Computer Access Programming Language; GUI: graphical user interface.

(Continued)

Black-box testing has been the norm for security analysis across all industries. Although risk identification is almost always desirable through the PenTest, the tools rarely support the gray-box PenTest because the supply chain is very complex in automotive and aerospace industries, and multiple suppliers provide specific embedded devices for dedicated

functionalities. To protect their intellectual property, suppliers are reluctant to hand over their device architectures, which makes it difficult for them to use the white- and gray-box PenTest. As a result, most of the tools follow the black-box PenTest methodology. Our gray-box PenTest methodology, which combines risk analysis, architecture evaluation, and security

TABLE 1. The security-testing tools. (cont.)

Tool name	Application domain	Methodology	URL	Supplier	Portability			Usage scheme
					Windows	Linux	Mac	
Nmap	IT	Information gathering	https://nmap.org/	Nmap	✓	✓	✓	<ul style="list-style-type: none"> • Network scanning • Port scanning
AttifyOS	IoT	IoT PenTest framework	https://www.attify.com/attifyos	Attify				<ul style="list-style-type: none"> • IoT devices and connectivity PenTests
	IT, IoT, medical	Network and application PenTest framework	https://www.kali.org/	Kali by Offensive Security				<ul style="list-style-type: none"> • Network and applications PenTest
Keysight PenTest Platform and PathWave	Automotive, aerospace, energy	Vulnerabilities and their severity reports, exploitation methods	https://www.keysight.com/de/pd-3008427-pn-SA8710A/automotive-cyber-security-penetration-test-platform?nid=-31903.1276591&cc=DE&lc=ger	Keysight	✓			<ul style="list-style-type: none"> • Controllers, complex subsystems, complete-car PenTests, automotive, aerospace system, and wireless communication-system testing
CANBadger	Automotive	ECU PenTest	https://github.com/Gutenshit/CANBadger/wiki/Getting-the-board-ready		✓	✓		<ul style="list-style-type: none"> • ECU hacking using man-in-the middle attack and hijacking security access
Saleae Logic 8	Embedded devices	Firmware and logic analyzer	https://www.saleae.com/	Saleae	✓			<ul style="list-style-type: none"> • Embedded device logic analysis

testing, has proven highly efficient and effective to detect vulnerabilities.

Cybersecurity for connected systems has gained huge relevance with the convergence of IT and embedded systems. Because of the introduction of classic IT attack surfaces and vulnerabilities to critical infrastructures, the amount of attacks is fast growing. Since they are used across industries that have great relevance to our society, such systems must be thoroughly protected and hardened. Safety requires security as a mandatory condition, which means that any safety-critical system, at a minimum, needs to be protected. Security must be integrated early in the design phase to understand the threats and risks to embedded functions. Security today is mandatory due not only to its safety impact but product liability. It is not excusable anymore to say that hacking is inevitable. We must protect connected

systems as best we can and prove that we have taken the necessary actions in terms of processes, education, management, and technology. Testing plays a critical role in this process.

In this article, we looked to the testing environments for hardening. Specifically, we investigated current test tools, such as Metasploit, and evaluated how they can be used for security testing in converging IoT systems. While traditional security testing took a black-box approach, we recommend a grey-box methodology building upon a TARA and known component, interface, and network vulnerabilities. We showed how an initial security analysis and technical concept based on a given reference architecture shows threats and risks and is used to guide mitigation. We focused specifically on typical Ethernet-protection mechanisms, such as firewall policies, IDS/IPS and VLAN. Adding intelligence to testing tools by introducing machine- and deep-learning concepts will benefit

Strengths	Limitations	License	Cost	Notable qualities
<ul style="list-style-type: none"> • Bypass firewall or IDS • Service/operating-system detection capabilities 	<ul style="list-style-type: none"> • Scanning weaker devices and congested networks can cause an unintentional denial of service or network slowdown. 	GNU GPLv2	Low	Reliable scanning tool
<ul style="list-style-type: none"> • This distro contains the tools required to PenTest embedded firmware and software. • Radio-network PenTest 	<ul style="list-style-type: none"> • It needs to be installed on a virtual machine. 	Open source	Low	Effectiveness due to availability of most IoT PenTest tools under one distribution
<ul style="list-style-type: none"> • This is a Linux distribution that contains most of the tools such as Nmap and Wireshark) required to PenTest networks and software. 	<ul style="list-style-type: none"> • It needs to be installed on a virtual machine or separate hardware. 	Open source	Low	Availability of number of tools required for the PenTest
<ul style="list-style-type: none"> • Covers all relevant interfaces from hardware connectivity to the application layer • Continuous update of threat database • Keysight provides a complete platform, including software and hardware required for PenTests. 	<ul style="list-style-type: none"> • Using recommended Keysight hardware makes it expensive • Requires personal training and support 	Commercial	High	Easy integration with our own threat database
<ul style="list-style-type: none"> • Automated fuzzing and testing • CANBadger server helps in multiple devices 	<ul style="list-style-type: none"> • The hardware used for the PenTest is not readily available. We have to build it using the recommended block diagram. 	Open source	Low	Provides number of operating modes
<ul style="list-style-type: none"> • Tool provided with the hardware make analysis easy. 	<ul style="list-style-type: none"> • Speed may drop with increase in the channels used. 	Commercial	Medium	Ease of use

IDS: intrusion-detection system.

the process by improving metrics including speed, the response rate, the vulnerabilities found, and so forth.

We strive not only to provide guidance for specific misuse cases but to change the mentality of embedded-systems engineers toward designing for security rather than functionality. With the convergence of IT and embedded systems among industries, cybersecurity is a major requirement. Isolated mechanisms, such as distributed functionality in proprietary subsystems, protection at the component level, gateways and firewalls between components, and the validation of critical functions, is insufficient. Software-process evangelist Tom Gilb once observed, "If you don't actively attack risks, they will actively attack you." That mind-set should guide us toward improving security. Cybersecurity can never be comprehensive, but it can be vastly improved through risk-oriented testing with an optimized mix of strategies and tools. 🌐

REFERENCES

1. C. Ebert and A. Dubey, "Convergence of enterprise IT and embedded systems," *IEEE Softw.*, vol. 36, no. 3, pp. 92–97, May–June 2019. doi: 10.1109/MS.2019.2896508.
2. S. Morgan, "Global ransomware damage costs predicted to hit \$11.5 billion by 2019," *Cybercrime Mag.*, Nov. 14, 2017. [Online]. Available: <https://cybersecurityventures.com/ransomware-damage-report-2017-part-2/>
3. C. Osborne, "NonPetya ransomware forced Maersk to reinstall 4000 servers, 45000 PCs," *ZDNet*, Jan. 26, 2018. [Online]. Available: <https://www.zdnet.com/article/maersk-forced-to-reinstall-4000-servers-45000-pcs-due-to-notpetya-attack/>
4. A. Shostack, *Threat Modeling: Designing for Security*. Hoboken, NJ: Wiley, 2014.
5. M. Cheah, S. A. Shaikh, J. Bryans, and P. Wooderson, "Building an automotive security assurance case using systematic security evaluations," *Comput. Secur.*, vol.

77, pp. 360–379, Apr. 2018. doi: 10.1016/j.cose.2018.04.008.

6. C. Ebert, “Cyber security requirements engineering,” in *Requirements Engineering for Service and Cloud Computing*, M. Ramachandran and Z. Mahmood, Eds. Berlin: Springer, 2017, pp. 209–229.



CHRISTOF EBERT is the managing director at Vector Consulting Services. A professor at the University of Stuttgart and Sorbonne in Paris, he is a Senior Member of the IEEE and is the editor of the “Software Technology” department in *IEEE Software*. Contact him at christof.ebert@vector.com.



YOUSSEF REKIK is a consultant at Vector Consulting Services. His research interests focus on cybersecurity and software technology. Rekik received an M.S. in computer science from the National University of Computer Science of Tunisia, Tunis. Contact him at Youssef.Rekik@vector.com.



RAHUL KARADE is with the University of Stuttgart and Robert Bosch, both in Germany. His research interests are in embedded systems and security engineering. Previously, he worked at automotive and IT companies where he focused on cybersecurity along the IT embedded convergence. Contact him at rahulskarade@gmail.com.

ADVERTISER INFORMATION

Advertising Coordinator

Debbie Sims
 Email: dsims@computer.org
 Phone: +1 714-816-2138 | Fax: +1 714-821-4010

Advertising Sales Contacts

Mid-Atlantic US:
 Dawn Scoda
 Email: dscoda@computer.org
 Phone: +1 732-772-0160
 Cell: +1 732-685-6068 | Fax: +1 732-772-0164

Southwest US, California:
 Mike Hughes
 Email: mikehughes@computer.org
 Cell: +1 805-208-5882

Northeast, Europe, the Middle East and Africa:
 David Schissler
 Email: d.schissler@computer.org
 Phone: +1 508-394-4026

Central US, Northwest US, Southeast US, Asia/Pacific:
 Eric Kincaid
 Email: e.kincaid@computer.org
 Phone: +1 214-553-8513 | Fax: +1 888-886-8599
 Cell: +1 214-673-3742

Midwest US:
 Dave Jones
 Email: djones@computer.org
 Phone: +1 708-442-5633 Fax: +1 888-886-8599
 Cell: +1 708-624-9901

Jobs Board (West Coast and Asia), Classified Line Ads

Heather Bounadies
 Email: hbuonadies@computer.org
 Phone: +1 623-233-6575

Jobs Board (East Coast and Europe), SE Radio Podcast

Marie Thompson
 Email: marie.thompson@computer.org
 Phone: +1 714-813-5094



CALL FOR SPECIAL ISSUE PROPOSALS

Computer solicits special issue proposals from leaders and experts from a broad range of computing communities. Proposed themes/issues should address timely, emerging topics that will be of broad interest to *Computer's* readership. Special issues are an important component of *Computer*, as they deliver essential research insights and well-developed perspectives on new and established technologies and computing strategies.

We encourage submissions of high-quality proposals for the 2023 editorial calendar. Of particular interest are proposals centered on:

- offsite educational and business continuity technology challenges,
- privacy related to personal location tracking and surveillance (digital and physical),
- artificial intelligence and machine learning,
- technology's role in disrupted supply chains,
- misinformation and disinformation (fake information—malicious or non-malicious), and
- cyberwarfare/cyberterrorism

Proposal guidelines are available at:

www.computer.org/csdl/magazine/co/write-for-us/15911



DEPARTMENT: FORMAL METHODS

Formal Methods Boost Experimental Performance for Explainable AI

Frederik Gossen, *Department of Computer Science and Information Systems, University of Limerick, Limerick, Ireland, also Lero, the SFI Research Centre for Software, Ireland, and also Chair of Programming Systems, Faculty of Computer Science, TU Dortmund University, Dortmund, Germany*

Tiziana Margaria , *Department of Computer Science and Information Systems, University of Limerick, Limerick, Ireland and also Lero, the SFI Research Centre for Software, Ireland*

Bernhard Steffen, *Chair of Programming Systems, Faculty of Computer Science, TU Dortmund University, Dortmund, Germany*

MOTIVATION

IN “Towards Explainability in Machine Learning: The Formal Methods Way,”¹ we illustrated last year how Explainable AI can profit by formal methods in terms of its explainability. In fact, Explainable AI is a new branch of AI, directed to a finer granular understanding of how the fancy heuristics and experimental fine tuning of hyperparameters influence the outcomes of advanced AI tools and algorithms. We discussed the concept of “explanation,” and showed how the stronger meaning of explanation in terms of formal models leads to a precise characterization of the phenomenon under consideration. We illustrated how, following the Algebraic Decision Diagram (ADD)-based aggregation technique originally established in Gossen and Steffen’s work² we can produce precise information about and exact, deterministic prediction of the outcome from a random forest consisting of 100 trees.

