

Computer Architecture

Lecture 3b: Memory Systems: Challenges and Opportunities

Prof. Onur Mutlu

ETH Zürich

Fall 2020

24 September 2020

Recall Our Dream (from 2007)

- An embedded device that can perform comprehensive genome analysis in real time (within a minute)
- Still a long ways to go
 - Energy efficiency
 - Performance (latency)
 - Security
 - **Huge memory bottleneck**

Four Key Directions

- Fundamentally **Secure/Reliable/Safe** Architectures
 - Fundamentally **Energy-Efficient** Architectures
 - **Memory-centric** (Data-centric) Architectures
 - Fundamentally **Low-Latency and Predictable** Architectures
-
- Architectures for **AI/ML, Genomics, Medicine, Health**

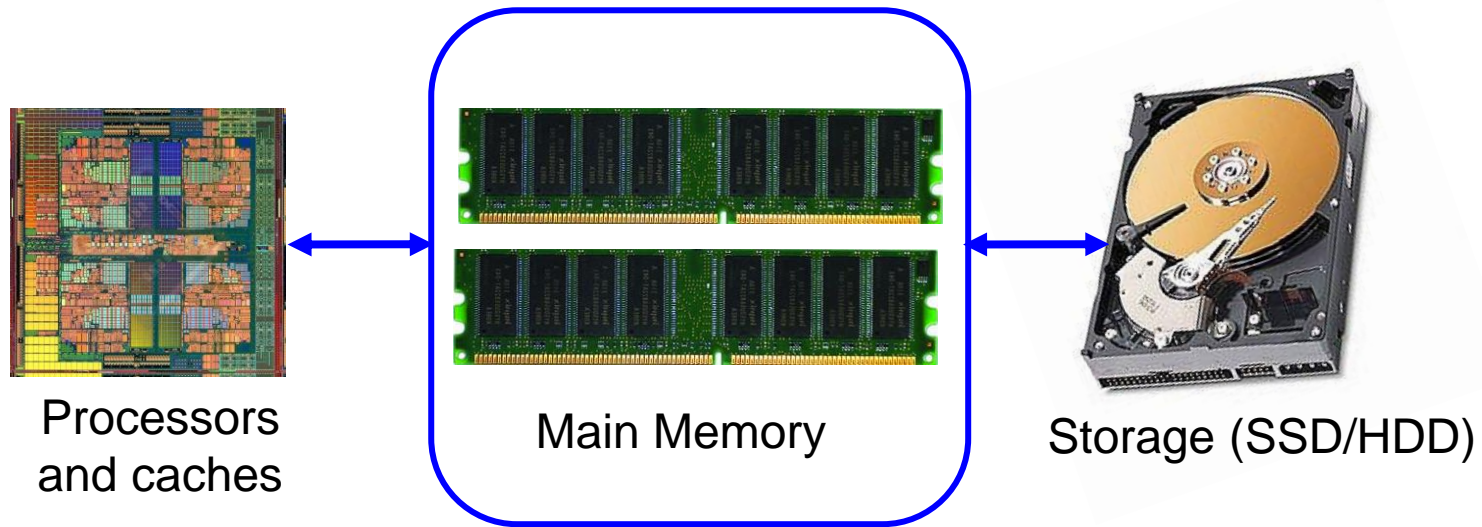
Memory & Storage

Why Is Memory So Important? (Especially Today)

Importance of Main Memory

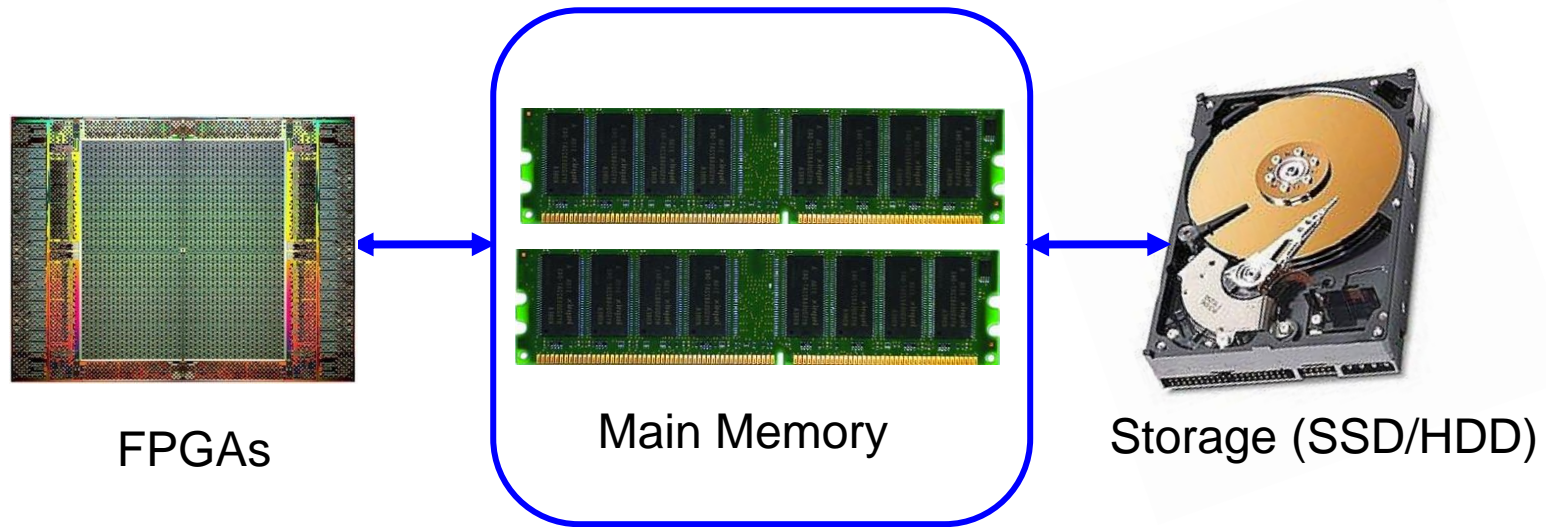
- The Performance Perspective
- The Energy Perspective
- The Scaling/Reliability/Security Perspective
- Trends/Challenges/Opportunities in Main Memory

The Main Memory System



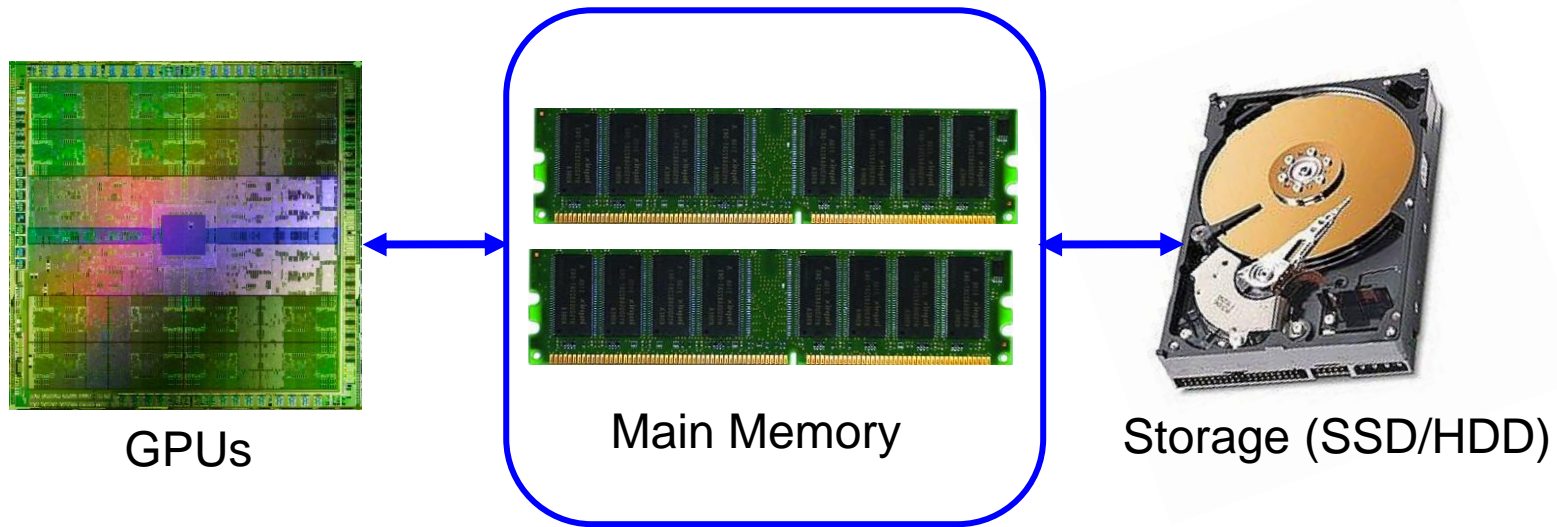
- Main memory is a critical component of all computing systems: server, mobile, embedded, desktop, sensor
- Main memory system must scale (in *size, technology, efficiency, cost, and management algorithms*) to maintain performance growth and technology scaling benefits

