



Association for  
Computing Machinery

*Advancing Computing as a Science & Profession*

## NEWS RELEASE

**Contact:** Jim Ormond  
212-626-0505  
[ormond@acm.org](mailto:ormond@acm.org)

### **SPLASH 2018 BRINGS TOGETHER “BEST OF THE BEST” IN PROGRAMMING TECHNOLOGIES**

#### **Leaders from Industry and Academia to Explore Intersection of Programming Languages and Software Engineering**

**New York, NY, November 5, 2018**– The Association for Computing Machinery’s Special Interest Group on Programming Languages (SIGPLAN) announced that it will hold its annual [Conference on Systems, Programming, Languages and Application: Software for Humanity](#) (SPLASH) from November 4 – 9, 2018, in Boston, Massachusetts. The event brings together educators, researchers and industry to showcase the latest advances in software construction, delivery and engineering. SPLASH 2018 will be held in conjunction with four other conferences: OOPSLA, Onward!, GPCE and SLE.

“With four co-located conferences, one symposium, 14 workshops, and the deep involvement of Boston’s world-class technology community, SPLASH represents the largest academic gathering of researchers, practitioners and students interested in programming technologies,” said General Chair Jan Vitek. “This year, we’re excited to offer extended talks and a variety of targeted tracks as we continue to build upon the offerings of this best-in-class event.”

#### **2018 ACM SPLASH Highlights**

##### **Keynotes**

##### **“In Defense of ‘Little Code’”**

*Kathi Fisler, Brown University*

In today’s computing world, Big Code presents a wealth of opportunity to generate data to study how computer scientists program. Fisler’s talk describes studies and results on how students choose program structure, how language choice interacts with program structure, what misconceptions students have about semantics, and how students test programs.

##### **“Reasoning about Security of Amazon Web Services”**

*Byron Cook, Amazon Web Services*

In this talk, the founder and leader of Amazon’s Automated Reasoning Group (ARG) will discuss the development and use of formal verification tools within Amazon Web Services (AWS) to increase the security assurance of its cloud infrastructure and to help customers secure themselves. He will also discuss some remaining challenges, which could inspire future research in the HCI and cybersecurity communities.

### **“Beauty Is the Promise of Happiness”**

*Jenny Quillien, Embodied Making Institute*

In much scientific work the intuitive recognition of beauty is still seen as the highest integrative level of understanding. Quillien’s talk explores the definition of beauty, outlining some its key characteristics, and ultimately proposes that striving for beauty should be a compass for those in the sciences and the arts.

### **“A New Modularity for Software”**

*Daniel Jackson, MIT*

Jackson’s current focus is on new approaches to software design for improved usability and reduced development cost; new programming paradigms; security-by-design for cyberphysical systems and web apps. He has general interests in lightweight formal methods (such as Alloy) and the role of design thinking in software.

### **“How a Computer Can Write a Poem and Make It Sound Like an Angry Type Theorist, or Proving Theorems and Seeing Cats”**

*Richard P. Gabriel, Dream Songs, Inc. & HPI*

Gabriel will discuss how a program he wrote for the Defense Advanced Research Projects Agency (DARPA) to help thwart terrorist plots, was able to write a haiku. During his presentation he will discuss how and why programming is stranger than most people think, and why artificial intelligence needs to be a blend of proving theorems (in deference to John McCarthy) and seeing cats (in deference to Andrew Ng).

### **“50 Years of Programming and Language Design”**

*Guy L. Steele Jr., Oracle Labs*

Steele is a software architect at Oracle Labs. He has also taught at Carnegie Mellon University and worked for Tartan Laboratories and Sun Microsystem Laboratories. He is author or co-author of reference manuals or specifications for Scheme, Common Lisp, C, High Performance Fortran, Java and JavaScript.

### **Featured Splash-I Talks (partial list)**

#### **“Peering Behind the Turing Mirror”**

*Ben L. Titzer, Google*

Until now, CPU designers and computer scientists have assumed “Vegas” rules at the hardware level: what happens in speculation stays in speculation. Yet in the wake of the Spectre and Meltdown attacks, it is clear a new, massive class of security vulnerabilities lies at the microarchitectural level. Many computer scientists have assumed that virtualization through emulation made the worlds below the Turing machine undetectable, hidden behind a mathematically perfect mirror. This talk will explore how computer scientists have now learned to see through that mirror, into the very bizarre and alien world of microarchitectures, illuminating a tiny world of astounding complexity.

#### **“Developing Opal, an App for Cancer Patients, as a Computer Scientist and Cancer Patient”**

*Laurie Hendren, McGill University*

Being a cancer patient is a complex, frustrating, scary and intimidating experience, especially for those with complex or prolonged treatments. One way of dealing with cancer is to determine what can be done to improve the patient experience, and to work with a great team from within the health care system in order to make those improvements happen. This talk is about Hendren’s experience as a

cancer patient, and the work she has done in creating Opal, a patient-centered app that supports automated and personalized data, notifications, appointment schedule, patient-reported outcome questionnaires, just-in-time educational material, and much more.

### **“Software Is Eating the World, but ML Is Going to Eat Software”**

*Joe Pamer, Facebook*

“Democratizing ML” is a hot topic these days—particularly in industry. Unfortunately, while machine learning has the potential to fundamentally improve how software is constructed, opportunities to leverage machine learning to improve more conventional developer tools (languages, compilers, and integrated development environments (IDE’s) for example) have largely gone untapped. Pamer’s talk will detail the work of his team to improve developer efficiency and resource utilization at Facebook. Pamer will discuss an initiative to update the Hack programming language to support probabilistic programming techniques, as well as an effort to develop a new suite of AI-driven developer tools. He will discuss the lessons his team has learned along the way, as well as future opportunities they see to optimize or auto-tune other common pieces of developer architecture.

### **Conference and Workshops Invited Speakers (partial list)**

#### **“BEAM: A Virtual Machine for Handling Millions of Messages per Second”**

*Erik Stenman, Aeternity*

BEAM, the virtual machine for Erlang, was built by Ericsson to handle internet traffic. Today Erlang is used in many high volume settings like gaming, messaging and financial services. For example, WhatsApp uses Erlang to handle close to 100 billion messages per day. In his talk, Stenman will look at the BEAM in detail to see how it is implemented. He will also look at the motivation behind the Erlang design and how it has affected the virtual machine.

#### **“A New Approach for Software Correctness and Reliability”**

*Martin Rinard, MIT*

In this talk Rinard will consider two very different approaches to solving correctness and security problems, failure-oblivious computing and domain-specific languages. He will discuss how these approaches (as well as others) interact with the cognitive limitations and available technical skills of the human population of software developers that currently must be part of any solution for it to be successful. Rinard will conclude by outlining a new approach that, by deploying automated programming language technology in an appropriately targeted way, may interact more productively with the characteristics of the developer population as a whole.

#### **About SIGPLAN**

[SIGPLAN](#) is a Special Interest Group of ACM that focuses on Programming Languages. In particular, SIGPLAN explores the design, implementation, theory, and efficient use of programming languages and associated tools. Its members are programming language users, developers, implementers, theoreticians, researchers and educators.

#### **About ACM**

[ACM, the Association for Computing Machinery](#), is the world’s largest educational and scientific computing society, uniting computing educators, researchers and professionals to inspire dialogue, share resources and address the field’s challenges. ACM strengthens the computing profession’s collective voice through strong leadership, promotion of the highest standards, and recognition of technical excellence. ACM supports the professional growth of its members by providing opportunities for lifelong learning, career development, and professional networking.

###