For reasons of brevity, we used the familiar example of the Iris classification problem,³ which is small enough that we could publish pictures of the classifiers, this way explaining the underlying method, which uses aggregation using ADDs. We showed that the concise class characterization is highly relevant for practical applications. For example, it is useful to reverse the learned classification function that associates customer to products, when looking for the tailored public for a given product, for example, in order to obtain an optimized customer list for a specific product campaign, i.e., associating a product to

its potential customers. Moreover, the size and, therefore, the comprehensibility of the class characterization seem to hardly explode. In our example with only three classes, the model characterization ADD had more than 1100 nodes, while all the class characterization ADDs have less than 60 nodes: this size is still well within the range of a visual investigation.

In this article, we first revisit the essential traits of the method, and then discuss the performance effects.

ADD-BASED AGGREGATION FOR RANDOM FORESTS

Random forests, one of the most popular logic-based classifiers in machine learning, have typically large sizes because the larger they are, the more precise the outcome of their predictions. Random forests are a collection of many decision trees, each learned from a random sample of the training dataset. Each tree has a different structure and represents a different decision function. The training method is easy to understand and to implement, and at the same time achieves impressive classification accuracies in many applications. Figure 1 shows the random forest with three trees that were learned from the iris classification³ problem of the popular UCI dataset.⁴ It is the same that we have already used for illustration in Gossen *et al.*’s work.¹ The dataset lists dimensions of iris flowers’ sepals and petals for three different species of flowers (*iris setosa*, *iris virginiana*, and *iris versicolor*), that are our three classes.

Because different trees can produce different decisions for the same input data, it is important to have 69 many trees, and such that they consider different subsets 70 of the features.

When they individually classify previously unseen input data, every decision tree is evaluated separately,

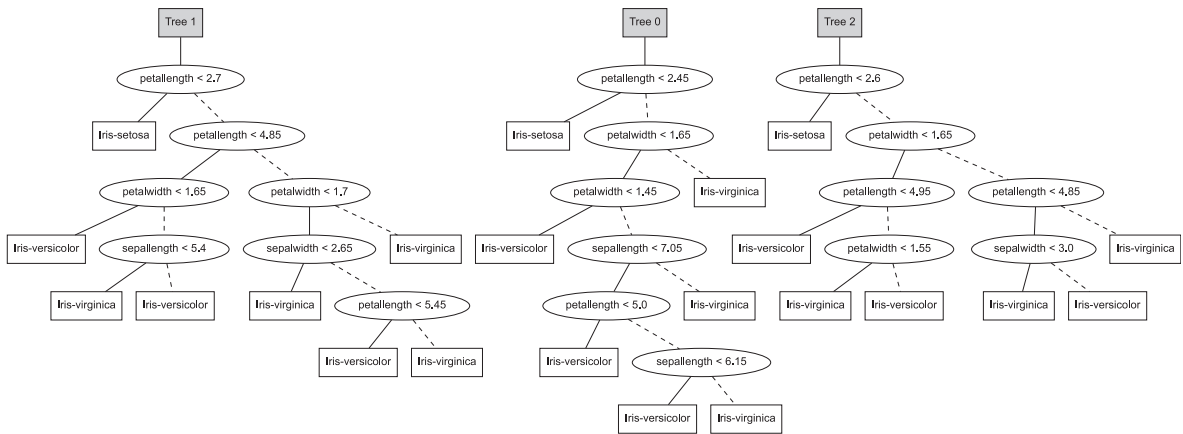


FIGURE 1. Random forest with three trees for the iris classification problem.³

potentially in parallel, and they collectively can produce the best classification result, which is inherently based on a “majority vote” principle. This is a way to implicitly aggregate the individual prediction to a consensus. Key advantage of this approach is the reduced variance compared to single decision trees. Key disadvantage is their evaluation time, which grows linearly with the number of trees in the forest. This is typically due to the high degree of redundancy in such evaluations: some predicates may be evaluated in many, if not in all trees of the forest! Figure 2 shows the algebraic aggregation of the random forest of Figure 1.

The ADD-based aggregation simply computes this aggregation in a systematic and representationally efficient way, using ADDs⁹ as an underlying data structure and aggregation mechanism. In Gossen *et al.*'s work,¹ we sketched a construction, called *algebraic aggregation* in Gossen and Steffen's work,² that transforms entire random forests into single trees which are guaranteed to have no redundant predicates in the following sense:

each path contains a predicate at most once, and only if it is relevant for classification. This transformation is provably optimal for a given order of the involved predicates.

Algebraic aggregation was presented in Gossen *et al.*'s work¹ as a mean for explainability: The resulting ADDs can be considered as *minimal white-box models* for the original black-box models given in terms of random forests.

Here, we position algebraic aggregation as a means for *runtime optimization*: in terms of execution efficiency, the abovementioned optimality results becomes: *Given a predicate ordering, which has to be obeyed during execution, the runtime is provably optimal.*

PERFORMANCE EVALUATION

Our experiments with popular datasets from the UCI Machine Learning Repository⁵ show performance gains of several orders of magnitude (see Figure 3 and Table 1). The results were achieved using the standard

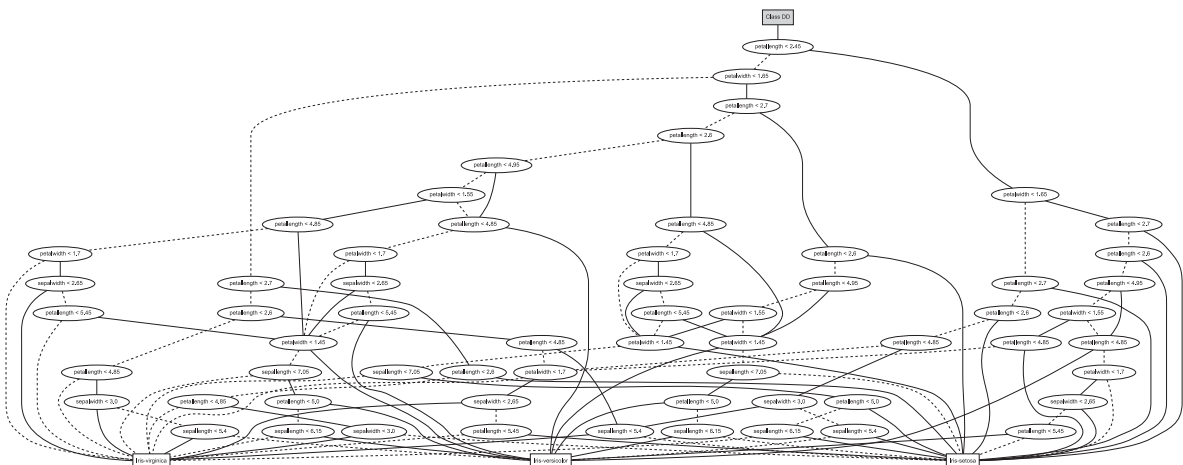


FIGURE 2. Algebraic aggregation of the random forest of Figure 1.

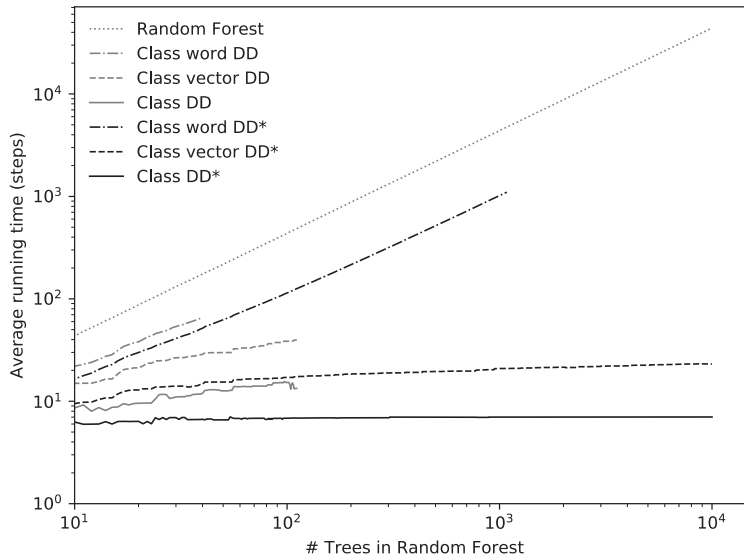


FIGURE 3. Average running time for classification over all examples in the iris dataset.

random forest implementation in Weka⁶ and on the ADD implementation of the ADD-Lib.^{7,8} These datasets were developed by independent parties with evaluations of this kind in mind, and we are not using any additional data for our transformation; thus, our analysis can be considered unbiased.

Experimental Setting: As wall-clock time measurements are very sensitive to implementation details and machine profiles, we decided to use, instead, the step count measure for performance analysis, which is in our eyes more objective in terms of measuring effort in a portable, machine-independent way. We consider, here, the steps through the corresponding data structures, and in the cases where the most frequent class must be computed at runtime, we account one additional step per read. For both the original random forest and the word-based decision diagram, these are n additional steps.

TABLE 1. Running time improvements for classification with random forests of 1000 trees.

Dataset	Running time in steps		Improvement factor
	Random forest	Aggregated ADD	
Balance scale	8014.12	7.73	1037
Breast cancer	13,020.03	17.11	761
Lenses	4431.42	3.67	1207
Iris	4395.77	6.97	631
Tic-Tac-Toe	10,733.68	14.22	755
Vote	6921.56	8.33	831

Results: Figure 3 shows the average evaluation times of the decision models for random forests of up to 10,000 trees. The curves of interest are the dotted line, indicating the evaluation time for the random forest, and the solid line for Class DD*, the fully optimized result ADD. The other curves indicate the performance of intermediate optimization steps, e.g., before unfeasible paths are eliminated. Unfeasible paths can be eliminated because they have contradicting predicate evaluations. The impact of unfeasible path elimination is indicated by the “asterisk.” Note that no tree of the original random forest has any unfeasible path. They only arise during aggregation.

The evaluation time of the original random forest grows linearly as expected: Every new tree contributes approximately the same running time. Due to the large number of trees relative to their individual sizes, our measurements appear as an almost straight line. Striking is the solid line indicating the performance of the fully optimized ADD: it hardly grows for large forest.

Key Insights: Intuitively, this seems to indicate that the complexity of the results of algebraic aggregation depends on the *inherent complexity* of the classification problem more than on the details introduced by the structure of the original random forest. This observation is also supported by Table 1, which summarizes the results for other datasets. As random forests are random, understanding what inherent complexity precisely means in this context is not so easy. It is apparent that the considered datasets are still “academic” and in this sense simple, as is indicated by Figure 4 and Table 2 showing how the size develops with increasing

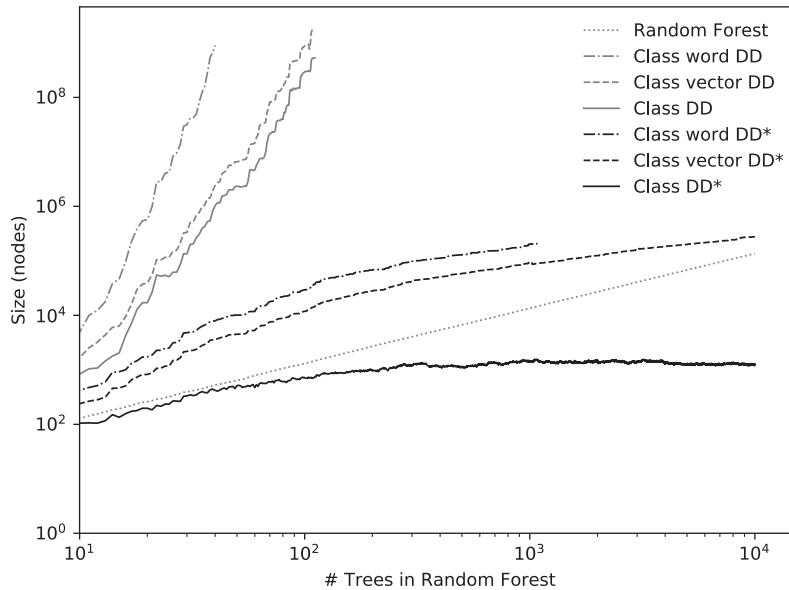


FIGURE 4. Size of the classifier representation in number of nodes.

forest size. Also here we observe a tendency to stabilization: the ADDs no longer grow for forests with more than 500 trees.