The Main Memory System



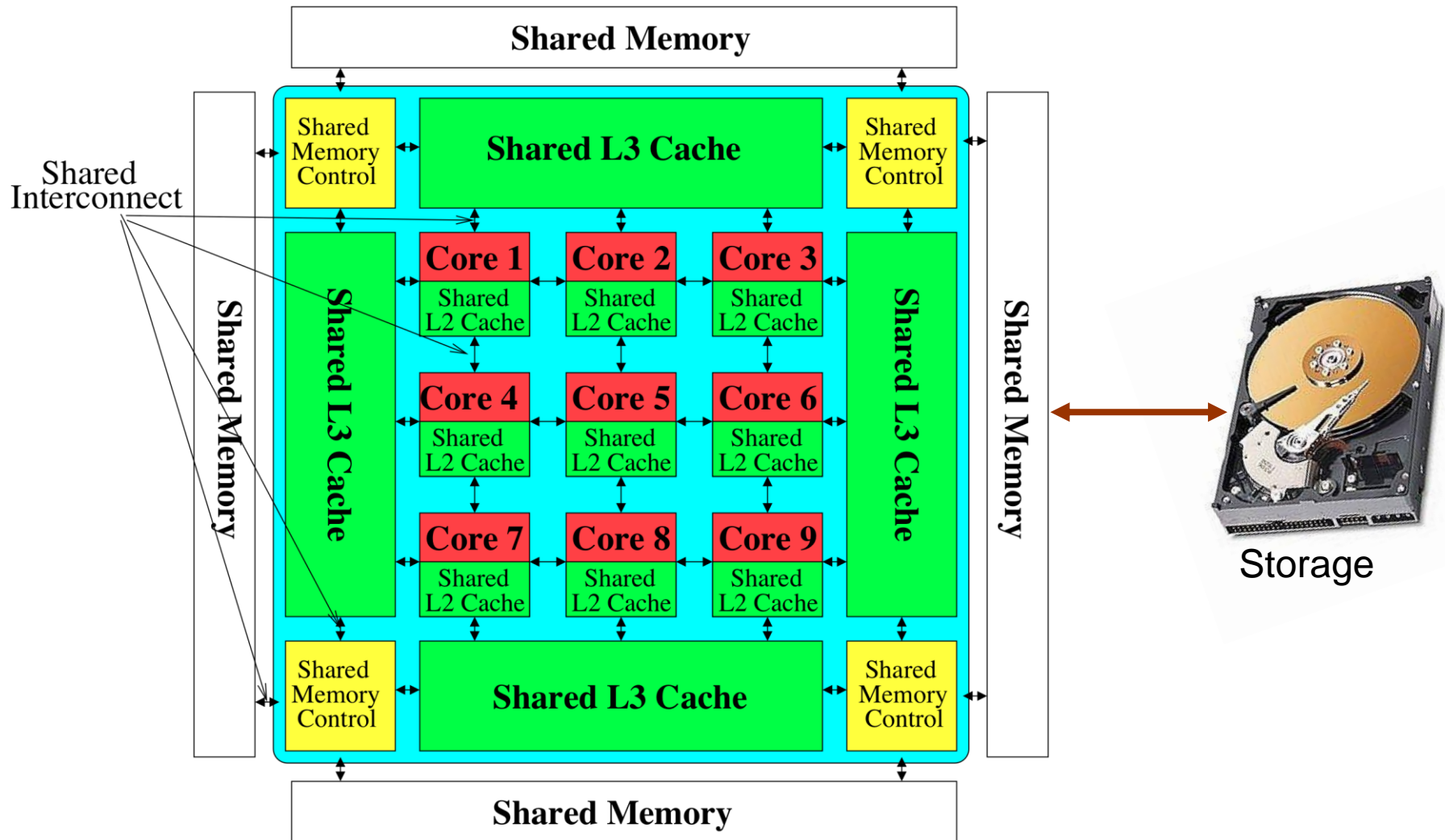
- Main memory is a critical component of all computing systems: server, mobile, embedded, desktop, sensor
- Main memory system must scale (in *size, technology, efficiency, cost, and management algorithms*) to maintain performance growth and technology scaling benefits

The Main Memory System



- Main memory is a critical component of all computing systems: server, mobile, embedded, desktop, sensor
- Main memory system must scale (in *size, technology, efficiency, cost, and management algorithms*) to maintain performance growth and technology scaling benefits

Memory System: A *Shared Resource* View



Most of the system is dedicated to storing and moving data

State of the Main Memory System

- Recent technology, architecture, and application trends
 - lead to new requirements
 - exacerbate old requirements
- DRAM and memory controllers, as we know them today, are (will be) unlikely to satisfy all requirements
- Some emerging non-volatile memory technologies (e.g., PCM) enable new opportunities: memory+storage merging
- We need to rethink the main memory system
 - to fix DRAM issues and enable emerging technologies
 - to satisfy all requirements

Major Trends Affecting Main Memory (I)

- Need for main memory capacity, bandwidth, QoS increasing
- Main memory energy/power is a key system design concern
- DRAM technology scaling is ending

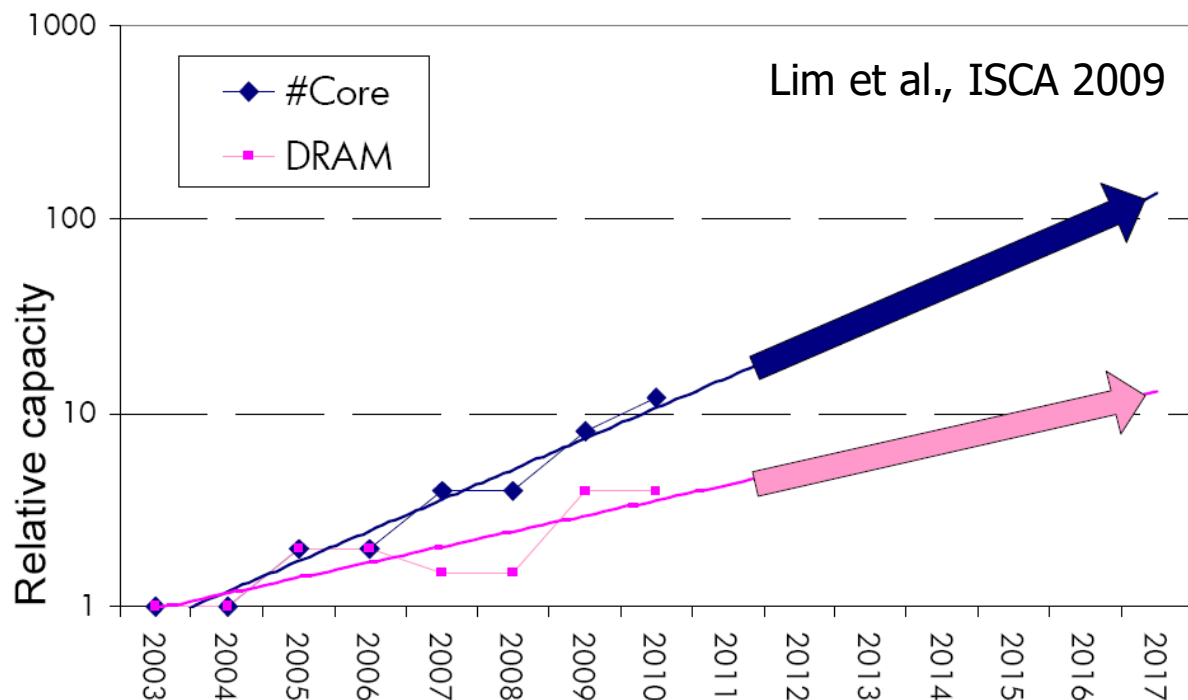
Major Trends Affecting Main Memory (II)

- Need for main memory capacity, bandwidth, QoS increasing
 - **Multi-core**: increasing number of cores/agents
 - **Data-intensive applications**: increasing demand/hunger for data
 - **Consolidation**: cloud computing, GPUs, mobile, heterogeneity
- Main memory energy/power is a key system design concern
- DRAM technology scaling is ending

Consequence: The Memory Capacity Gap

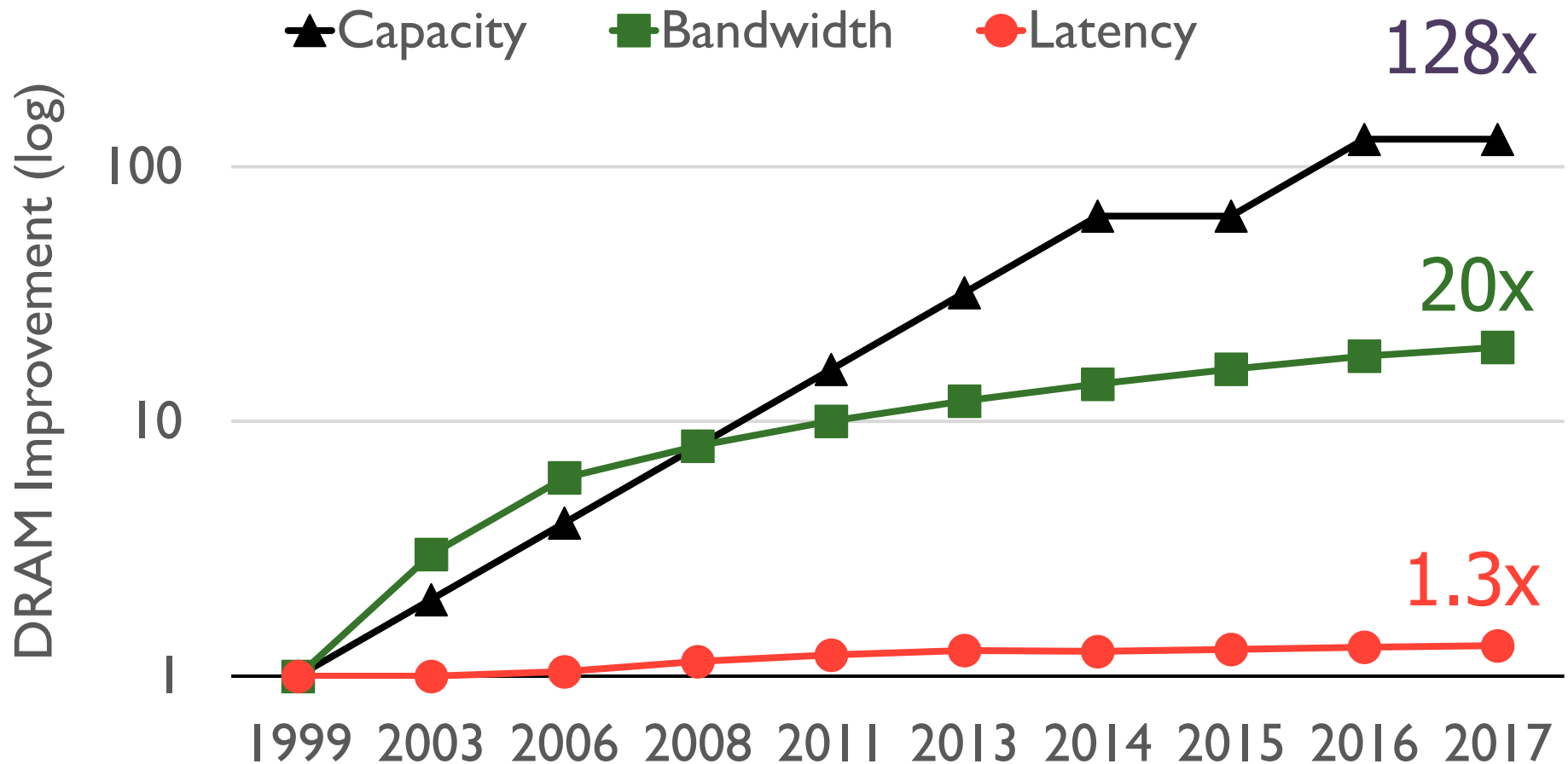
Core count doubling ~ every 2 years

DRAM DIMM capacity doubling ~ every 3 years

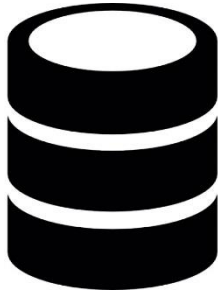


- *Memory capacity per core* expected to drop by 30% every two years
- Trends worse for *memory bandwidth per core*!

