It is our pleasure to introduce this year's Top Picks in Computer Architecture. We cochaired the selection committee, which had the formidable task of selecting the top computer architecture papers published in conferences in the previous year. Many excellent papers are published every year, and choosing among them is challenging, not least because of the need to define a “top pick.” To focus the selection process, we encouraged committee members to consider whether a paper exhibited certain attributes—namely, did it

- challenge conventional wisdom,
- establish a new area of research,
- have a high potential for industrial impact,
- serve as the definitive “last word” in an established area, and/or
- encourage the reader to recommend it to students and colleagues?

This list is imperfect and incomplete, yet it captures much of what we were seeking. We were not simply looking for good papers, but rather for papers that are worthy of being among a small number of Top Picks.

Because only a small number of papers can be Top Picks, we followed the precedent set by last year’s cochairs, in which the selection committee identified another set of extremely worthy papers in addition to those that appear in this special issue. These 12 superb papers are recognized as honorable mentions, and we strongly encourage you to read them, too. (See the “Honorable Mentions” sidebar for more information.)
**Honorable Mentions**

<table>
<thead>
<tr>
<th>Paper</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>“Stash: Have Your Scratchpad and Cache It Too” by R. Komuravelli, M. Sinclair, J. Alsop, M. Huzaifa, M. Kotsifakou, P. Srivastava, S. Adve, and V. Adve (ISCA 2015)</td>
<td>Stash is a new type of hybrid memory structure that combines the efficiency of scratchpads with the global address space and coherence of caches. Recently proposed GPU memory models restrict synchronization to hierarchical scopes, which can improve performance but complicate programming. This paper shows how to get the performance benefits of scopes without their complexity.</td>
</tr>
<tr>
<td>“Efficient GPU Synchronization without Scopes: Saying No to Complex Consistency Models” by M. Sinclair, J. Alsop, and S. Adve (MICRO 2015)</td>
<td>This paper explores analog circuits for accelerating approximation-tolerant software, and it describes a compiler to convert this software into a neural model for execution on co-designed circuitry.</td>
</tr>
<tr>
<td>“General-Purpose Code Acceleration with Limited-Precision Analog Computation” by R. St. Amant, A. Yazdanbakhsh, J. Park, B. Thwaites, H. Esmailzadeh, A. Hassibi, L. Ceze, and D. Burger (ISCA 2014)*</td>
<td>Server-side applications that are event-driven, rather than assigning threads to each request, have different performance issues. Bottlenecks are identified and mitigated with memory hierarchy optimizations.</td>
</tr>
<tr>
<td>“Microarchitectural Implications of Event-Driven Server-Side Web Applications” by D. Richins, Y. Zhu, M. Halpern, and V.J. Reddi (MICRO 2015)</td>
<td>SHIM provides accurate and extremely fine-grained performance monitoring of an application thread by using an independent thread to poll the performance counters for the application thread.</td>
</tr>
<tr>
<td>“Computer Performance Microscopy with SHIM” by X. Yang, S. Blackburn, and K. McKinley (ISCA 2015)</td>
<td>GPS uses machine learning techniques to predict the performance of an application on a GPU based only on a single-threaded CPU version of the application. GPS thus helps programmers decide which applications are worth porting to GPUs.</td>
</tr>
<tr>
<td>“A Scalable Processing-in-Memory Accelerator for Parallel Graph Processing” by J. Ahn, S. Hong, S. Yoo, O. Mutlu, and K. Choi (ISCA 2015)</td>
<td>DeSC extends the idea of decoupled access/execute microarchitectures to heterogeneous chips with accelerators, thus mitigating the challenge of providing sufficient access bandwidth to accelerators without greatly complicating the programming effort.</td>
</tr>
<tr>
<td>“Cross-Architecture Performance Prediction (XAPP) Using CPU Code to Predict GPU Performance” by N. Ardalani, C. Lestourgeon, K. Sankaralingam, and X. Zhu (MICRO 2015)</td>
<td>FlashMap is a new interface for solid-state drives (SSDs) that is optimized for memory-mapped SSDs. FlashMap is more efficient because it combines multiple levels of interfaces into a single interface.</td>
</tr>
<tr>
<td>“DeSC: Decoupled Supply-Compute Communication Management for Heterogeneous Architectures” by T. Ham, J. Aragon, and M. Martonosi (MICRO 2015)</td>
<td>DEUCE improves encryption for write-limited phase change memory by writing only the words that have changed.</td>
</tr>
<tr>
<td>“Unified Address Translation for Memory-Mapped SSOS with FlashMap” by J. Huang, A. Badam, M. Qureshi, and K. Schwan (ISCA 2015)</td>
<td>Rumba is a scheme for predicting when an approximate computation has a large error. Rumba exploits machine learning techniques that enable low-cost prediction.</td>
</tr>
<tr>
<td>“DEUCE: Write-Efficient Encryption for Non-Volatile Memories” by V. Young, P. Nair, and M. Qureshi (ASPLOS 2015)</td>
<td>Authenticache provides physically unclonable functions by lowering the power supply voltage and using the random locations of induced bit errors in the cache.</td>
</tr>
<tr>
<td>“Rumba: An Online Quality Management System for Approximate Computing” by D. Khudia, B. Zamirai, M. Samadi, and S. Mahlke (ISCA 2015)</td>
<td>*Because this paper was coauthored by one of last year’s Top Picks guest editors, it was eligible for consideration this year.</td>
</tr>
</tbody>
</table>
colleagues provide a microarchitectural performance analysis of Google’s datacenter workloads at scale running on thousands of machines.