That this stabilization is a clear *semantic* phenomenon (as is also the inherent complexity) becomes apparent when comparing the curves with and without “asterisk”: It is the infeasible path reduction that controls the potential explosion! However, understanding the interplay between (the orthogonal) algebraic aggregation and infeasible paths elimination is difficult and, therefore, a fruitful field of future research.

CONCLUSION AND PERSPECTIVES

Classification is of increasing importance in a world that is increasingly steered by automated decision taking. This poses two problems: 1) the reliability of

TABLE 2. Size improvements for the classification with random forests of 1000 trees.

Dataset	Size in nodes		Improvement factor
	Random forest	Aggregated ADD	
Balance scale	214,844	139	1546
Breast cancer	546,504	3647	150
Lenses	14,132	11	1285
Iris	13,492	1458	9
Tic-Tac-Toe	570,976	1593	358
Vote	97,770	1168	84

classification, perhaps supported by means of explainability (see Gossen *et al.*'s work)¹ and 2) the velocity at which decisions can be taken in urgent situations. Algebraic aggregation² combined with semantic analyses like SMT solving for infeasible path reduction, has an impact on both explainability and velocity. In this article, we have indicated their impact concerning running time. Our evaluation leads to very promising results, with runtime improvements of several orders of magnitude. It has to be seen how these still initial results carry over to practical situations.

The main problem in practice will be the size explosion due to radically growing sizes of predicates: During the learning of the individual trees classification predicates are dynamically introduced in a way that strongly depends on the considered sets of samples. This means that small variations of the sample sets may introduce different thresholds and, therefore, different predicates serving the same purpose of separation. This is critical, because it may impose an exponential growth during algebraic aggregation. Here, unfeasible path elimination is a countermeasure, and its effectiveness for predicate elimination needs to be further studied.

On the other hand, the extension to weighted random forests, or even to forests that propose distributions, is quite straightforward as it only requires us to adapt the underlying leaf algebra.

Of course, these are first steps in a very ambitious new direction and it has to be seen how far this approach carries. Achieving scalability will probably also require some decomposition methods, perhaps in a

similar fashion as illustrated by the difference between model explanation and the considerably smaller class characterization. A more radical approach would be to introduce some notion of predicate normalization: collapse predicates that are indistinguishable in their effect on the considered training set. We consider such an optimization as extremely powerful, but it comes at a price: the classification function of the resulting ADDs is modified. This collapse may be acceptable, because the impact of this change can be formally analyzed and explained by looking at the performed collapses. Thus, we still preserve control over the change, so we may dynamically adapt the degree of collapse in order to achieve scalability while we maintain explainability. In the world of (statistical) machine learning, such a degree of control over the postprocessing steps should be sufficient, as we are not expecting 100% precision here anyway. 🤖

REFERENCES

1. F. Gossen, T. Margaria, and B. Steffen, "Towards explainability in machine learning: The formal methods way," *IT Professional*, vol. 22, no. 4, pp. 8–12, Jul./Aug. 2020, doi: 10.1109/MITP.2020.3005640.
2. F. Gossen and B. Steffen, "Algebraic aggregation random forests: Towards explainability and rapid evaluation," *Int. J. Softw. Tools Technol. Transfer*, vol. 22, pp. 8–12, Jul./Aug. 2020.
3. R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Ann. Eugenics*, vol. 7, no. 2, pp. 179–188, 1936.
4. UCI Machine Learning Repository: Iris Data Set. Accessed: Aug. 2021. [Online]. Available: archive.ics.uci.edu
5. D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: <http://archive.ics.uci.edu/ml>
6. I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*, 4th ed. San Francisco, CA, USA: Morgan Kaufmann, 2016.
7. F. Gossen, T. Margaria, A. Murtovi, S. Naujokat, and B. Steffen, "DSLs for decision services: A tutorial introduction to language-driven engineering," in *Proc. Leveraging Appl. Formal Methods, Verification Validation. Model.*, vol. 11244, 2018, pp. 546–564, doi: 10.1007/978-3-030-03418-4_33.
8. F. Gossen, A. Murtovi, J. Linden, and B. Steffen, "The Java library for algebraic decision diagrams." Accessed: Aug. 2021. [Online]. Available: <https://add-lib.scce.info>
9. R. Bahar *et al.*, "Algebraic decision diagrams and their applications," *Formal Methods Syst. Des.*, vol. 10, pp. 171–206, 1997, doi: 10.1023/A:1008699807402.

FREDERIK GOSSEN is currently an ML compiler engineer at Google, Munich, Germany. His research interests include algebraic decision diagrams, domain-specific program optimization, machine learning, and explainability. Gossen is currently working toward the Ph.D. degree jointly with the University of Limerick, Limerick, Ireland, and the TU Dortmund University, Dortmund, Germany. Since 2021, he has been a member of the Young Advisory Board to the new Immersive Software Engineering course at the University of Limerick, committed to shaping a novel practice-oriented academic entry into the field of computer science. Contact him at frederik.gossen@tu-dortmund.de.

TIZIANA MARGARIA is currently the chair of software systems with the Department of Computer Science and Information Systems, University of Limerick, Limerick, Ireland. She is also a Principal Investigator with Lero, the Irish National Centre for Software Research, Confirm, the Irish National Research Centre for Smart Manufacturing, and LDRC, the Limerick Digital Cancer Research Center. She co-directs the Irish Centre of Research and Training in Artificial Intelligence. She is also the CEO of METAFrame Technologies GmbH, Dortmund, Germany, a company that embeds domain-specific knowledge in "intelligent" low-code application development environments through formal methods. She is the vice chair of the IFIP 10.5 Working Group, a board member of FMICS, the ERCIM Working Group on Formal Methods for Industrial Critical Systems, and current VP and past president of the European Association of Software Science and Technology. She is a founder of ISoLA and the International Journal on Software Tools for Technology Transfer. She is a Fellow and vice president of the Board of the Irish Computer Society. Contact her at Tiziana.Margaria@ul.ie.

BERNHARD STEFFEN is currently the chair of programming systems and compiler construction at the University of Dortmund, Dortmund, Germany. He is the author of more than 400 refereed papers (h-index 65) concerning various aspects of formal (verification) methods and tools for program analysis, compiler optimization, model generation, testing, and service-oriented software development. In 2002, he wrote the first paper on test-based model generation using learning technology. He is a co-founder of the International Conference on Tools and Algorithms for the Construction and Analysis of Systems. In 2004, he co-founded ISoLA (Int. Symposium on Leveraging Applications of Formal Methods, Verification and Validation). He is the founder and Editor-in-Chief of *Software Tools for Technology Transfer* and the Editor of Springer's *Lecture Notes in Computer Science (LNCS)*. He was the recipient of the Most Influential PLDI Paper Award for Lazy Code Motion in 2002. In 2015, the accordingly developed LearnLib received the CAV Artefact Award. Contact him as Steffen@cs.tu-dortmund.de.

IEEE Computer Society Has You Covered!

WORLD-CLASS CONFERENCES — Stay ahead of the curve by attending one of our 210 globally recognized conferences.

DIGITAL LIBRARY — Easily access over 800k articles covering world-class peer-reviewed content in the IEEE Computer Society Digital Library.

CALLS FOR PAPERS — Discover opportunities to write and present your ground-breaking accomplishments.

EDUCATION — Strengthen your resume with the IEEE Computer Society Course Catalog and its range of offerings.

ADVANCE YOUR CAREER — Search the new positions posted in the IEEE Computer Society Jobs Board.

NETWORK — Make connections that count by participating in local Region, Section, and Chapter activities.

Explore all of the member benefits at www.computer.org today!



DEPARTMENT: INTERNET ETHICS

Artificial Intelligence and the Right to Explanation as a Human Right

Michael Winikoff  and Julija Sardelic , Victoria University of Wellington, Wellington 6012, New Zealand

Artificial intelligence (AI), which increasingly fuels Internet applications, has huge implications on the lives of ordinary people. This article examines explanations for AI decision-making as it concerns end users through the lens of human rights.

Technological developments can have significant social consequences, including human right implications. For example, the Internet has facilitated open communication and increased transparency, but also enabled wide-scale surveillance. The details of a technology can have a range of consequences for human rights. For instance, Internet protocols can be examined from a human rights perspective,¹ highlighting specific features of protocols that have consequences for human rights.

In this article, we focus on human rights implications of Internet applications. Internet applications are increasingly underpinned by artificial intelligence (AI) and reach not only into privacy and freedom of expression but also into the provision of public services and judicial processes that can affect the lives of ordinary people. We consider specifically the role of *explanation*, and how it links to human rights. Specifically, in what situations can the provision of explanation be motivated by human rights considerations?

AI AND EXPLAINABLE AI (XAI)

Systems that use AI techniques (“intelligent systems” hereafter) can behave in ways that are difficult for humans to understand. This can be because the system has additional knowledge (e.g., a navigation system routing around traffic congestion that the human is unaware of), or because the system operates very differently to a human (e.g., recognizing an image based on patterns of color, rather than the presence of geometric features).

This difficulty has led to the emergence of a whole subfield of research, XAI. XAI develops techniques for

intelligent systems to be able to meaningfully explain why they have taken certain actions, or made certain recommendations.² Explanation is widely seen as a crucial part of transparency,^{3,4} and is important in obtaining an appropriate level of trust in systems.^{5,6} Put simply: even if an intelligent system is doing a great job, if we do not understand its behavior, then we may not trust it.

When discussing explanation there are three important points to bear in mind. First, because AI involves a variety of techniques, so does XAI. Second, we assume that explanations are engineered to be honest, rather than deceptive. Finally, there is a distinction between providing a *generic* explanation for a mechanism, and providing a *specific* explanation for a given case. For example, a generic explanation might be that a credit card approval system was trained on a wide range of data, and validated, and that the data fed to it does not include ethnicity or gender. A specific explanation might be that the key factor that resulted in your application being declined was your income and your level of debt.

In general, when we discuss explanation (and especially understandable explanation) we are considering specific explanations rather than generic. One reason is that specific explanations are more likely to be understandable since they are about the specifics of a given case. Another reason is that, in order to avoid releasing valuable intellectual property, companies will likely limit the level of detail that they provide in generic explanations, making them unlikely to be sufficiently specific and detailed to be useful.

BUT WILL IT BE USED?

But when XAI develops mechanisms, will they be used? To what extent are organizations, such as companies, NGOs, governments, and government agencies, obliged to ensure that their intelligent systems provide meaningful explanations? Organizations might be reluctant to adopt explanations without some level of external compulsion. Providing explanation may

require additional work, and some organizations might have a desire to avoid transparency for various reasons. However, organizations are nevertheless accountable, including their obligations under (legally binding) international human rights treaties.

In this article, we therefore explore whether the external compulsion to provide explanation could be motivated by the international legal framework of human rights. Specifically, we pose the question: *in what situations can the right to explanation be positioned as a human right?*

Imagine a situation where some organization is taken to international court because they deploy or develop an intelligent system that is argued to violate human rights due to not providing explanation facilities. This could provide a means to exert external compulsion to provide explanations, within existing legal frameworks.

We are not the first to raise this idea of a “right to explanation.” Others have called for there to be a right to explanation,⁷ and the GDPR⁸ mentions a right to explanation. However, the mention of “an explanation of the decision” appears in the nonbinding recital. What actually appears in the binding clauses (13–15) is a requirement for “meaningful information about the logic involved,” rather than an explanation of a specific decision made. And, of course, the GDPR is only law in the European Union (although some rights apply outside the EU).