DRAM Capacity, Bandwidth & Latency



DRAM Is Critical for Performance



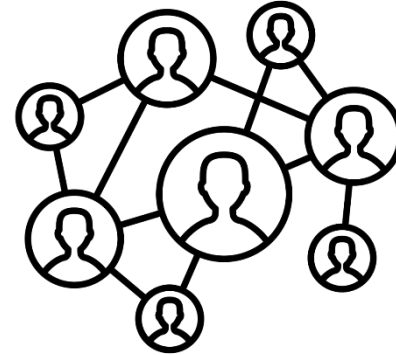
In-memory Databases

[Mao+, EuroSys'12;
Clapp+ (Intel), IISWC'15]



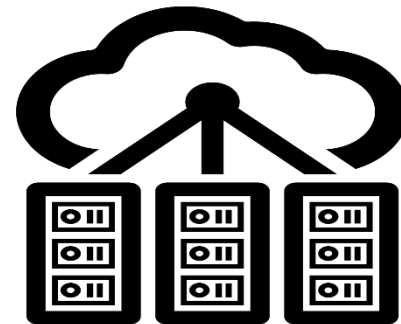
In-Memory Data Analytics

[Clapp+ (Intel), IISWC'15;
Awan+, BDCloud'15]



Graph/Tree Processing

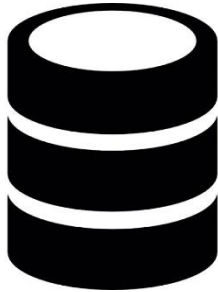
[Xu+, IISWC'12; Umuroglu+, FPL'15]



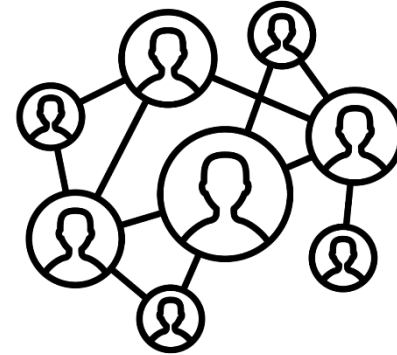
Datacenter Workloads

[Kanev+ (Google), ISCA'15]

DRAM Is Critical for Performance



In-memory Databases



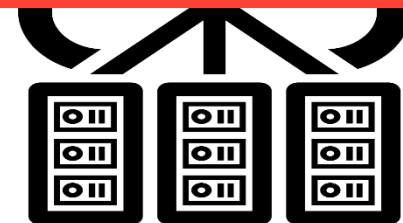
Graph/Tree Processing

Memory → performance bottleneck



In-Memory Data Analytics

[Clapp+ (Intel), IISWC'15;
Awan+, BDCloud'15]



Datacenter Workloads

[Kanev+ (Google), ISCA'15]

DRAM Is Critical for Performance



Chrome

Google's web browser



TensorFlow Mobile

Google's machine learning
framework

VP9



Video Playback

Google's **video codec**

VP9



Video Capture

Google's **video codec**

DRAM Is Critical for Performance



Chrome



TensorFlow Mobile

Memory → performance bottleneck

VP9



Video Playback

Google's **video codec**

VP9



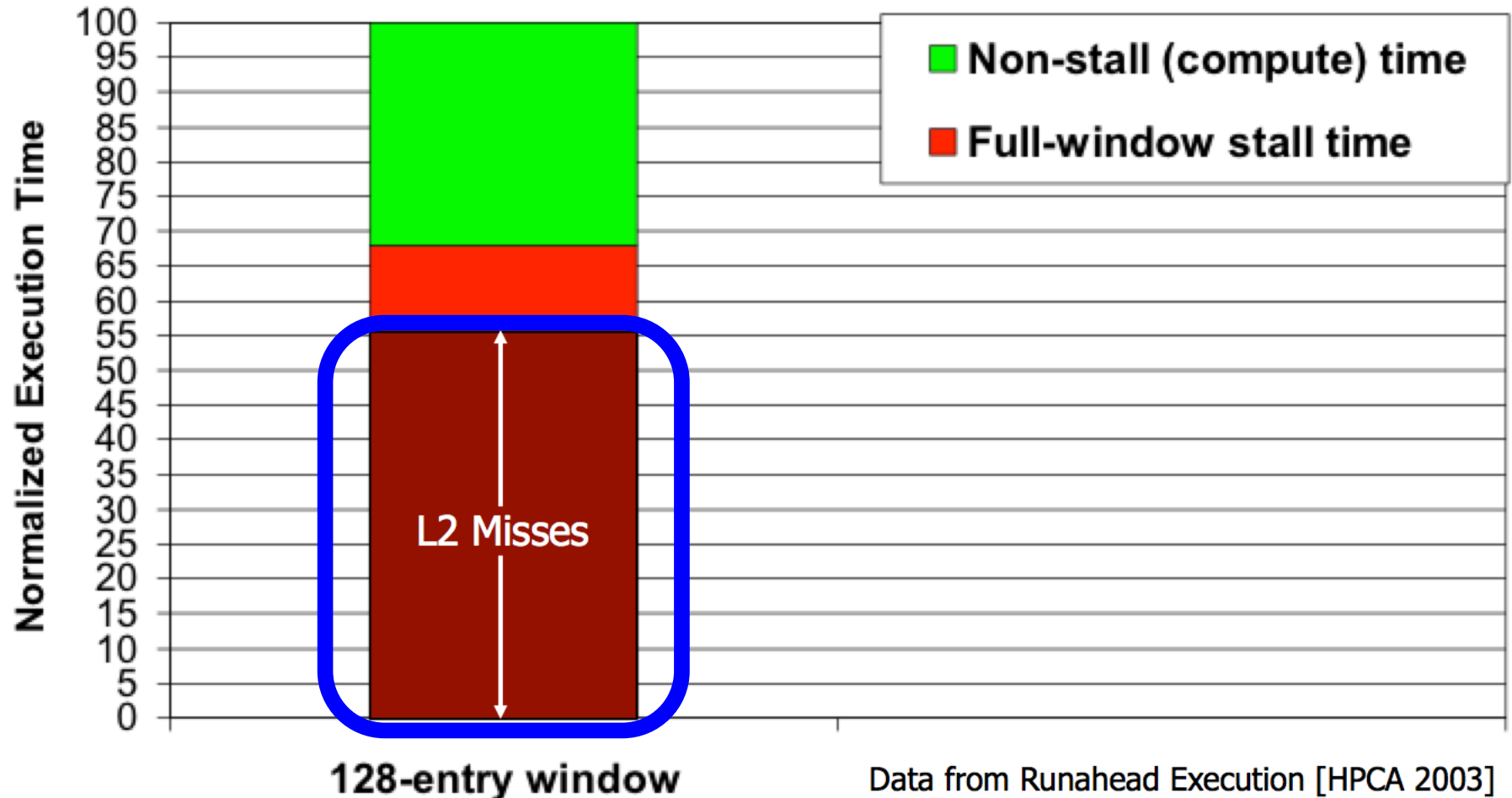
Video Capture

Google's **video codec**

Memory Bottleneck

I expect that over the coming decade memory subsystem design will be the *only* important design issue for microprocessors.

- **“It’s the Memory, Stupid!”** (Richard Sites, MPR, 1996)



The Memory Bottleneck

- Onur Mutlu, Jared Stark, Chris Wilkerson, and Yale N. Patt,
"Runahead Execution: An Alternative to Very Large Instruction Windows for Out-of-order Processors"
Proceedings of the 9th International Symposium on High-Performance Computer Architecture (HPCA), Anaheim, CA, February 2003. [Slides \(pdf\)](#)
One of the 15 computer architecture papers of 2003 selected as Top Picks by IEEE Micro.

Runahead Execution: An Alternative to Very Large Instruction Windows for Out-of-order Processors

Onur Mutlu § Jared Stark † Chris Wilkerson ‡ Yale N. Patt §

§ECE Department
The University of Texas at Austin
{onur,patt}@ece.utexas.edu

†Microprocessor Research
Intel Labs
jared.w.stark@intel.com

‡Desktop Platforms Group
Intel Corporation
chris.wilkerson@intel.com

The Memory Bottleneck

- Onur Mutlu, Jared Stark, Chris Wilkerson, and Yale N. Patt,
"Runahead Execution: An Effective Alternative to Large Instruction Windows"
IEEE Micro, Special Issue: Micro's Top Picks from Microarchitecture Conferences (MICRO TOP PICKS), Vol. 23, No. 6, pages 20-25, November/December 2003.

RUNAHEAD EXECUTION: AN EFFECTIVE ALTERNATIVE TO LARGE INSTRUCTION WINDOWS

It's the Memory, Stupid!

RICHARD SITES

It's the Memory, Stupid!

When we started the Alpha architecture design in 1988, we estimated a 25-year lifetime and a relatively modest 32% per year compounded performance improvement of implementations over that lifetime (1,000× total). We guesstimated about 10× would come from CPU clock improvement, 10× from multiple instruction issue, and 10× from multiple processors.

5, 1996  MICROPROCESSOR REPORT

An Informal Interview on Memory

- Madeleine Gray and Onur Mutlu,
"It's the memory, stupid': A conversation with Onur Mutlu"
HiPEAC info 55, HiPEAC Newsletter, October 2018.
[Shorter Version in Newsletter]
[Longer Online Version with References]