Implementation-Aware Architectures
Three articles focus on hardware implementation issues for microarchitectures for specific platforms. In “MBus: A System Integration Bus for the Modular Microscale Computing Class,” Pat Pannuto and colleagues develop a new bus tailored to meet the requirements of embedded systems with extremely small energy budgets. In “Nonvolatile Processor Architectures: Efficient, Reliable Progress with Unstable Power,” Kaisheng Ma and colleagues explore the design of processors that rely on ambient energy harvesting for their power. In “Exploiting Interposer Technologies to Disintegrate and Reintegrate Multicore Processors,” Ajaykumar Kannan and colleagues explore the design of multicore processors that are connected with silicon interposers, focusing on the interconnection network issues that arise in such platforms.

Hardware/Software Codesign
Three Top Picks articles exploit the codesign of hardware and software for various purposes. In “Achieving One Billion Key-Value Requests per Second on a Single Server,” Sheng Li and colleagues push the limits of key-value store performance by codesigning the hardware and software. In “Unlocking Ordered Parallelism with the Swarm Architecture,” Mark C. Jeffrey and colleagues present a programming model and architectural support for software with ordered irregular parallelism. In “Range Transformations for Fast Virtual Memory,” Jayneel Gandhhi and colleagues codesign hardware and software to improve the performance of virtual memory.

The Committee and the Selection Process
The selection committee consisted of 33 outstanding computer architects from academia and industry (see the “Selection Committee” sidebar). Committee members hailed from six countries, eight companies, and numerous universities. Every committee member was physically present for the meeting that we held on 8 January 2015 in Chicago. Prior to the meeting, the committee members reviewed the 101 papers that were submitted. Each paper received at least five reviews from committee members.

To guide the review and selection processes, we adopted the “Identify the Champion” methodology, which recently has been adopted in several other disciplines.1 With Identify the Champion, the goal is to determine which papers have champions on the committee who...
will strongly advocate for their acceptance. We were more eager to select papers with strong advocates over papers with relatively high numerical scores but no strong enthusiasm (that is, papers that get accepted because they are solid and nobody objects to them). As such, our review forms and our committee meeting were tailored to identify champions instead of ranking submissions according to numerical scores.

The discussion was organized into two rounds. In the first round, the committee discussed the 37 submissions with one or more champions. For each of these papers, the five committee members who reviewed the submission provided their assessments, followed by a full-committee discussion. Selection for either Top Pick or Honorable Mention was decided by a simple majority vote of all committee members. After the first round, the committee had identified seven submissions as Top Picks and 16 as Honorable Mentions. In the second round, the committee revisited the 10 mostly highly ranked Honorable Mentions and elevated four of them to Top Picks selections, resulting in the 11 Top Picks articles and 12 Honorable Mentions described earlier.

We hope that you enjoy reading these articles and that you will explore the original conference versions, as well as the honorable mention papers. We welcome your feedback on this special issue.

Acknowledgments
We owe thanks to the following people who helped make this issue possible. The selection committee performed many reviews and spent a day in Chicago for the committee meeting. Luwa Matthews served as our assistant and administered the submission website. Karin Strauss and Sandhya Dwarkadas handled the papers for which both of us had conflicts of interest. Lieven Eeckhout, as editor in chief of IEEE Micro, provided guidance and entrusted us with this special issue. Theresa McNeil from IEEE helped us arrange the committee meeting. Karin Strauss and Luis Ceze, last year’s Top Picks co-chairs, shared their wisdom and answered numerous questions.

Reference

Milo Martin is a staff software engineer at Google. Contact him at milom@google.com.

Daniel Sorin is the W.H. Gardner Jr. Professor of Electrical and Computer Engineering at Duke University. Contact him at sorin@ee.duke.edu.

IEEE micro
Calls for Papers
IEEE Micro seeks general-interest submissions for publication in upcoming issues. These works should discuss the design, performance, or application of microcomputer and microprocessor systems. Of special interest are articles on performance evaluation and workload characterization. Summaries of work in progress and descriptions of recently completed works are most welcome, as are tutorials. IEEE Micro does not accept previously published material.

Visit our author center (www.computer.org/micro/author.htm) for word, figure, and reference limits. All submissions pass through peer review consistent with other professional-level technical publications, and editing for clarity, readability, and conciseness.

Contact IEEE Micro at micro-ma@computer.org with any questions.

www.computer.org/micro/cfp