HUMAN RIGHTS ...

We focus on three key international human rights documents: the Universal Declaration on Human Rights (UDHR),⁹ the International Covenant on Civil and Political Rights (ICCPR),¹⁰ and the International Covenant on Economic, Social and Cultural Rights (ICESCR).¹¹ The UDHR was developed as a response to World War II, and was adopted by the UN General Assembly in December 1948. Although the UDHR is arguably the best-known document, it is in fact aspirational, not legally binding. However, the UDHR was followed by the two Covenants, which *are* legally binding on states that have ratified them. These three documents constitute the international bill of human rights.

The ICCPR covers civil and political rights every individual should have. These range from the most basic rights, such as the right not to be arbitrarily deprived of life or tortured or enslaved, to the right to freedom of thought and free assembly. Some of the rights in the ICCPR cannot be limited under any circumstances (such as the right not to be enslaved or tortured), whereas others, such as the freedom of movement and assembly, can be limited under certain conditions, such as public health or security emergency, but only if they comply with the rule of law of each individual country. The ICCPR roughly

corresponds to the previously developed Articles 1–21 of the UDHR. It is often forgotten that ICESCR is also a part of the package of the international human rights bill and it includes the right to work, education, and the adequate standard of living as articulated in Articles 22–29 of the UDHR. In the discussion that follows, we mostly refer to Articles in the UDHR. Although these are not legally binding, they do have counterpart articles in the other two documents that provide hard legal power.

In discussing human rights and intelligent systems, interpretation is required: the international human rights frameworks we consider were developed decades ago, prior to the development of modern computing or AI. It is therefore necessary, and indeed, entirely appropriate, to consider how human right principles and laws apply in today’s world.

There is a range of work that considers humans rights and intelligent systems,^{12,13} but it does not attempt to link human rights and the right to explanation. However, this body of work provides useful discussion of the wide range of ways in which the use of intelligent systems might impact various human rights.

Figure 1 shows a high-level mind map. It includes application domains (blue rectangles), issues (red hexagons), and human rights principles (depicted as green folders, with numbering corresponding to the clause of the UDHR, and labels being our own attempt to distill the essence of each depicted principle into a few words). In general, the human rights principles indicate rights that people have, and the issues show particular areas where these rights might be violated. This figure is obviously partial: it does not show all principles, all AI techniques, all application domains, or all issues. Instead, we focus on those where explanation is relevant. For example, there are a range of issues that relate to the use of facial recognition in a range of scenarios, but explanation is not clearly relevant. For readability, we also omit some links. For example, the yellow oval in the center (“Intelligent Systems ...”) should be linked to the “Warfare” rectangle, and also directly linked to the issue of Discrimination, as should a number of human rights (e.g., principles 25, 26).

...AND THE RIGHT TO EXPLANATION

Looking at Figure 1, where can a right to explanation be positioned as arising from existing human rights? We highlight two broad areas where we argue that in order to avoid infringements of human rights, explanation could be required.

Discrimination and the Right to Explanation

The first area concerns situations where an intelligent system is making decisions or recommendations in

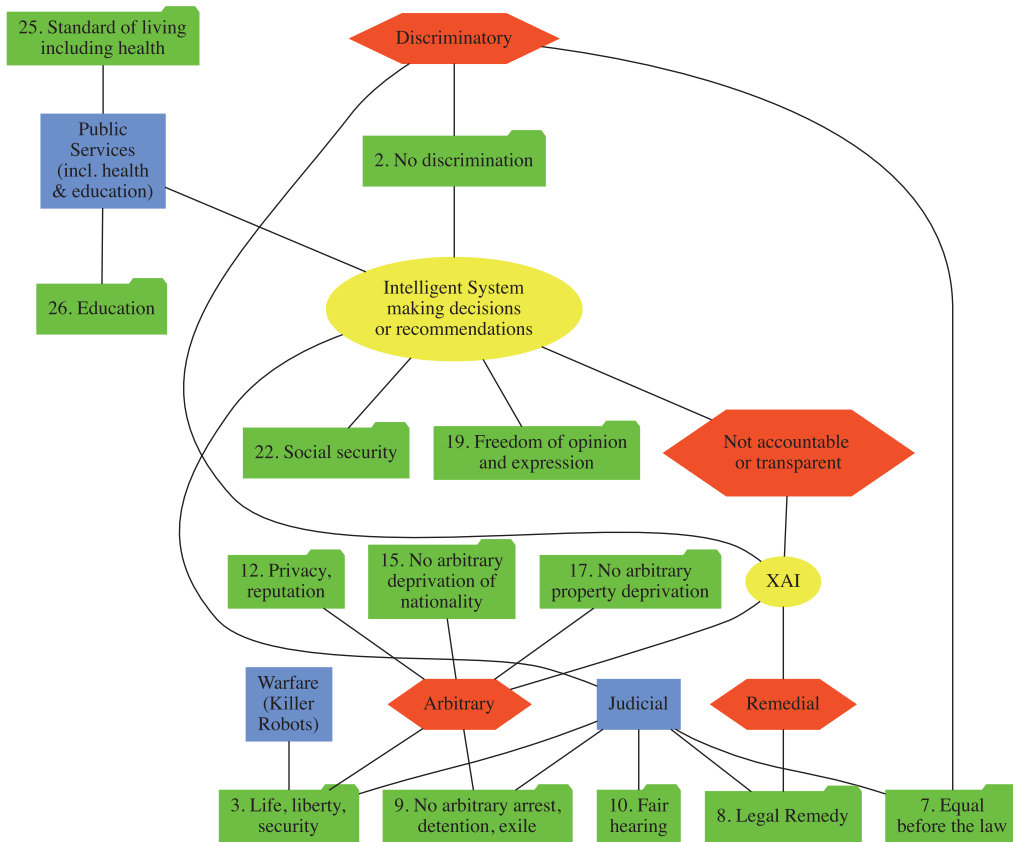


FIGURE 1. High-level mind map showing (some) links between human rights principles (green folders, with numbering corresponding to UDHR clauses), issues (red hexagons), and application domains (blue rectangles).

contexts where this can affect human rights, and there is potential for discrimination. Human rights that can be affected include “the right to social security” (Article 22 of UDHR), “the right to a standard of living ...and well-being ...including food, clothing, housing and medical care and necessary social services” (Article 25 of UDHR), and arguably also the right to access higher education (“higher education shall be equally accessible to all on the basis of merit,” Article 26 of UDHR). There is potential for discrimination where decisions are made on the basis of data that includes prohibited criteria “such as race, colour, sex, language, religion, political or other opinion, national or social origin, property, birth or other status” (Article 2 of UDHR), or proxies for these.¹⁴

The possibility of a machine learning system identifying proxies for these, such as postcode as a proxy for ethnicity, makes it difficult to argue that discrimination is not occurring, unless specific explanation can be provided. Unfortunately, there are a range of examples that demonstrate that discrimination can occur in a range of ways, for example, gender bias in credit card

limits.¹⁵ Indeed, in the example of the Apple credit card, the issuing company claimed that there was no bias in the algorithm, justifying this with the (invalid) argument that gender was not given as an input.

Had effective explanation facilities been available, this would have made it easier to detect the discrimination, by identifying the factors that justified a range of individual decisions. Although in this case gender would not have been an explanatory factor, the existence of proxies for gender that played a role in the decision should have been apparent, and prompted investigation. Indeed, the use of an explanation mechanism may have even led to this issue being identified before the system was deployed.

Judicial Applications of Intelligent Systems and the Right to Explanation

The second area concerns judicial applications of intelligent systems (blue rectangle “Judicial” toward the bottom of Figure 1). This area is singled out from other application domains for two reasons. First, legal rights are quite prominent in human rights documents

(Articles 6–11 of UDHR). Second, the judicial system is supposed to be highly accountable, more so than profit-driven companies.

There are many ways in which the use of intelligent systems in a judicial context can infringe on human rights, including being used to provide evidence in a trial, especially where the outcome of the trial affects physical freedom (Articles 3 and 9 of UDHR), privacy (Article 12), property ownership (Article 17), or the removal of citizenship (Article 15). In considering the role of explanation, it is important to highlight the word “arbitrary.” This word appears in a number of UDHR clauses and, we argue, links directly and strongly to explanation: without explanation, one cannot rule out that something was indeed arbitrary. Furthermore, without explanation, it is not possible to ensure that one has been treated equally before the law “*without any discrimination*” (Article 7) and that one has had a fair hearing “*by an independent and impartial tribunal*” (Article 10).

Again, there are extant examples of human rights being infringed by the inappropriate use of intelligent systems, for example, the COMPAS case,^{16,17} where discriminatory risk assessments were used in making parole decisions. Again, had explanation facilities been provided, we might expect that the issues would have been surfaced earlier, or even been avoided, had judges the ability to see why the system flagged a person as a low, or as a high risk.

Another link to explanation is the right “*To be informed promptly and in detail in a language which he [or she] understands of the nature and cause of the charge against him [or her]*” (Article 13 of ICCPR), as well as the right to remedy when fundamental rights are violated (Article 8 of UDHR): providing appropriate remedy for violation of rights requires the ability to detect such violations, which, as we have argued earlier, can require explanation, for instance to detect discrimination.

Other Cases

In addition to these two areas where we argue that the right to explanation follows as a natural consequence of existing human rights, there are also a range of other areas where an argument could be made, although we feel it is somewhat weaker in these cases.

The first relates to Article 19 of UDHR, which is the “*right to freedom of opinion and expression*.” In the context of social media where what each person sees is determined by an (opaque) algorithm, it could be argued that the lack of explanation results in a situation where the right to express opinions and access others’ opinions is being impinged in arbitrary ways. Risse¹⁸ also flags the issue of AI-generated fake videos as a threat to freedom of speech and expression.

The second concerns the application domain of health, which relates to the right to medical care

(Article 25). Specifically, consider an intelligent system that makes medical recommendations relating to diagnosis and/or treatment. One specific form of discrimination that could arise is where recommendations are based on a biased dataset, which results in them being more effective for particular groups. This is not the more general discrimination case where access is affected, but a more subtle situation where some groups may have equal access, but receive less effective treatment.

Finally, we turn to lethal autonomous weapons systems (LAWS), also referred to as “killer robots.” While clearly these have potential to result in arbitrary loss of life (violating Article 6 of the ICCPR), it is less clear that providing explanation facilities would help avoid this. Additionally, war and conflict sits in a different legal context, and is best analyzed separately. It is also worth highlighting that whereas, say, a bank could be taken to court to force it to change practices to avoid violating human rights, it is unlikely that this would be practical in the case of use of LAWS, especially if the user is a nonstate actor.

CONCLUSION

Before concluding, we note that although these issues apply when decisions are made by humans, for instance, bank managers declining loans in a potentially biased way, the context is different, and suitable solutions are therefore different. In the case of human decision makers, there are many humans making decision, and their biases may be unconscious, or be lied about. This means that asking an individual bank manager for an explanation is not particularly useful: it only applies to a small portion of the cases, and may not reflect their real decision-making process. By contrast, an intelligent system applies the same mechanism consistently to all cases, and can be engineered to provide explanations that do reflect its decision-making.

In conclusion, we have considered the question of whether a right to explanation could be argued to be a natural consequence of existing human rights. We have considered existing, legally binding, human rights, and considered a range of ways in which intelligent systems could be used in a way that might infringe these rights, and where providing explanation would help avoid these infringements. This analysis paves the way to enforcing a right to explanation in at least these situations, without having to develop new laws. 🌐

REFERENCES

1. N. ten Oever and C. Cath, “Research into human rights protocol considerations,” *Request Comment Ser.*, vol. 8280, pp. 1–81, 2017.