'It's the memory, stupid': A conversation with Onur Mutlu

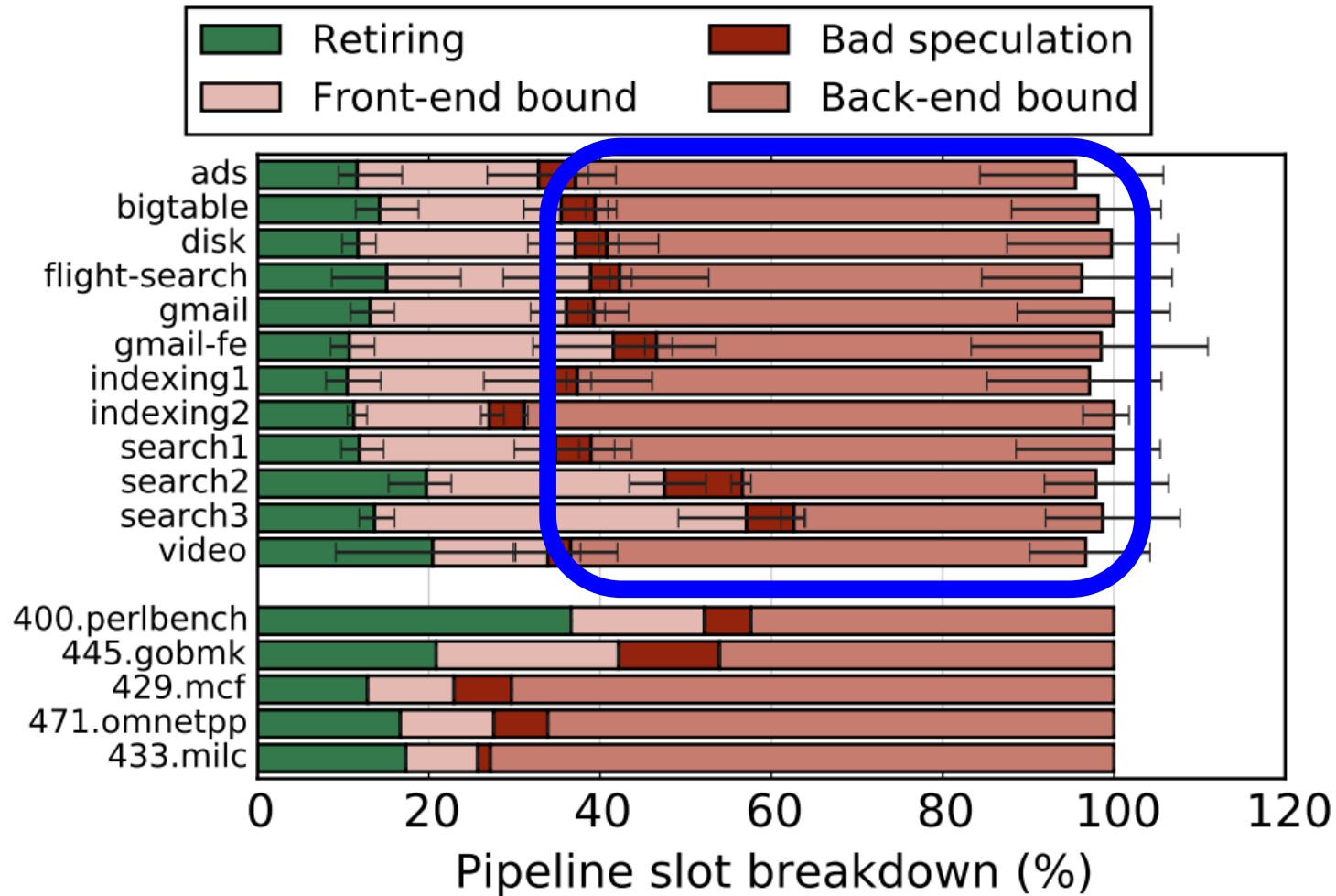
'We're beyond computation; we know how to do computation really well, we can optimize it, we can build all sorts of accelerators ... but the memory – how to feed the data, how to get the data into the accelerators – is a huge problem.'

This was how ETH Zürich and Carnegie Mellon Professor Onur Mutlu opened his course on memory systems and memory-centric computing systems at HiPEAC's summer school, ACACES18. A prolific publisher – he recently bagged the top spot on the International Symposium on Computer Architecture (ISCA) hall of fame – Onur is passionate about computation and communication that are efficient and secure by design. In advance of our Computing Systems Week focusing on data centres, storage, and networking, which takes place next week in Heraklion, HiPEAC picked his brains on all things data-based.



The Memory Bottleneck

- All of Google's Data Center Workloads (2015):



The Memory Bottleneck

- All of Google's Data Center Workloads (2015):

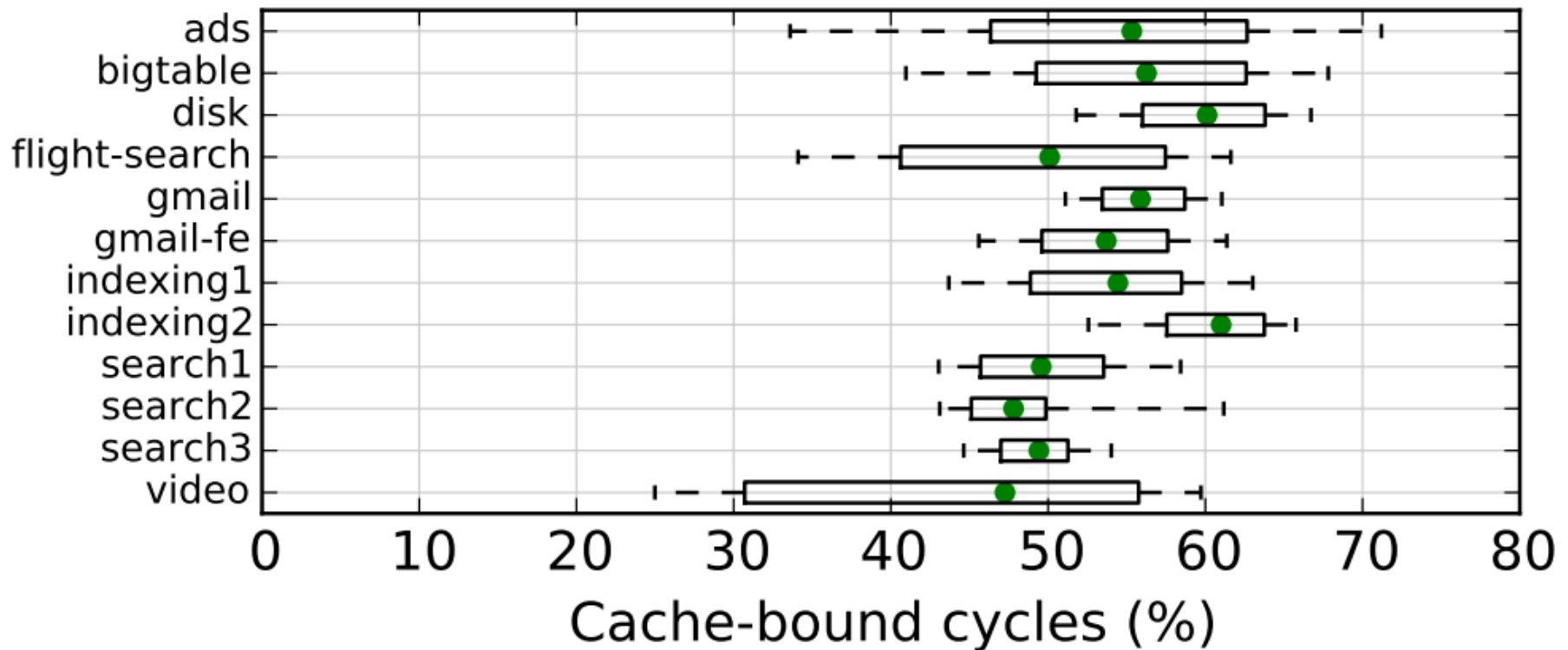


Figure 11: Half of cycles are spent stalled on caches.

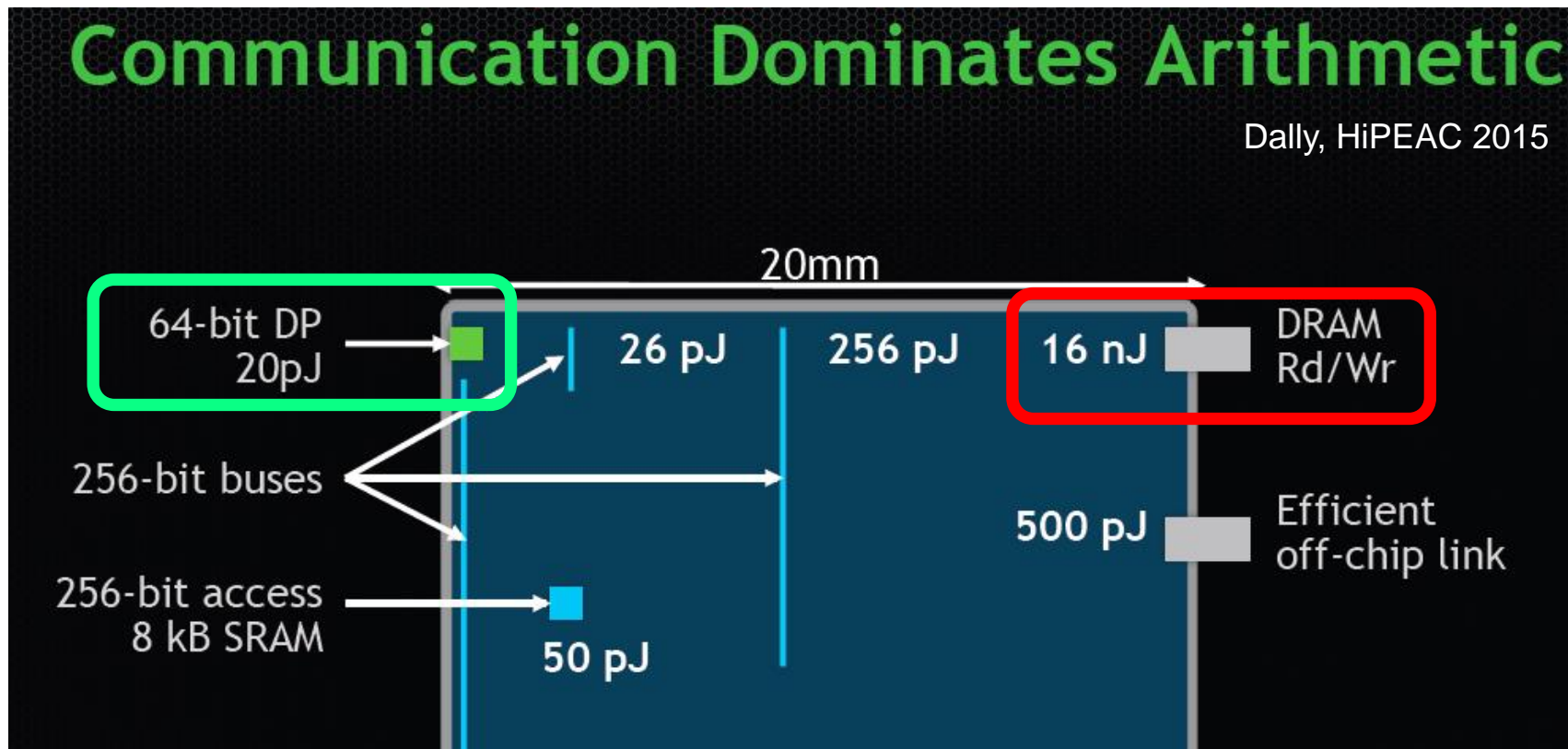
Major Trends Affecting Main Memory (III)

- Need for main memory capacity, bandwidth, QoS increasing
- Main memory energy/power is a key system design concern
 - ~40-50% energy spent in off-chip memory hierarchy [Lefurgy, IEEE Computer'03] >40% power in DRAM [Ware, HPCA'10][Paul, ISCA'15]
 - DRAM consumes power even when not used (periodic refresh)
- DRAM technology scaling is ending

Energy Cost of Data Movement

Communication Dominates Arithmetic

Dally, HiPEAC 2015



A memory access consumes $\sim 1000\times$ the energy of a complex addition

Energy Waste in Mobile Devices

- Amirali Boroumand, Saugata Ghose, Youngsok Kim, Rachata Ausavarungnirun, Eric Shiu, Rahul Thakur, Daehyun Kim, Aki Kuusela, Allan Knies, Parthasarathy Ranganathan, and Onur Mutlu, **"Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks"** *Proceedings of the 23rd International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, Williamsburg, VA, USA, March 2018.

