Andrew A. Chien

DOI:10.1145/3243136

Computer Architecture: Disruption from Above

I RECENTLY ATTENDED the 45th ACM/IEEE International Symposium on Computer Architecture (ISCA) in Los Angeles, and was struck by the atmosphere of dramatic change in the field. First, a bit of perspective: For the last 15 years, computer architecture, and as a consequence computing as a whole, has been disrupted from below with the end of Dennard Scaling,¹ producing a dramatic slowing in the growth of clock rate and single-thread performance, a shift to multicore, and the rise of throughput engines such as GPUs. In addition, and in particular on mobile platforms, we have seen the rise of heavily customized architectures in mobile and embedded devices. These changes, precipitated by device and circuit-level effects, have rippled through the software stack compelling large-scale software rewrites, new compiler techniques and programming approaches, and the pervasive adoption of parallelism as a fundamental basis of performance. In the past three years, we have seen the effective end of Moore's Law scaling, with per transistor prices flat or increasing with recent technology nodes,² and the slowing rate of advance to new process nodes at Intel (and across the industry).³ Effectively, these changes have rung the death knell for these two foundational scaling trends that fueled computing's rapid advance for over 50 years. How to cope with these disruptions from below have been a core theme at ISCA for the past 15 years.

At ISCA 2018, a clear theme emerged that I'll call "Disruption from Above" (DfA).

DfA Vector #1: A primary limitation of customization and accel-

erators has been the lack of a broad, powerful computing model that was amenable to dramatic performance improvements. The broad computing trend of data-enabled machine learning, specifically where trained deep neural networks (DNNs) replace complex software, whimsically called "Software 2.0"⁴ has created a new opportunity. Widespread adoption of this new computing model—of remarkable generality that seems to grow by the month—and the opportunity for dramatically simpler hardware implementation (neurons and structured parallel networks, *not* cores) that are efficiently scalable has triggered a tornado of research activity. A terrific perspective on the radical changes to the field of computer architecture were captured in Kunle Olukotun's keynote "Designing Computer Systems for Software 2.0" and throughout the conference (10 papers on neural network computation engines). Software 2.0 (trained DNNs) and Software 2.0 architectures is a new computing stack.

DfA Vector #2: The Spectre and Meltdown security attacks break down isolation between applications and between applications and the operating system.⁵ Spectre and Meltdown are the first attacks that exploit microarchitecture performance features such as speculation that were intended to be functionally invisible. That is, intended not to change the functional behavior of the architecture. Because Spectre and Meltdown exploit the performance visibility of speculative actions to create information side channels, they extend the functional specification of the architecture to include its detailed

performance. This disrupts the basic definition of the instruction set architecture specification. In short, the definition of an instruction set architecture, such as x86, the traditional basis of portability beginning with the IBM 360 in the 1960s, is no longer a sufficient description of architecture portability for security properties. Directly, making strong assurances of application security on a computing system requires detailed performance information. Functional and performance specifications constitute a new definition of the computer architecture interface.

So, after 15 years of disruption from below, the architecture community faces several major disruptions from above! Both Software 2.0 and Spectre/Meltdown represent significant challenges to systems, to the relationship of software and hardware, and to the roles of a variety of industry players. For the research community, opportunities to shape a profoundly different, new age of computer architecture are emerging.

Exciting times!

Andrew A. Chien, EDITOR-IN-CHIEF

Andrew A. Chien is the William Eckhardt Distinguished Service Professor in the Department of Computer Science at the University of Chicago, Director of the CERES Center for Unstoppable Computing, and a Senior Scientist at Argonne National Laboratory.

References

1. Dennard, R.H. et al. Design of ion-implanted MOSFET's with very small physical dimensions. *IEEE J. Solid State Circuits* 9, 5 (Oct. 1974).
2. *Economist*. The end of Moore's Law. (Apr. 19, 2015).
3. Gwennap, L. TSMC 7nm Approaches Intel's Prowess. Microprocessor Report, Apr. 16, 2018.
4. Karpathy, A. Software 2.0; <https://medium.com/@karpathy/software-2-0-a64152b37c35>
5. Meltdown and Spectre, <https://meltdownattack.com/>

Copyright held by author.