2. T. Miller, "Explanation in artificial intelligence: Insights from the social sciences," *Artif. Intell.*, vol. 267, pp. 1–38, 2019.
3. The IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems, *Ethically Aligned Design: A Vision for Prioritizing Human Well-Being With Autonomous and Intelligent Systems, Version 2*, 2017. [Online]. Available: http://standards.ieee.org/develop/indconn/ec/autonomous_systems.html
4. V. Dignum, *Responsible Artificial Intelligence: How to Develop and Use AI in a Responsible Way*. New York, NY, USA: Springer, 2019.
5. P. K. Murukannaiah and M. P. Singh, "From machine ethics to internet ethics: Broadening the horizon," *IEEE Internet Comput.*, vol. 24, no. 3, pp. 51–57, May/Jun. 2020.
6. M. Winikoff, "Towards trusting autonomous systems," in *Proc. 5th Int. Workshop Eng. Multi-Agent Syst., Revised Sel. Papers*, 2017, vol. 10738, pp. 3–20.
7. J. F. Weaver, "Artificial intelligence owes you an explanation: When an A.I. does something, you should be able to ask, 'Why?'," 2017. [Online]. Available: <https://slate.com/technology/2017/05/why-artificial-intelligences-should-have-to-explain-their-actions.html>
8. European Union, *EU General Data Protection Regulation*, Apr. 2016. [Online]. Available: <http://tinyurl.com/GDPREU2016>
9. The United Nations, *The Universal Declaration of Human Rights*, 1948. [Online]. Available: <https://www.un.org/en/universal-declaration-human-rights/>
10. The United Nations, *International Covenant on Civil and Political Rights*, 1966. [Online]. Available: <https://www.ohchr.org/en/professionalinterest/pages/ccpr.aspx>
11. The United Nations, *International Covenant on Economic, Social and Cultural Rights*, 1966. [Online]. Available: <https://www.ohchr.org/en/professionalinterest/pages/cescr.aspx>
12. Access Now, *Human Rights in the Age of Artificial Intelligence*, 2018. [Online]. Available: <https://www.accessnow.org/cms/assets/uploads/2018/11/AI-and-Human-Rights.pdf>
13. High-Level Expert Group on Artificial Intelligence, *Ethics Guidelines for Trustworthy AI*, 2019. [Online]. Available: <https://ec.europa.eu/futurium/en/ai-alliance-consultation/guidelines>
14. C. O'Neil, *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*, 1st ed. New York, NY, USA: Crown, 2016.
15. W. Knight, "The Apple card didn't 'see' gender—and that's the problem," 2019. [Online]. Available: <https://www.wired.com/story/the-apple-card-didnt-see-genderand-thats-the-problem/>
16. J. Angwin, J. Larson, S. Mattu, and L. Kirchner, *Machine Bias*, 2016. [Online]. Available: <https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing>
17. J. Larson, S. Mattu, L. Kirchner, and J. Angwin, "How we analyzed the COMPAS recidivism algorithm," 2016. [Online]. Available: <https://www.propublica.org/article/how-we-analyzed-the-compas-recidivism-algorithm>
18. M. Risse, "Human rights and artificial intelligence: An urgently needed agenda," *Human Rights Quart.*, vol. 41, no. 1, pp. 1–16, Feb. 2019.

MICHAEL WINIKOFF is currently a Full Professor with the School of Information Management, Victoria University of Wellington, Wellington, New Zealand. He is known for his work on engineering aspects of autonomous systems, including the *Prometheus* methodology. More recently, he has been working on issues relating to trust in autonomous systems, including verification and explanation. He is on the IFAAMAS board of directors. He is on the editorial board of the *Journal of Autonomous Agents and Multi-Agent Systems*, and was the Program Co-Chair and General Co-Chair for the conference on Autonomous Agents and Multi-Agent Systems in 2012 and 2017, respectively. He is the corresponding author of this article. Contact him at michael.winikoff@vuw.ac.nz.

JULIJA SARDELIC is currently a Lecturer in Political Science and International Relations Program with the Victoria University of Wellington, Wellington, New Zealand. Here, she teaches courses on human rights and coteaches courses on the politics of forced migration. Her research deals with different aspects of human rights, but she has particularly focused on the rights of minorities and forced migrants. Her book entitled "*The Fringes of Citizenship*" will be published in 2021 with Manchester University Press. Prior to her position in Wellington, she was a Marie Skłodowska Curie Fellow at Leuven International and European Studies (University of Leuven, Leuven, Belgium). Before that, she was also a Post-doctoral Researcher with the University of Liverpool, Max Weber Fellow with the European University Institute, and Research Fellow with the University of Edinburgh. Contact her at julija.sardelic@vuw.ac.nz.

IEEE COMPUTER SOCIETY D&I FUND

Drive Diversity & Inclusion in Computing



Supporting projects and programs that positively impact diversity, equity, and inclusion throughout the computing community.



Do you have a great idea for new programs that will positively impact diversity, equity, and inclusion throughout the computing community?

The IEEE Computer Society Diversity & Inclusion Committee seeks proposals for projects, programs, and events that further its mission. New programs that deliver education, outreach, and support, including, but not limited to, mentoring programs at conferences, panel discussions, and webinars, are welcomed.

Help propel the Computer Society's D&I programs—submit a proposal today!

<https://bit.ly/CS-Diversity-CFP>



Donations to the IEEE Computer Society D&I Fund are welcome!



IEEE Foundation

DEPARTMENT: PERVASIVE HEALTHCARE

Pervasive Healthcare IRBs and Ethics Reviews in Research: Going Beyond the Paperwork

Jina Huh-Yoo , Drexel University, Philadelphia, PA, 19104, USA

Reema Kadri , University of Michigan, Ann Arbor, MI, 48109, USA

Lorraine R. Buis , University of Michigan, Ann Arbor, MI, 48109, USA

Ethics committee approval is often viewed as a necessary hoop to jump through before a research study can begin. However, when focusing primarily on the administrative burden associated with this process, researchers may miss the opportunity to use this process as a scaffolding for thinking critically about the risks and benefits of the research for participants.

Human subjects research is increasingly used in computing and engineering within the context of user centered design. Although many researchers in these fields may be new to working with human subjects, even the most seasoned clinical researchers struggle to think thoroughly through ethical considerations. Training in the responsible conduct of research with human subjects is often routine in academia, yet for many researchers, even those with a lot of experience working with human subjects, the process of thinking through ethical issues present in our own research may be overlooked. We often believe that usability and small pilot studies with participants is harmless. However, no matter how seemingly benign, technologies and research may involve risks that are not intuitive. In this article, we discuss thinking through the process of conducting user research with technology to identify implications around risks. To aid this discussion, we present a brief history of research ethics oversight committees and ways to work through understanding and communicating risks of technology-based research.

REGULATING HUMAN SUBJECT RESEARCH: RESEARCH ETHICS COMMITTEES AND INSTITUTIONAL REVIEW BOARDS

Adequate training in the responsible conduct of research is essential for public funding and is a basic

fundamental requirement for approving the conduct of human research in the U.S.; however, training typically relies on self-paced training modules, where the level of understanding of ethical principles can vary greatly depending on how serious the learner is in learning the material. Moreover, without adequate research experience involving human subjects, learning to think through research risks can be difficult. Even experienced researchers often overlook important ethical considerations reinforcing the fact that this is a skill that requires years to hone. Although most people can likely identify gross violations of ethical standards, more unassuming research activities, such as small formative usability studies and small pilots may seem incredibly benign and risk free to researchers.

Research involving humans is typically subject to review by some type of ethics committee. Outside of the U.S., these are often known as Ethics Review Boards or Ethics Committees, whereas in the U.S., they are commonly referred to as Institutional Review Boards (IRBs). IRBs are formally designated groups that review and monitor research involving human subjects. IRBs often exist within research institutions, but independent IRBs also exist and can be contracted to regulate research originating from institutions and companies without their own. IRBs have the authority to approve, require modifications, or deny research protocols based on ethical concerns. The goal of an IRB is not to judge the quality of research proposed; rather they seek to protect the welfare of human research subjects.

IRBs have a long history in the U.S. From 1932 to 1972, the U.S. Public Health Service, in collaboration with Tuskegee University, conducted the Tuskegee Syphilis study.¹ This study sought to observe the effects of untreated syphilis in African American men, some of whom had syphilis and controls who did not. Participants with syphilis were not told of their diagnosis and despite promises of free medical care, adequate care

1536-1268 © 2021 IEEE
 Digital Object Identifier 10.1109/MPRV.2020.3044099
 Date of current version 9 March 2021.

was not provided. In 1972, a whistleblower leaked details on the unethical conduct of the study to the press. This led to major changes in U.S. law and regulation on the protection of human subjects, including the requirement of informed consent. The Tuskegee Syphilis study, along with other high-profile instances of unethical human subject research, contributed to the development of The Declaration of Helsinki, The Nuremberg Code, and The Belmont Report, which have shaped the ethical guidelines that our IRBs follow today, with the primary goal being to “do no harm.”²

UNDERSTANDING RISK: MORE THAN JUST PHYSICAL HARM

Many researchers believe that their studies pose “no risk” to participants; however, most ethics committees acknowledge that no research study is free from risk. Risks may be minimal, but they are not nonexistent. Indeed, in our previous work, Huh-Yoo and Radar demonstrated that IRB members viewed the collection and use of digital data of today’s technology as having added risk above and beyond that of nondigital data, citing additional concerns such as breaches of confidentiality, unintended collection of sensitive data, and unauthorized reuse.³ Physical harm to a participant is the most obvious possible harm, but there are more to consider, including psychological, social, economic, and legal harm, as well as loss of autonomy and any forms of injustice documented as harms in the Belmont Report.² The Tuskegee Syphilis study shows evident physical harm and the systemic racism is inherent in the study premise and design. In the present time, when many of us are reflecting on how to be more antiracist in our own work, thinking about more subtle harms and risks is essential when considering the ethical issues inherent in our own work. As we develop and evaluate novel technologies, it is imperative that we think about how use of these products can influence a person and their physical self, as well as their behavior, psychological state, social standing, privacy, finances, and/or legal affairs.

To illustrate, in 2012, Facebook conducted a one-week study among randomly selected users to test the effects of manipulating algorithms that decide what to present on a user’s newsfeed. Known as the Emotion Contagion Experiment, researchers manipulated users’ news feeds to test, among other things, whether fewer positive posts in the news feed can lead to greater expressions of sadness by the user.⁴ The researchers of this study did not obtain full IRB review and approval, nor did they engage in informed consent processes with those who received the

experimental condition. This article received a lot of media attention and was strongly criticized for being ethically problematic due to its lack of informed consent. Moreover, the use of user-based data was seen by some as a violation of identity-based norms and an exploitation of the vulnerability of users who self-disclose on social media with no control over how their data are presented.⁵ The study, published in the Proceedings of the National Academy of Sciences, even drew editorial comment explaining the reasoning behind the decision to publish the study, while acknowledging that it “may have involved practices that were not fully consistent with the principles of obtaining informed consent and allowing participants to opt out.”⁶ Facebook argued that they were within their rights to manipulate their service as specified in their Terms of Service, a point that many scholars have debated. Whether Facebook was within their rights to conduct this study is not for us to decide; however, it can be a useful case study to illustrate potential risks inherent in study design. Without informed consent, some users were unknowingly participating in an experiment with demonstrated effects on psychological state. Moreover, if the experimental condition had enough of an effect, it is possible that the intentionally suppressed positive posts could have affected the user’s social standing among others and may have led to social and psychological harm.