**62.7% of the total system energy
is spent on data movement**

Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks

Amirali Boroumand¹

Saugata Ghose¹

Youngsok Kim²

Rachata Ausavarungnirun¹

Eric Shiu³

Rahul Thakur³

Daehyun Kim^{4,3}

Aki Kuusela³

Allan Knies³

Parthasarathy Ranganathan³

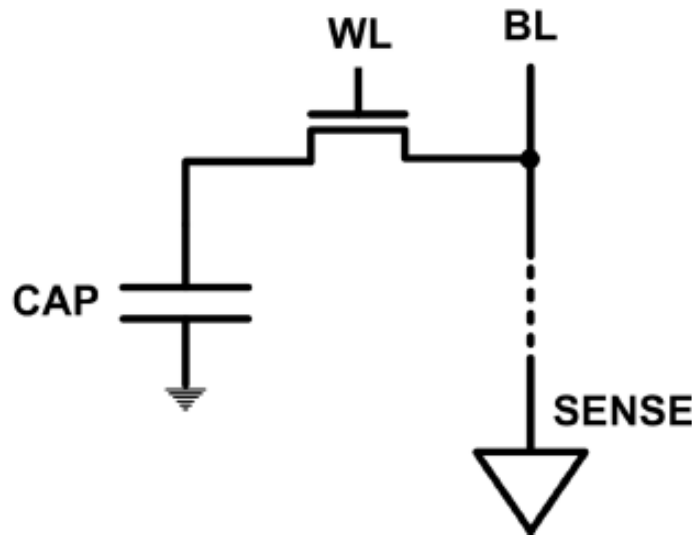
Onur Mutlu^{5,1}

Major Trends Affecting Main Memory (IV)

- Need for main memory capacity, bandwidth, QoS increasing
- Main memory energy/power is a key system design concern
- DRAM technology scaling is ending
 - ITRS projects DRAM will not scale easily below X nm
 - Scaling has provided many benefits:
 - higher capacity (density), lower cost, lower energy

The DRAM Scaling Problem

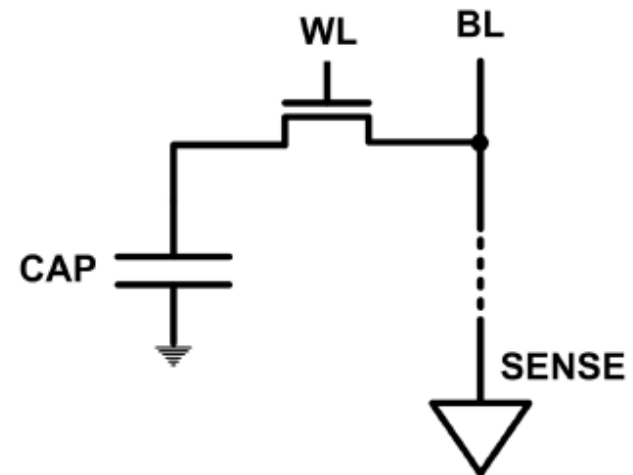
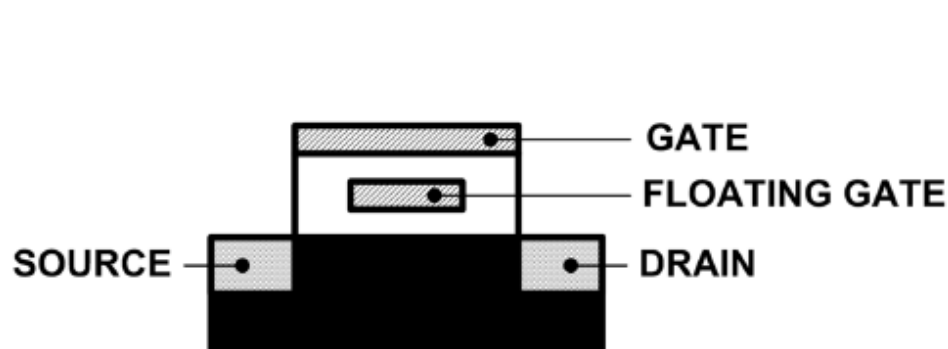
- DRAM stores charge in a capacitor (charge-based memory)
 - Capacitor must be large enough for reliable sensing
 - Access transistor should be large enough for low leakage and high retention time
 - Scaling beyond 40-35nm (2013) is challenging [ITRS, 2009]



- DRAM capacity, cost, and energy/power hard to scale

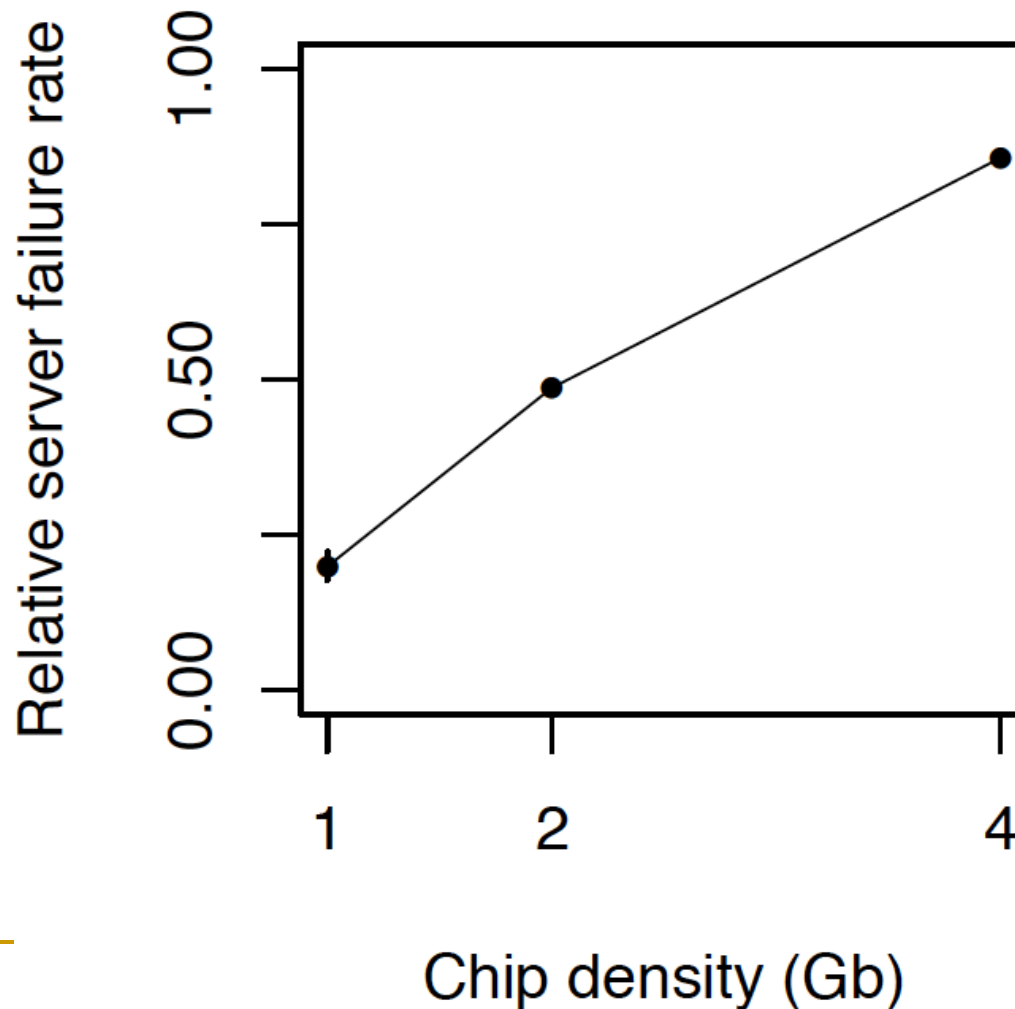
Limits of Charge Memory

- Difficult charge placement and control
 - Flash: floating gate charge
 - DRAM: capacitor charge, transistor leakage
- Data retention and reliable sensing becomes difficult as charge storage unit size reduces



As Memory Scales, It Becomes Unreliable

- Data from all of Facebook's servers worldwide
- Meza+, "Revisiting Memory Errors in Large-Scale Production Data Centers," DSN'15.



*Intuition:
quadratic
increase
in
capacity*

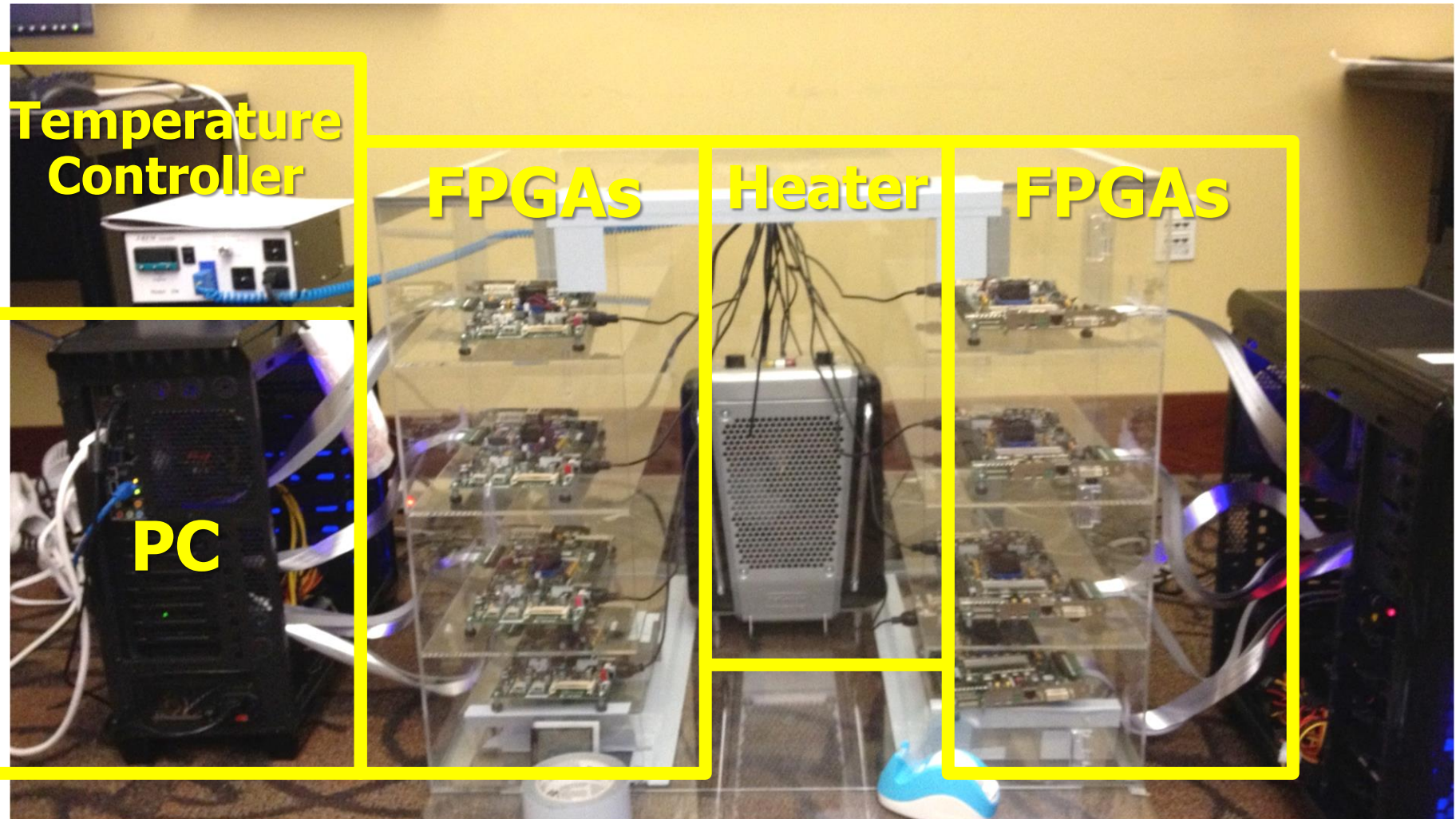
Large-Scale Failure Analysis of DRAM Chips

- Analysis and modeling of memory errors found in all of Facebook's server fleet
- Justin Meza, Qiang Wu, Sanjeev Kumar, and Onur Mutlu,
"Revisiting Memory Errors in Large-Scale Production Data Centers: Analysis and Modeling of New Trends from the Field"
Proceedings of the 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Rio de Janeiro, Brazil, June 2015.
[[Slides \(pptx\)](#)] [[pdf](#)] [[DRAM Error Model](#)]

Revisiting Memory Errors in Large-Scale Production Data Centers: Analysis and Modeling of New Trends from the Field

Justin Meza Qiang Wu* Sanjeev Kumar* Onur Mutlu
Carnegie Mellon University * Facebook, Inc.

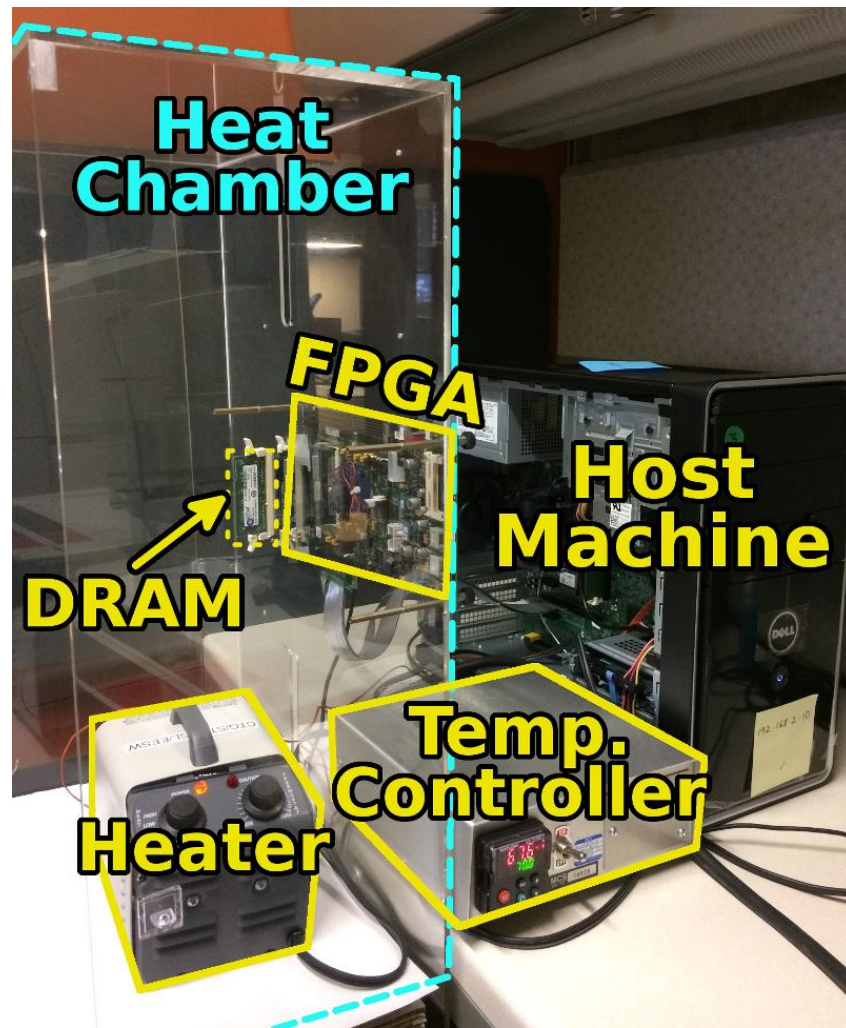
Infrastructures to Understand Such Issues



SoftMC: Open Source DRAM Infrastructure

- Hasan Hassan et al., “**SoftMC: A Flexible and Practical Open-Source Infrastructure for Enabling Experimental DRAM Studies**,” HPCA 2017.

- Flexible
- Easy to Use (C++ API)
- Open-source
github.com/CMU-SAFARI/SoftMC



- <https://github.com/CMU-SAFARI/SoftMC>

SoftMC: A Flexible and Practical Open-Source Infrastructure for Enabling Experimental DRAM Studies

Hasan Hassan^{1,2,3} Nandita Vijaykumar³ Samira Khan^{4,3} Saugata Ghose³ Kevin Chang³
Gennady Pekhimenko^{5,3} Donghyuk Lee^{6,3} Oguz Ergin² Onur Mutlu^{1,3}

¹*ETH Zürich* ²*TOBB University of Economics & Technology* ³*Carnegie Mellon University*
⁴*University of Virginia* ⁵*Microsoft Research* ⁶*NVIDIA Research*

A Curious Discovery [Kim et al., ISCA 2014]

One can
predictably induce errors
in most DRAM memory chips

DRAM RowHammer

A simple hardware failure mechanism
can create a widespread
system security vulnerability

WIRED

Forget Software—Now Hackers Are Exploiting Physics

BUSINESS	CULTURE	DESIGN	GEAR	SCIENCE
----------	---------	--------	------	---------

ANDY GREENBERG SECURITY 08.31.16 7:00 AM

SHARE



SHARE
18276



TWEET

FORGET SOFTWARE—NOW HACKERS ARE EXPLOITING PHYSICS

RowHammer: Six Years Ago...

- Yoongu Kim, Ross Daly, Jeremie Kim, Chris Fallin, Ji Hye Lee, Donghyuk Lee, Chris Wilkerson, Konrad Lai, and Onur Mutlu,
"Flipping Bits in Memory Without Accessing Them: An Experimental Study of DRAM Disturbance Errors"
Proceedings of the 41st International Symposium on Computer Architecture (ISCA), Minneapolis, MN, June 2014.
[[Slides \(pptx\)](#)] [[pdf](#)] [[Lightning Session Slides \(pptx\)](#)] [[pdf](#)] [[Source Code and Data](#)]

Flipping Bits in Memory Without Accessing Them: An Experimental Study of DRAM Disturbance Errors

Yoongu Kim¹ Ross Daly* Jeremie Kim¹ Chris Fallin* Ji Hye Lee¹
Donghyuk Lee¹ Chris Wilkerson² Konrad Lai Onur Mutlu¹

¹Carnegie Mellon University ²Intel Labs

The Reliability & Security Perspectives

- Onur Mutlu,
"The RowHammer Problem and Other Issues We May Face as Memory Becomes Denser"
Invited Paper in Proceedings of the Design, Automation, and Test in Europe Conference (DATE), Lausanne, Switzerland, March 2017.
[[Slides \(pptx\)](#)] [[pdf](#)]

The RowHammer Problem and Other Issues We May Face as Memory Becomes Denser

Onur Mutlu
ETH Zürich
onur.mutlu@inf.ethz.ch
<https://people.inf.ethz.ch/omutlu>

A RowHammer Retrospective

- Onur Mutlu and Jeremie Kim,
"RowHammer: A Retrospective"
IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD) Special Issue on Top Picks in Hardware and Embedded Security, 2019.
[[Preliminary arXiv version](#)]

RowHammer: A Retrospective

Onur Mutlu^{§‡} Jeremie S. Kim^{‡§}
§ETH Zürich ‡Carnegie Mellon University

RowHammer in 2020 (I)

- Jeremie S. Kim, Minesh Patel, A. Giray Yaglikci, Hasan Hassan, Roknoddin Azizi, Lois Orosa, and Onur Mutlu,
"Revisiting RowHammer: An Experimental Analysis of Modern Devices and Mitigation Techniques"
Proceedings of the 47th International Symposium on Computer Architecture (ISCA), Valencia, Spain, June 2020.
[[Slides \(pptx\)](#)] [[pdf](#)]
[[Lightning Talk Slides \(pptx\)](#)] [[pdf](#)]
[[Talk Video](#) (20 minutes)]
[[Lightning Talk Video](#) (3 minutes)]

Revisiting RowHammer: An Experimental Analysis of Modern DRAM Devices and Mitigation Techniques

Jeremie S. Kim^{§†} Minesh Patel[§] A. Giray Yağlıkçı[§]
Hasan Hassan[§] Roknoddin Azizi[§] Lois Orosa[§] Onur Mutlu^{§†}
[§]*ETH Zürich* [†]*Carnegie Mellon University*

RowHammer in 2020 (II)

- Pietro Frigo, Emanuele Vannacci, Hasan Hassan, Victor van der Veen, Onur Mutlu, Cristiano Giuffrida, Herbert Bos, and Kaveh Razavi, **"TRRespass: Exploiting the Many Sides of Target Row Refresh"**
Proceedings of the 41st IEEE Symposium on Security and Privacy (S&P), San Francisco, CA, USA, May 2020.
[[Slides \(pptx\)](#)] [[pdf](#)]
[[Talk Video](#) (17 minutes)]
[[Source Code](#)]
[[Web Article](#)]
Best paper award.

TRRespass: Exploiting the Many Sides of Target Row Refresh

Pietro Frigo^{*†} Emanuele Vannacci^{*†} Hasan Hassan[§] Victor van der Veen[¶]
Onur Mutlu[§] Cristiano Giuffrida^{*} Herbert Bos^{*} Kaveh Razavi^{*}

RowHammer in 2020 (III)

- Lucian Cojocar, Jeremie Kim, Minesh Patel, Lillian Tsai, Stefan Saroiu, Alec Wolman, and Onur Mutlu,
"Are We Susceptible to Rowhammer? An End-to-End Methodology for Cloud Providers"
Proceedings of the 41st IEEE Symposium on Security and Privacy (S&P), San Francisco, CA, USA, May 2020.
[[Slides \(pptx\)](#)] [[pdf](#)]
[[Talk Video](#) (17 minutes)]

Are We Susceptible to Rowhammer?