BEYOND RESEARCH ETHICS APPROVAL: REFLECTING ON RISK

In formative technology development stages, ethics committee oversight may not be required, but we should still think through ethical issues inherent in our processes. What would seem to be “minimal risk” for testing (no greater risk than those risks encountered in daily life; a condition for IRB exemption in the U.S.) can generate risks that may not be minimal. In fact, we would argue that in the absence of IRB or ethics committee oversight, such as in formative development and small scale usability testing, thinking critically about ethical considerations in our work is even more imperative and our own assessment of risk is even more important. Without oversight, it falls to us as researchers to ensure that we take the full responsibility of protecting our human subjects from harm and mitigating risks to the best of our ability.

For instance, smart home devices are owned by many. Conducting usability testing on Amazon Alexa, a digital voice assistant, seems to pose minimal risk. However, if tested in the home, any bystanders who speak within its listening range may be recorded,

regardless of whether they agreed to participate in research using the device. Potential risks exist if Alexa were to capture mandatory reporting events (e.g., child abuse) or other sensitive information (e.g., undocumented residency status, illegal activity, etc.). Moreover, due to algorithmic recommendations based on users' input, the output from Alexa could unintentionally reveal private information to other household members. It is also important to consider the fact that information captured by Alexa is not under the researcher's ability to manage, control, or discard, and third party vendors that produce the device own the data (e.g., Amazon) and could be subpoenaed by the government or other interested parties. Even testing by members of the development or research team should be carefully considered, and depending on your IRB, could require its own approval process and informed consent.

The issues surrounding control are not uncommon. Novel technologies often involve complicated data flows that may involve third party vendors that are outside a researcher's control. Vendors often cannot or will not give clear answers to describe how information flows due to trade secrets, machine learning algorithms, and/or multiple third-party companies embedded in a technology. For example, if you are building a mobile app that incorporates Fitbit data via an API, your app relies on third-party data. The data you receive are not data you actually "own" and may change. Fitbit can change the way it collects and shares data at any time, which could affect your product and your research. When forecasting risks inherent to a new technology, including its design and evaluation, it is essential that all parties of the research project (e.g., researchers, participants, research institution, and funders) understand how the technology works, and to make sure that there are appropriate safeguards in place. When trying to understand these concerns, we must ask how does the data flow from the user and how is it reused with or without the user's consent? Is this ethical? Is this putting anyone at risk?

Having approval from an ethics committee is not sufficient to prevent unanticipated events from happening. Despite our best intentions, the fact remains that when working with new technologies, existing regulatory approaches may not fully address new risks to testers, study participants, and household members or bystanders. It is imperative that researchers have the skills and training to recognize problems as they occur, and to work with their research ethics committees to handle

unanticipated situations in a responsible and ethical manner.

**BEYOND GETTING A SIGNATURE:
MITIGATING RISKS BY
COMMUNICATING WITH
PARTICIPANTS**

Ensuring that participants understand the risks and benefits of a research study through the informed consent process is vital. Although there are instances where informed consent requirements can be waived, this decision should not be considered lightly. Careful planning and well-informed consent processes can mitigate risks involved in the research process, in the experimental and/or control conditions, in data collection and management, and in data analysis and reporting. Although these elements pertain to all research studies, those that involve technology tend to involve an added layer of potential ethical considerations as data are often collected or generated by the user and in the cases where the technology is created by a third party, ownership of the data is ambiguous.

The informed consent process provides an opportunity for researchers to communicate directly to participants about what their involvement entails, what risks are anticipated, and what steps are taken to mitigate risk. This process affords participants autonomy in deciding whether to participate in the study or not. However, when the study involves new kinds of technology that is unfamiliar to the participants, in addition to explaining study procedures, researchers need to make sure that the technology is explained in a clear, complete, and accurate way, in language the participant can understand. The informed consent process might be different depending on the tech savviness of the participant but should always be presented in a way that allows participants with a lower educational background to make an informed choice to participate. If participants have difficulties understanding how the technology works, or even the concepts of privacy, anonymity, and confidentiality, one of the core principles of ethical research—voluntariness—will be challenged.

A good informed consent process should adequately describe the technology used within the study and the anticipated risks and benefits of the technology and research process. This includes the data that will be collected by the study device and the study team, as well as how that data will not only be accessed by the researcher but also any involved third parties where data may flow. Often overlooked risks to communication include the fact that we may

accidentally collect information participants did not approve or know about, or as mentioned earlier, information about nonparticipants. Participants need to know what data is shared and with whom, as well as the fact that that data shared outside the research team may not be controlled.

REASONS TO STRIVE FOR ETHICAL RESEARCH CONDUCT AND OVERSIGHT

We as researchers have a moral and professional obligation to do our best to mitigate the risk of harm to the research participants. Although we may not be able to erase all potential harm, taking steps to ensure that participants understand the potential risks and benefits and the procedures we undertake to mitigate risk, is well within our control. However, there are many practical reasons beyond this obligation that support the need to subject our research protocols to ethics committee oversight. Statements on research ethics oversight or exemption are often required for publishing research results. In addition, proper ethics training and oversight are often a requirement of funding agencies. Grant review panels are tasked with judging whether proposals consider ethical aspects of their proposed studies and take appropriate steps to mitigate unnecessary risks. Failure to adequately address risks could make a difference in funding decisions. In industry, funders may be investors and consumers rather than grant agencies. In this case, subjecting research studies to ethics approval may not be required. Still, it is considered a best practice and is viewed as a strategy to mitigate corporate risk. In addition, many businesses and organizations outside of academia may not have internal ethics committees, thus contracting with an external IRB is an inexpensive way to ensure adequate research conduct. In the Facebook Emotion Contagion experiment, not only could the controversy affect investors and their willingness to invest in the company, but it may have affected users and their trust for the company.

It may be tempting to think about obtaining ethics committee approval simply as an exercise in routine paperwork. However, we can reframe the approval process as a way to think critically about risks as we prepare documents and scripts that appropriately describe the risks, benefits, and protections. This includes understanding the risks inherent in the technology and its use, as well as the risks inherent in the design of the research

study, including how participants are identified and recruited, how they are informed about the risks and benefits of the research study and whether the incentives offered are coercive, what they are asked to do, what data are collected, and how that data are analyzed and reported.

CONCLUSION

In this piece, we unraveled the multifaceted processes of understanding and communicating risks of technology research that involves human subjects. Through focusing on the history and importance of IRB and ethics committee oversight, we highlighted the importance of going beyond our moral obligations to conduct ethical research and pointed out the practical and logistical reasons for adhering to research ethics review procedures. We urged the critical need to think proactively, rather than retroactively, of what risks we introduce to study participants, including potential physical, emotional, social, legal, and economic harms. We also examined how information flows to external entities outside of the research process need special consideration. Finally, informed consent can be reframed as a process beyond receiving signatures for formal, liability purposes. It is an opportunity to communicate risks to participants, and to put into place strategies to mitigate those risks. 🧠

ACKNOWLEDGMENT

The authors would like to acknowledge Emilee Rader for her collaboration with the first author on their interview study with U.S. IRB members that laid the foundation for this discussion.

REFERENCES

1. A. M. Brandt, "Racism and research: The case of the Tuskegee syphilis study," *Hastings Center Rep.*, pp. 21–29, 1978. <https://doi.org/10.2307/3561468>
2. National Commission for the Protection of Human Subjects of Biomedical and Behavioral Research, "The Belmont report: Ethical principles and guidelines for the protection of human subjects of research," *J. Dent Res.*, vol. 81, pp. 4–13, 1979.
3. J. Huh-Yoo and E. Rader, "It's the wild, wild west: Lessons learned from IRB members' risk perceptions toward digital research data," *Proc. ACM Hum.-Comput. Interact.*, vol. 4, 2020, Art. no. 039. <https://doi.org/10.1145/3392868>.

4. A. D. I. Kramer, J. E. Guillory, and J. T. Hancock, "Experimental evidence of massive-scale emotional contagion through social networks," *Proc. Nat. Acad. Sci.*, vol. 111, no. 24, pp. 8788–8790, 2014.
5. C. Flick, "Informed consent and the Facebook emotional manipulation study," *Res. Ethics*, vol. 12, no. 1, pp. 14–28, 2015. <https://doi.org/10.1177/1747016115599568>
6. I. M. Verma, "Editorial expression of concern: Experimental evidence of massive-scale emotional contagion through social networks," *Proc. Nat. Acad. Sci.*, vol. 111, no. 29, pp. 10779 LP–10779, 2014. <https://doi.org/10.1073/pnas.1412469111>

JINA HUH-YOO is an Assistant Professor of human-centered computing with the Department of Information Science, College of Computing & Informatics, Drexel University, Philadelphia, PA, and a Human Research Committee member of the American Psychological Association. Contact her at jh3767@drexel.edu.

REEMA KADRI is a Research Project Manager with the Department of Family Medicine, University of Michigan, Ann Arbor, MI. Contact her at rkadri@umich.edu.

LORRAINE R. BUIS is an Associate Professor with the Department of Family Medicine and the School of Information, University of Michigan, Ann Arbor, MI. Contact her at buisl@umich.edu.

IEEE COMPUTER SOCIETY
Call for Papers

Write for the IEEE Computer Society's authoritative computing publications and conferences.

GET PUBLISHED
www.computer.org/cfp

IEEE COMPUTER SOCIETY

IEEE

Get Published in the New *IEEE Open Journal of the Computer Society*

Submit a paper today to the premier new open access journal in computing and information technology.

Your research will benefit from the IEEE marketing launch and 5 million unique monthly users of the IEEE *Xplore*[®] Digital Library. Plus, this journal is fully open and compliant with funder mandates, including Plan S.

Submit your paper today!

Visit www.computer.org/oj to learn more.



Building an Accessible Digital World

Sarah Horton, *University of Southampton*

Who is responsible for making sure the digital world is accessible to people with disabilities? Examined through the lens of the physical built world, those of us who design and construct the digital environment have a significant role to play. But are we prepared?

In his September 2020 “Software Engineering” column, “A Brief History of Software Professionalism and the Way Forward,” Phil Laplante poses the question:

Society expects a standard of competence, professionalism, and accountability from its doctors, nurses, and other professionals who hold lives in trust. Yet anyone can write software that can appear in or interact with critical systems, so what does “software professional” mean, and what are society’s expectations for those individuals?¹

As an accidental technology professional, this question resonated for me. After stumbling into a tech career with a music theory degree and some natural aptitude, I have been building my competence, professionalism, and accountability as I go. Discovering accessibility helped me realize the impact of my design decisions and coding approaches and the need to prioritize user needs. Technology done right can open doors to opportunity and participation for people with disabilities. Technology designed and built without attention to accessibility produces barriers.

In my previous position as a digital accessibility consultant at The Paciello Group, I had the privilege of working alongside and learning from the most

competent and professional accessibility specialists out there. More recently, I had the opportunity to learn about accessibility and disability inclusion by completing the Americans with Disabilities Act (ADA) Coordinator Training Certification Program. The more I learn about accessibility and disability inclusion, the more I appreciate the seriousness of our work. We are no longer digital pioneers exploring and building a brave new World Wide Web. We are building the world we live in. We hold lives in trust.

In this article, I explore accessibility in the digital world using an analogous example of accessibility in the built environment. With the support of this scaffolding, we can examine the gaps in our profession that lead to inaccessible technology and disability discrimination and propose ways to build competency and capacity for an accessible and inclusive digital world.

DEFINING THE ROLE OF ACCESSIBLE TECHNOLOGY

During the COVID-19 pandemic, the idea of “critical systems” moved from the margins to the mainstream, as so many systems have become critical to society and well-being. Those who live on the near side of the digital divide have enjoyed a level of continuity in access to employment, health care, education, goods, and services, thanks to Internet-enabled devices like computers and smartphones.