An End-to-End Methodology for Cloud Providers

Lucian Cojocar, Jeremie Kim^{§†}, Minesh Patel[§], Lillian Tsai[‡],
Stefan Saroiu, Alec Wolman, and Onur Mutlu^{§†}
Microsoft Research, [§]ETH Zürich, [†]CMU, [‡]MIT

Major Trends Affecting Main Memory (V)

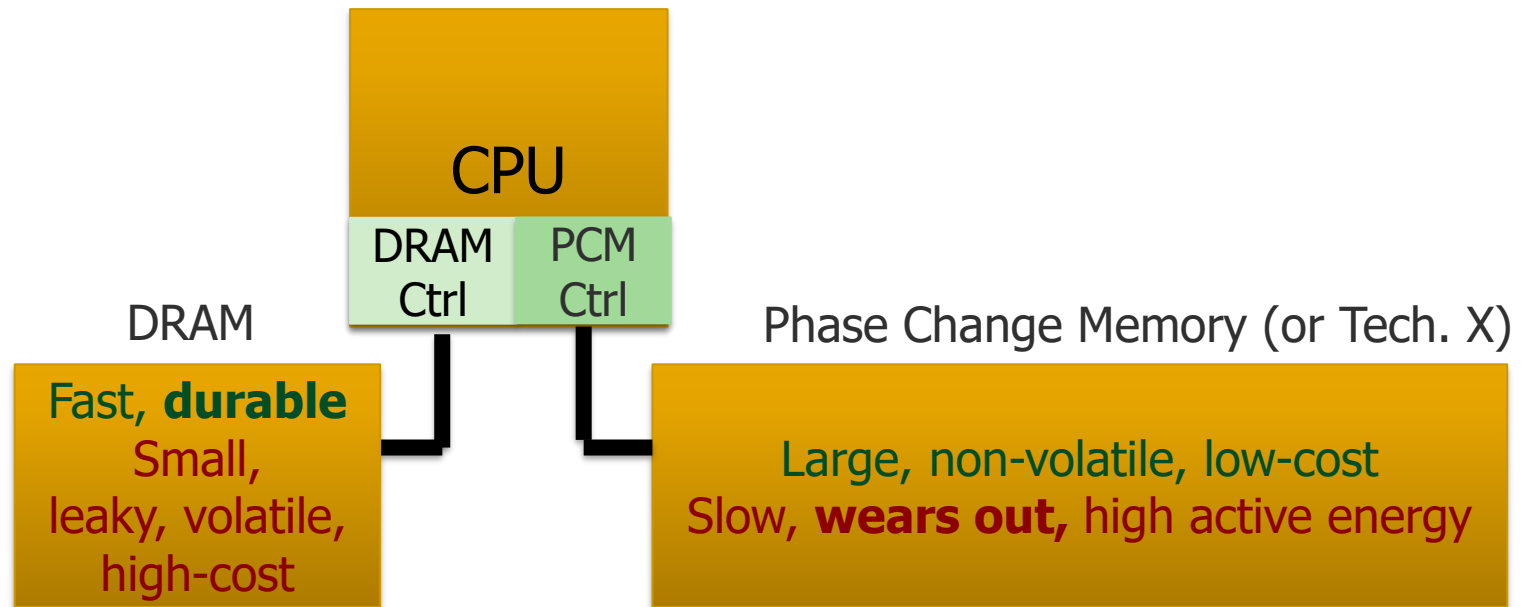
- DRAM scaling has already become increasingly difficult
 - Increasing cell leakage current, reduced cell reliability, increasing manufacturing difficulties [Kim+ ISCA 2014], [Liu+ ISCA 2013], [Mutlu IMW 2013], [Mutlu DATE 2017]
 - **Difficult to significantly improve capacity, energy**
- **Emerging memory technologies** are promising

Major Trends Affecting Main Memory (V)

- DRAM scaling has already become increasingly difficult
 - Increasing cell leakage current, reduced cell reliability, increasing manufacturing difficulties [Kim+ ISCA 2014], [Liu+ ISCA 2013], [Mutlu IMW 2013], [Mutlu DATE 2017]
 - **Difficult to significantly improve capacity, energy**
- **Emerging memory technologies** are promising

3D-Stacked DRAM	higher bandwidth	smaller capacity
Reduced-Latency DRAM (e.g., RL/TL-DRAM, FLY-RAM)	lower latency	higher cost
Low-Power DRAM (e.g., LPDDR3, LPDDR4, Voltron)	lower power	higher latency higher cost
Non-Volatile Memory (NVM) (e.g., PCM, STTRAM, ReRAM, 3D Xpoint)	larger capacity	higher latency higher dynamic power lower endurance

Major Trend: Hybrid Main Memory



Hardware/software manage data allocation and movement
to achieve the best of multiple technologies

Meza+, "[Enabling Efficient and Scalable Hybrid Memories](#)," IEEE Comp. Arch. Letters, 2012.
Yoon+, "[Row Buffer Locality Aware Caching Policies for Hybrid Memories](#)," ICCD 2012 Best Paper Award.

Main Memory Needs Intelligent Controllers

Industry Is Writing Papers About It, Too

DRAM Process Scaling Challenges

❖ Refresh

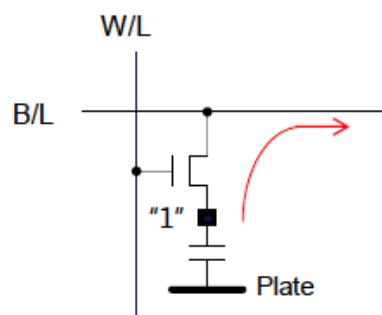
- Difficult to build high-aspect ratio cell capacitors decreasing cell capacitance
- Leakage current of cell access transistors increasing

❖ tWR

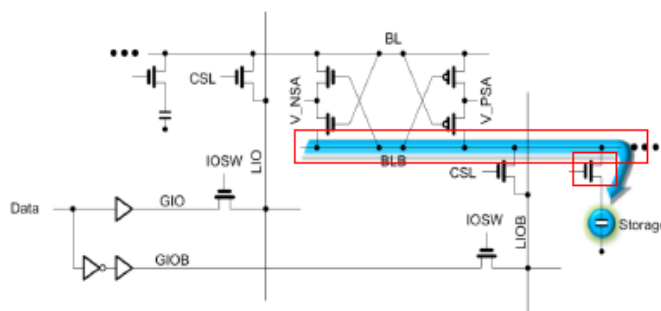
- Contact resistance between the cell capacitor and access transistor increasing
- On-current of the cell access transistor decreasing
- Bit-line resistance increasing

❖ VRT

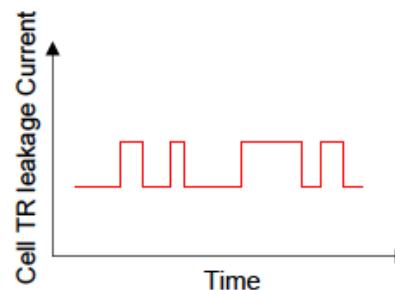
- Occurring more frequently with cell capacitance decreasing



Refresh



tWR



VRT

Call for Intelligent Memory Controllers

DRAM Process Scaling Challenges

❖ Refresh

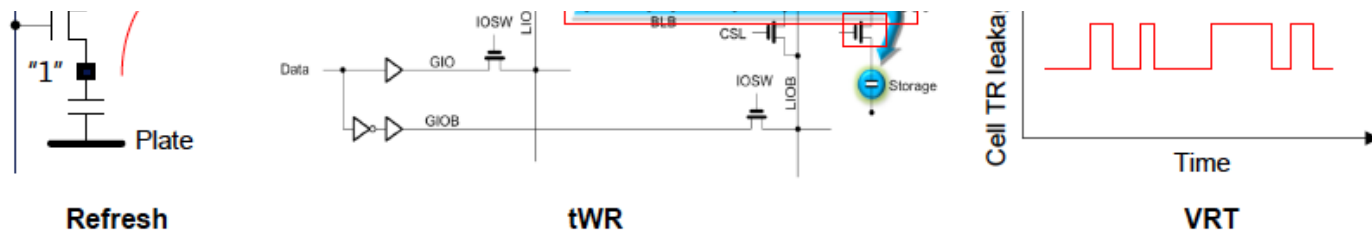
- Difficult to build high-aspect ratio cell capacitors decreasing cell capacitance

THE MEMORY FORUM 2014

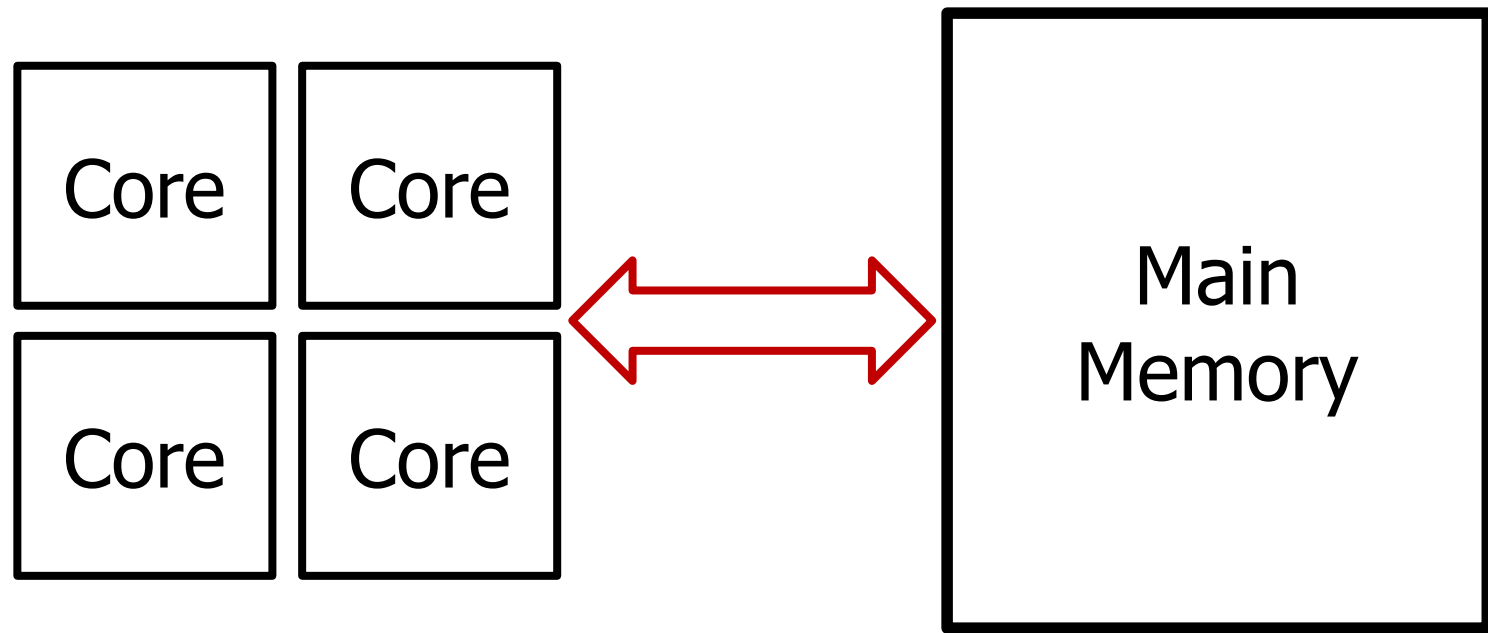
Co-Architecting Controllers and DRAM to Enhance DRAM Process Scaling