However, access to technology does not guarantee access to Internet-based programs and services. Researchers at Carnegie Mellon University reviewing

Digital Object Identifier 10.1109/MC.2021.3122476

Date of current version: 12 January 2022



Twitter data to identify accessibility issues surfaced by the rapid transition to the digital world found people with disabilities reporting issues accessing products, education, and public health information.² Blind and low vision participants in a 2020 American Foundation for the Blind study reported issues accessing COVID-19 information as well as more general challenges using technology for health information, transportation, shopping, employment, socializing, education, and voting.³

Technology can help overcome impairments

Humans use technology to overcome impairments. People who have vision impairments use eyeglasses and contact lenses to correct their vision and magnification and other settings to adjust displays. People who have mobility impairments use mobility aids, like wheelchairs and scooters, and operate technology using different inputs, including speech and eye-tracking. People who are blind use text-to-speech software to operate graphical interfaces and consume digital information. People who have learning difficulties also use text-to-speech tools to help with reading and writing. Technology is an essential tool in making opportunities and enabling participation.

Assistive technology and accessibility strategies enable access to the digital world. Screen reader software converts text to speech for people who are blind and people who need reading assistance. Speech recognition enables speech control and input for people who can't operate a mouse, touchscreen, or keyboard. Magnification and other modifications allow people with vision impairments to make necessary adjustments to the display of content and controls. Like an accessible building entrance, accessible technology opens doors to opportunities and participation.

But assistive technology and accessibility strategies only work when the features that comprise the

digital world are built to accessibility standards. The systems and software, the websites, applications, and apps that we use to get information, access programs and services, communicate, and connect must include accessibility features, and the content they provide must be accessible. Otherwise, people who have accessibility needs may be locked out.

DURING THE COVID-19 PANDEMIC, THE IDEA OF "CRITICAL SYSTEMS" MOVED FROM THE MARGINS TO THE MAINSTREAM, AS SO MANY SYSTEMS HAVE BECOME CRITICAL TO SOCIETY AND WELL-BEING.

Technology barriers can cause exclusion

We are all responsible for preventing exclusion, in our lives and in our work. Under the social model, disability is "the socially created disadvantage and marginalization experienced by people who have (or are perceived to have) 'impairments'."⁴ This view is supported by disability and equality laws, policies, and legislation around the world, including the ADA, civil rights law that prohibits discrimination on the basis of disability. With this view, everyone in society has a role to play in eliminating existing barriers and preventing new ones.

Disability discrimination occurs when a person with an impairment is treated differently and less favorably due to their disability. Architectural and communication barriers cause disability discrimination, for example, when a customer with a mobility impairment is unable to enter a shop or restaurant due to steps, or a citizen with hearing loss cannot access public health video updates due to absence of captions, or a student who uses a screen reader can't complete assignments due to inaccessible course materials.

Laws prohibit disability discrimination in the physical and digital world. Title III of the ADA requires public accommodations—places that offer goods and services to the general public—to “afford goods, services, facilities, privileges, advantages, and accommodations to an individual with a disability in the most integrated setting appropriate to the needs of the individual,” and discrimination is “a failure to design and construct facilities ... that are readily accessible to and usable by individuals with disabilities.”⁵

PEOPLE WHO NEED INVERTED COLORS CAN CHANGE THE DISPLAY SETTINGS ON THEIR DEVICE, AND WELL-BUILT WEBSITES AND APPS WILL ADAPT TO THEIR SETTINGS.

On the question of whether a website is considered a place of public accommodations, the U.S. Department of Justice issued a letter stating, “The Department first articulated its interpretation that the ADA applies to public accommodations’ websites over 20 years ago.”⁶ Another place to look for evidence of disability discrimination through inaccessible technology is the rising number of ADA Title III lawsuits related to web accessibility.⁷

The ADA requires standards compliance for new construction and alterations and removal of architectural and communication barriers in existing facilities. If we apply that guidance to the digital world, this means that new technology must comply with accessibility guidelines and existing technology must be remediated to remove barriers.

Example: Access routes and entrances

For built facilities, accessibility standards have extensive requirements and guidance for accessible access routes and entrances—and rightly so! What good are accessible programs and services if people can’t get in the door? With technology, we can think of sign-ups and logins as a digital form of access routes and entrances. Signing up for a service is the access route, a path to participation; logging in is getting through the front door. A designed-in barrier could be a submit button that only works with a pointing device and

not a keyboard. Another barrier could be a mandatory data field without a visible or accessible label. Verification tests, like CAPTCHA, that require users to transcribe characters, images, or audio can be impossible to complete. These barriers mean some people will be unable to complete and submit the form with the required data, effectively barring access to the services on the basis of disability.

The relatively plastic digital world can be more amenable to accessibility than the hard edges of the physical world. Rather than requiring a range of accessibility features to ensure the access route and entrance works for everyone, digital resources built on accessibility standards can adapt to meet individual accessibility needs. For example, some people with visual impairments see better with inverted colors—light colors on a dark background. Business owners do not need to provide an inverted color option on their website to meet their nondiscrimination obligations. People who need inverted colors can change the display settings on their device, and well-built websites and apps will adapt to their settings.

DEFINING THE ACCESSIBILITY ROLE OF DESIGNERS AND BUILDERS

In the built environment, responsibility for ADA compliance includes entities offering programs and services as well as everyone involved in the design and construction of places of public accommodation, including architects, civil engineers, interior designers, consultants, construction managers, general contractors, and subcontractors.⁸

When we apply this principle to design and construction roles in the digital environment, that means software engineers, product owners, designers, developers, and others are responsible for accessibility compliance. Are the designers and builders of the digital world adequately prepared to meet that responsibility?

Professionalism can help meet accessibility responsibilities

The built environment has a defined regulatory framework that reflects its impact on critical factors like life safety and accessibility. With accessibility, there are laws and policies that establish requirements and

standards and guidelines that define accessibility features. Education programs cover accessibility topics and accessibility is a requirement in program accreditation. Architects must have a license to practice and must meet licensing requirements, including education from an accredited program. Programs are synchronized to support uniform preparation through education, experience, and examination. With this framework in place, society can have some assurance that built features of the physical world will provide a level of accessibility.

A prerequisite for professionalism is education programs that are defined by accreditation requirements. For an architectural program to be accredited by the National Architectural Accrediting Board, for example, graduates must demonstrate the “[a]bility to design sites, facilities, and systems that are responsive to relevant codes and regulations, and include the principles of life-safety and accessibility standards.” Acknowledging the challenges of balancing multiple and sometimes conflicting priorities, they must also demonstrate the “[a]bility to make design decisions within a complex architectural project while demonstrating broad integration and consideration of environmental stewardship, technical documentation, accessibility, site conditions, life safety, environmental systems, structural systems, and building envelope systems and assemblies.”⁹ Requirements and regulations don’t guarantee thoughtful consideration of accessibility and inclusive design. But including accessibility among accreditation requirements for architecture programs helps ensure tomorrow’s architects are aware of accessibility, recognize their professional obligations, and have the tools they need to include accessibility in their designs.

Technology professionals are not prepared for accessibility

Technology program accreditation requirements do not include accessibility. In the Accreditation Board for Engineering and Technology Criteria for Accrediting Engineering Programs, the software engineering curriculum requirements reference life-safety topics, including security, verification, and validation, but not accessibility. “The curriculum must include computing fundamentals, software design and construction, requirements analysis, security, verification, and

validation; software engineering processes and tools appropriate for the development of complex software systems; and discrete mathematics, probability, and statistics, with applications appropriate to software engineering.” The only place accessibility is mentioned is a list of example constraints, including “accessibility, aesthetics, codes, constructability, cost, ergonomics, extensibility, functionality, interoperability, legal considerations, maintainability, manufacturability, marketability, policy, regulations, schedule, standards, sustainability, or usability.”¹⁰

Without accreditation requirements for digital accessibility, efforts to incorporate the topic into education programs are limited and ad hoc. In their 2018 survey of computing and information science faculty at 318 institutions in the United States, Shinohara et al. found that most of the faculty who teach accessibility (375 out of 1857 responses, or 20%) teach it once a year in a class or two. Most respondents reported a main challenge to teaching accessibility was that it was “not a core part of the curriculum.”¹¹

Consequently, competence in web accessibility is not originating in formal education programs. In the Web Accessibility in Mind (WebAIM) Survey of Web Accessibility Practitioners, formal schooling (12.5%) came in last for ways practitioners learned about web accessibility. Most reported informal and unstructured learning experiences, including online resources (91.3%), on-the-job training or experiences (83.4%), and collaboration with peers or colleagues (81.1%).¹² In their 2020 survey of technology professionals, Patel et al. found that that 44% were either not very or not at all familiar with accessibility guidelines, and 63% were either not very or not at all familiar with accessibility laws. “Some participants were aware of ADA requirements for construction but did not know how those rules applied to software development.”¹³

Given the lack of formal structure and preparation, it’s not surprising that accessibility defects are commonplace in the digital world. The 2021 WebAIM analysis of 1 million home pages found an average of 51.4 automatically detectable accessibility errors per page, including low-contrast text, missing alternative text for images, and missing form input labels.¹⁴ Using sufficient color contrast, providing alternative text for images, and programmatically labeling form inputs are basic accessibility features that are readily achievable

and, like other marks of quality, should be core practice for any competent professional.

BUILDING A FRAMEWORK FOR ACCESSIBILITY PROFESSIONALISM

How can we ensure accessibility competency, such that design and engineering professionals are prepared to build accessible digital resources? While the digital world lacks a formal regulatory framework, there are solid building blocks to support accessibility professionalism:

- › *Digital accessibility standards:* The Web Content Accessibility Guidelines (WCAG, w3.org/TR/WCAG21/) is an international standard developed by the Worldwide Web Consortium. First published in 1999 and published as an International Organization for Standardization standard (ISO/IEC 40500) in 2012, the standards provide the specifications and requirements for supporting accessibility in digital resources. WCAG is the measure used to assess compliance with nondiscrimination laws and policies. Unfortunately, the standards are not widely known among technology professionals.
- › *Specialist certification programs:* The International Association of Accessibility Professionals (accessibilityassociation.org) offers professional certification programs on accessibility topics in digital and built environments. In the United States, the Department of Homeland Security Trusted Tester program (dhs.gov/508-training), and the ADA Coordinator Training Certification Program (adacoordinator.org) provide training and certification. While these types of certification programs help build specialist expertise, they do not address the urgent need for core competency in accessibility across technology design and engineering professions.
- › *Accessibility teaching:* To address shortcomings in accessibility education efforts, Teach Access (teachaccess.org) is working to advance accessibility in higher education through curriculum development, mentoring, advocacy and outreach, and industry demand for accessibility skills. Teaching Accessibility in the Digital Skillset (teachingaccessibility.ac.uk) is researching accessibility pedagogy, defining and resourcing the field of accessibility education. The Web Accessibility Initiative (WAI, w3.org/wai) provides resources for teaching accessibility. This work toward building solid accessibility teaching pedagogy and effective teaching resources is foundational to any effort to incorporate accessibility into core curricula in different technology disciplines.

Building blocks are just that—blocks to build on. On their own, they provide some support. Together, they become more substantial. Should we continue to disregard accessibility standards, delegate responsibility to specialists, and provide cursory coverage (at best) in the curriculum? Or can we combine the accessibility professionalism building blocks with those for other critical factors, such as cybersecurity, and build professionalism that recognizes the impact of our work on the digital world? Owning professional responsibility for digital accessibility and disability inclusion could be an impetus for establishing a regulatory framework that supports competence, professionalism, and accountability from technology professionals so that we *are* prepared to hold lives in trust. So that we do no harm. 🙏

ACKNOWLEDGMENTS

I am grateful to Valerie Fletcher, executive director of the Institute for Human Centered Design, and David Sloan, Research and Strategy lead at TPGi, for helping me frame and shape the argument and to Sarah Lewthwaite and Andy Coverdale, my colleagues on the Teaching Accessibility in the Digital Skill Set research team, for ongoing insights and support. This work was supported by U.K. Research and Innovation Future Leaders Fellowship MR/S01571X/1.

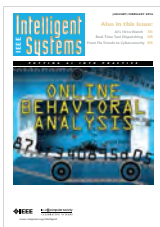
REFERENCES

1. P. Laplante, "A brief history of software professionalism and the way forward," *Computer*, vol. 53, no. 9, pp. 97–100, 2020, doi: 10.1109/MC.2020.3004017.
2. C. Gleason et al., "Disability and the COVID-19 pandemic: Using Twitter to understand accessibility during rapid societal transition," in *Proc. ACM SIGACCESS Conf. Comput. Accessibility, ASSETS '20*, 2020, pp. 1–14, doi: 10.1145/3373625.3417023.

3. L. P. Rosenblum et al., "Flatten inaccessibility: Impact of COVID-19 on adults who are blind or have low vision in the United States," American Foundation for the Blind, Arlington, VA, USA, 2020. [Online]. Available: <https://www.afb.org/research-and-initiatives/flatten-inaccessibility-survey>
4. A. Lawson and A. E. Beckett, "The social and human rights models of disability: Towards a complementarity thesis," *Int. J. Human Rights*, vol. 25, no. 2, pp. 348–379, 2021, doi: 10.1080/13642987.2020.1783533.
5. "Americans with disabilities act title III regulations," ADA.gov, 2017. [Online]. Available: https://www.ada.gov/regs2010/titleIII_2010/titleIII_2010_regulations.htm
6. L. Feingold, "Department of justice affirms ADA's coverage of websites," Law Office of Lainey Feingold, 2018. [Online]. Available: <https://www.lflegal.com/2018/09/doj-cut/>
7. K. M. Launey and M. N. Vu, "Federal website accessibility lawsuits increased in 2020 despite mid-year pandemic lull," Seyfarth, 2021. [Online]. Available: <https://www.adatitleiii.com/2021/04/federal-website-accessibility-lawsuits-increased-in-2020-despite-mid-year-pandemic-lull/>
8. J. A. Weil and A. D. Platt, "Disability, accessibility and liability: What an architect should know," AIA Trust, 2014. [Online]. Available: <https://theaiatrust.com/resource/disability-accessibility-liability-what-an-architect-should-know/>
9. "Conditions for accreditation," National Architectural Accrediting Board, Washington, DC, USA, 2014. [Online]. Available: https://www.naab.org/wp-content/uploads/01_Final-Approved-2014-NAAB-Conditions-for-Accreditation-2.pdf
10. "Criteria for accrediting engineering programs, 2021-2022," ABET, Baltimore, MD, USA, 2021. [Online]. Available: <https://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-engineering-programs-2021-2022/>
11. K. Shinohara, S. Kawas, A. J. Ko, and R. E. Ladner, "Who teaches accessibility?: A survey of U.S. computing faculty," in *Proc. ACM SIGCSE Tech. Symp. Comput. Sci. Educ.*, SIGCSE '18, 2018, pp. 197–202, doi: 10.1145/3159450.3159484.
12. "Survey of web accessibility practitioners #3 results," WebAIM, 2021. [Online]. Available: <https://webaim.org/projects/practitionersurvey3/>
13. R. Patel, P. Breton, C. M. Baker, Y. N. El-Glaly, and K. Shinohara, "Why software is not accessible: Technology Professionals' perspectives and challenges," in *Proc. ACM CHI Conf. Human Factors Comput. Syst.*, CHI EA '20, 2020, pp. 1–9, doi: 10.1145/3334480.3383103.
14. "The WebAIM million: An annual accessibility analysis of the top 1,000,000 home pages," WebAIM, 2021. [Online]. Available: <https://webaim.org/projects/million/>

SARAH HORTON is a member of the research team working on the Teaching Accessibility in the Digital Skill Set at the University of Southampton, Southampton, SO17 1BJ, U.K. Contact her at s.e.horton@soton.ac.uk.

stay on the **Cutting Edge** of Artificial Intelligence



IEEE Intelligent Systems provides peer-reviewed, cutting-edge articles on the theory and applications of systems that perceive, reason, learn, and act intelligently.

The #1 AI Magazine **Intelligent Systems**
www.computer.org/intelligent IEEE

100% ONLINE MASTER OF INFORMATION TECHNOLOGY

ANALYTICS & BUSINESS INTELLIGENCE
BIG DATA
BUSINESS INFORMATION SYSTEMS
CYBERSECURITY
CYBERSECURITY MANAGEMENT
CYBERSECURITY POLICY
DECISION SUPPORT SYSTEMS
HEALTH INFORMATION TECHNOLOGY
INNOVATION IN AI/ML
NETWORKING
SOFTWARE DEVELOPMENT
SOFTWARE ENGINEERING



MASTER OF
INFORMATION TECHNOLOGY
VIRGINIA TECH.

VTMIT.VT.EDU

Computing in Science & Engineering

The computational and data-centric problems faced by scientists and engineers transcend disciplines. There is a need to share knowledge of algorithms, software, and architectures, and to transmit lessons-learned to a broad scientific audience. *Computing in Science & Engineering (CiSE)* is a cross-disciplinary, international publication that meets this need by presenting contributions of high interest and educational value from a variety of fields, including physics, biology, chemistry, and astronomy. *CiSE* emphasizes innovative applications in cutting-edge techniques. *CiSE* publishes peer-reviewed research articles, as well as departments spanning news and analyses, topical reviews, tutorials, case studies, and more.

Read *CiSE* today! www.computer.org/cise



Conference Calendar

IEEE Computer Society conferences are valuable forums for learning on broad and dynamically shifting topics from within the computing profession. With over 200 conferences featuring leading experts and thought leaders, we have an event that is right for you. Questions? Contact conferences@computer.org.

MAY

4 May

- ICCPS (ACM/IEEE Int'l Conf. on Cyber-Physical Systems), Milano, Italy
- RTAS (IEEE Real-Time and Embedded Technology and Applications Symposium), Milano, Italy

9 May

- ICDE (IEEE Int'l Conf. on Data Eng.), virtual

15 May

- FCCM (IEEE Int'l Symposium on Field-Programmable Custom Computing Machines), New York, USA

16 May

- IC FEC (IEEE Int'l Conf. on Fog and Edge Computing), Messina, Italy

17 May

- ISORC (Int'l Symposium On Real-Time Distributed Computing), Västerås, Sweden

18 May

- ISCV (Int'l Conf. on Intelligent Systems and Computer Vision), Fez, Morocco
- ISMVL (IEEE Int'l Symposium on Multiple-Valued Logic), Dallas, USA

19 May

- AQTR (Int'l Conf. on Automation, Quality and Testing,

Robotics), Cluj-Napoca, Romania

- SELSE (IEEE Workshop on Silicon Errors in Logic – System Effects), virtual

21 May

- ICSE (IEEE/ACM Int'l Conf. on Software Eng.), Pittsburgh, USA

22 May

- ISPASS (IEEE Int'l Symposium on Performance Analysis of Systems and Software), Singapore
- SP (IEEE Symposium on Security and Privacy), San Francisco, USA

23 May

- ETS (IEEE European Test Symposium), Barcelona, Spain
- SEAMS (Int'l Symposium on Software Eng. for Adaptive and Self-Managing Systems), Pittsburgh, USA

25 May

- SERA (IEEE/ACIS Int'l Conf. on Software Eng., Management and Applications), Las Vegas, USA

30 May

- DCOSS (Int'l Conf. on Distributed Computing in Sensor Systems), Los Angeles, USA
- IPDPS (IEEE Int'l Parallel & Distributed Processing Symposium), Lyon, France

JUNE

6 June

- EuroS&P (IEEE European Symposium on Security and Privacy), Genoa, Italy
- MDM (IEEE Int'l Conf. on Mobile Data Management), Paphos, Cyprus

11 June

- ISCA (ACM/IEEE Int'l Symposium on Computer Architecture), New York, USA

14 June

- WoWMoM (IEEE Int'l Symposium on a World of Wireless, Mobile and Multimedia Networks), Belfast, UK

19 June

- CVPR (IEEE/CVF Conf. on Computer Vision and Pattern Recognition), New Orleans, USA

25 June

- CSCLOUD (IEEE Int'l Conf. on Cyber Security and Cloud Computing), Xi'an, China

26 June

- ICIS (IEEE/ACIS Int'l Conf. on Computer and Information Science), Zhuhai, China

27 June

- COMPSAC (IEEE Computers, Software, and Applications Conf.), Torino, Italy
- DSN (IEEE/IFIP Int'l Conf. on



Dependable Systems and Networks), Baltimore, USA

- HOST (IEEE Int'l Symposium on Hardware Oriented Security and Trust), McLean, Virginia, USA

JULY

1 July

- ICALT (IEEE Int'l Conf. on Advanced Learning Technologies), Bucharest, Romania

6 July

- ISVLSI (IEEE Computer Society Symposium on VLSI), Nicosia, Cyprus

10 July

- ICDCS (IEEE Int'l Conf. on Distributed Computing Systems), Bologna, Italy

11 July

- ICME (IEEE Int'l Conf. on Multimedia and Expo), Taipei, Taiwan

21 July

- CBMS (IEEE Int'l Symposium on Computer-Based Medical Systems), Shenzhen, China

AUGUST

1 August

- ICCP (Int'l Conf. on Computational Photography), Pasadena, USA

2 August

- MIPR (IEEE Int'l Conf. on Multimedia Information Processing and Retrieval), virtual

4 August

- BCD (IEEE/ACIS Int'l Conf. on Big Data, Cloud Computing,

and Data Science Eng.), Danang, Vietnam

7 August

- CSF (IEEE Computer Security Foundations Symposium), Haifa, Israel

9 August

- IRI (IEEE Int'l Conf. on Information Reuse and Integration for Data Science), virtual

15 August

- RE (IEEE Int'l Requirements Eng. Conf.), Melbourne, Australia

SEPTEMBER

6 September

- CLUSTER (IEEE Int'l Conf. on Cluster Computing), Heidelberg, Germany

12 September

- ARITH (IEEE Symposium on Computer Arithmetic), virtual

18 September

- QCE (IEEE Quantum Week), Broomfield, Colorado, USA

19 September

- AI4I (Int'l Conf. on Artificial Intelligence for Industries), Laguna Hills, USA
- AIKE (IEEE Int'l Conf. on Artificial Intelligence and Knowledge Eng.), Laguna Hills, USA
- ESEM (ACM/IEEE Int'l Symposium on Empirical Software Eng. and Measurement), Helsinki, Finland
- TransAI (Int'l Conf. on Transdisciplinary AI), Laguna Hills, USA

26 September

- ASE (IEEE/ACM Int'l Conf. on Automated Software Eng.), Ann Arbor, USA

OCTOBER

3 October

- ICSME (IEEE Int'l Conf. on Software Maintenance and Evolution), Limassol, Cyprus
- NAS (IEEE Int'l Conf. on Networking, Architecture and Storage), Philadelphia, USA

4 October

- IMET (Int'l Conf. on Interactive Media, Smart Systems and Emerging Technologies), Limassol, Cyprus

16 October

- MODELS (ACM/IEEE Int'l Conf. on Model Driven Eng. Languages and Systems), Montreal, Canada
- VIS (IEEE Visualization and Visual Analytics), Oklahoma City, USA

Learn more
about IEEE
Computer Society
conferences

computer.org/conferences

Evolving Career Opportunities Need Your Skills

Explore new options—upload your resume today

www.computer.org/jobs

Changes in the marketplace shift demands for vital skills and talent. The **IEEE Computer Society Jobs Board** is a valuable resource tool to keep job seekers up to date on the dynamic career opportunities offered by employers.

Take advantage of these special resources for job seekers:



JOB ALERTS



TEMPLATES



WEBINARS



CAREER
ADVICE



RESUMES VIEWED
BY TOP EMPLOYERS

No matter what your career level, the IEEE Computer Society Jobs Board keeps you connected to workplace trends and exciting career prospects.



IEEE
COMPUTER
SOCIETY



IEEE