Uksong Kang, Hak-soo Yu, Churoo Park, *Hongzhong Zheng,
**John Halbert, **Kuljit Bains, SeongJin Jang, and Joo Sun Choi

*Samsung Electronics, Hwasung, Korea / *Samsung Electronics, San Jose / **Intel*

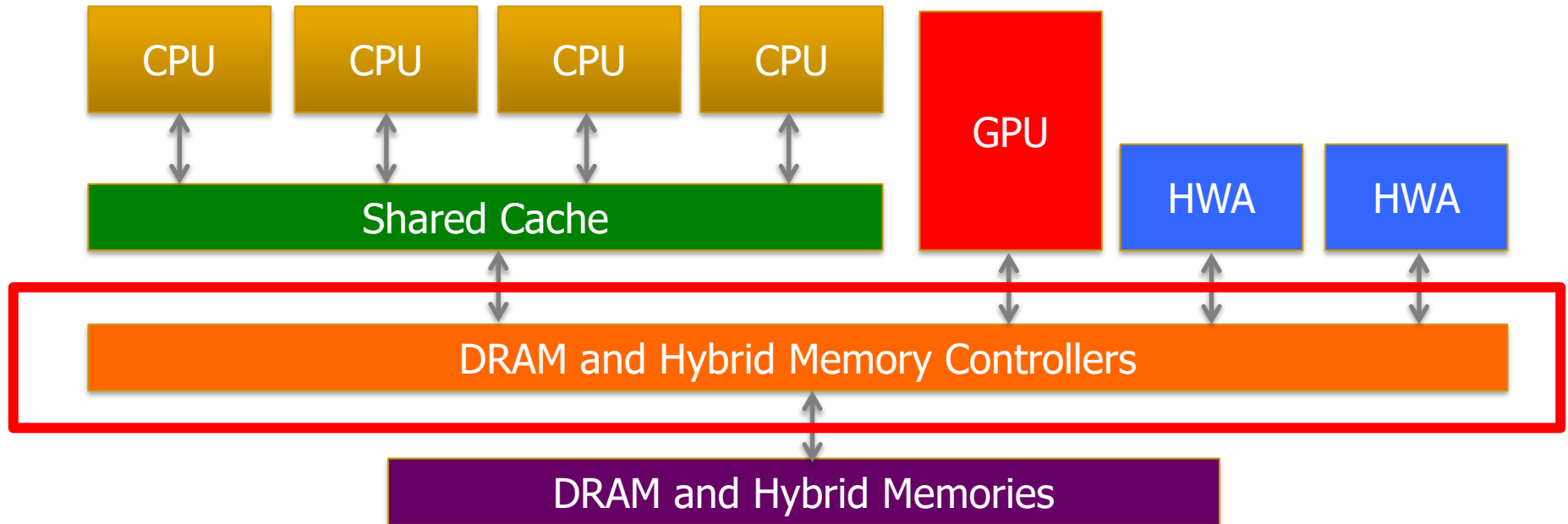


An Orthogonal Issue: Memory Interference



Cores' interfere with each other when accessing shared main memory
Uncontrolled interference leads to many problems (QoS, performance)

Goal: Predictable Performance in Complex Systems



- Heterogeneous agents: CPUs, GPUs, and HWAs
- Main memory interference between CPUs, GPUs, HWAs

How to allocate resources to heterogeneous agents to mitigate interference and provide predictable performance?

Main Memory Needs Intelligent Controllers

Computer Architecture

Lecture 3b: Memory Systems: Challenges and Opportunities

Prof. Onur Mutlu

ETH Zürich

Fall 2020

24 September 2020

We Did Not Cover The Rest of the Slides.
They Are For Your Benefit.

Trends, Challenges, and Opportunities in Main Memory

How Do We Solve The Memory Problem?

- **Fix it:** Make memory and controllers more intelligent
 - **New interfaces, functions, architectures:** system-mem codesign
- **Eliminate or minimize it:** Replace or (more likely) augment DRAM with a different technology
 - **New technologies and system-wide rethinking** of memory & storage
- **Embrace it:** Design heterogeneous memories (none of which are perfect) and map data intelligently across them
 - **New models for data management and maybe usage**
- ...

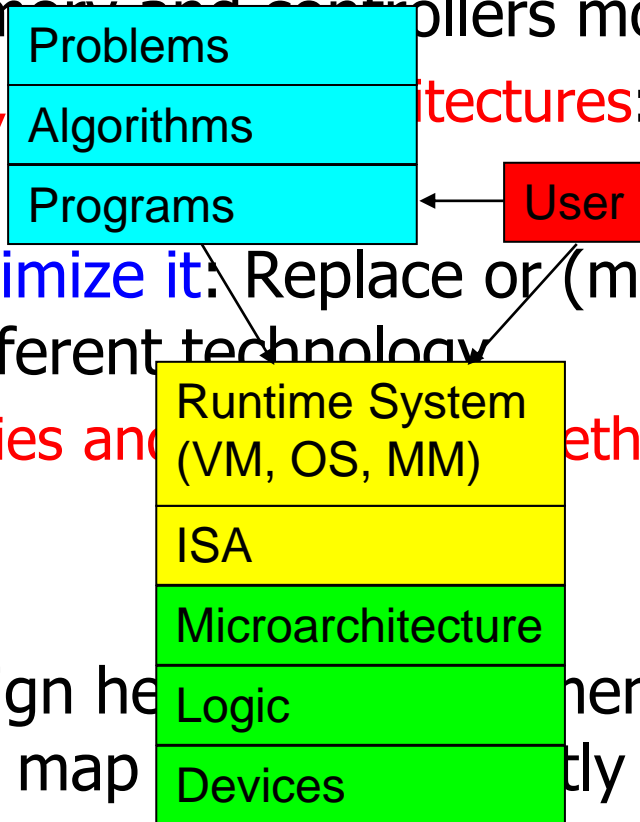
How Do We Solve The Memory Problem?

- **Fix it:** Make memory and controllers more intelligent
 - **New interfaces, functions, architectures:** system-mem codesign
- **Eliminate or minimize it:** Replace or (more likely) augment DRAM with a different technology
 - **New technologies and system-wide rethinking** of memory & storage
- **Embrace it:** Design heterogeneous memories (none of which are perfect) and map data intelligently across them
 - **New models for data management and maybe usage**

Solutions (to memory scaling) require software/hardware/device cooperation

How Do We Solve The Memory Problem?

- **Fix it:** Make memory and controllers more intelligent
 - **New interfaces, architectures:** system-mem codesign
- **Eliminate or minimize it:** Replace or (more likely) augment DRAM with a different technology
 - **New technologies and storage**
- **Embrace it:** Design heterogeneous memories (none of which are perfect) and map applications across them
 - **New models for data management and maybe usage**



Solutions (to memory scaling) require software/hardware/device cooperation

Solution 1: New Memory Architectures

- Overcome memory shortcomings with
 - ❑ Memory-centric system design
 - ❑ Novel memory architectures, interfaces, functions
 - ❑ Better waste management (efficient utilization)
- Key issues to tackle
 - ❑ Enable reliability at low cost → high capacity
 - ❑ Reduce energy
 - ❑ Reduce latency
 - ❑ Improve bandwidth
 - ❑ Reduce waste (capacity, bandwidth, latency)
 - ❑ Enable computation close to data

Solution 1: New Memory Architectures

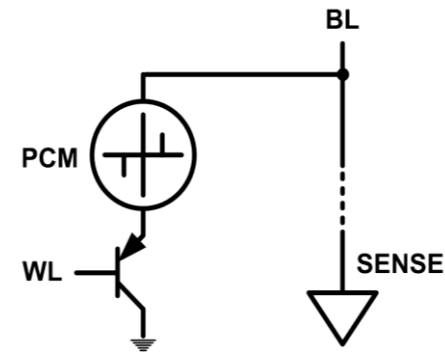
- Liu+, "RAIDR: Retention-Aware Intelligent DRAM Refresh," ISCA 2012.
- Kim+, "A Case for Exploiting Subarray-Level Parallelism in DRAM," ISCA 2012.
- Lee+, "Tiered-Latency DRAM: A Low Latency and Low Cost DRAM Architecture," HPCA 2013.
- Liu+, "An Experimental Study of Data Retention Behavior in Modern DRAM Devices," ISCA 2013.
- Seshadri+, "RowClone: Fast and Efficient In-DRAM Copy and Initialization of Bulk Data," MICRO 2013.
- Pekhimenko+, "Linearly Compressed Pages: A Main Memory Compression Framework," MICRO 2013.
- Chang+, "Improving DRAM Performance by Parallelizing Refreshes with Accesses," HPCA 2014.
- Khan+, "The Efficacy of Error Mitigation Techniques for DRAM Retention Failures: A Comparative Experimental Study," SIGMETRICS 2014.
- Luo+, "Characterizing Application Memory Error Vulnerability to Optimize Data Center Cost," DSN 2014.
- Kim+, "Flipping Bits in Memory Without Accessing Them: An Experimental Study of DRAM Disturbance Errors," ISCA 2014.
- Lee+, "Adaptive-Latency DRAM: Optimizing DRAM Timing for the Common-Case," HPCA 2015.
- Qureshi+, "AVATAR: A Variable-Retention-Time (VRT) Aware Refresh for DRAM Systems," DSN 2015.
- Meza+, "Revisiting Memory Errors in Large-Scale Production Data Centers: Analysis and Modeling of New Trends from the Field," DSN 2015.
- Kim+, "Ramulator: A Fast and Extensible DRAM Simulator," IEEE CAL 2015.
- Seshadri+, "Fast Bulk Bitwise AND and OR in DRAM," IEEE CAL 2015.
- Ahn+, "A Scalable Processing-in-Memory Accelerator for Parallel Graph Processing," ISCA 2015.
- Ahn+, "PIM-Enabled Instructions: A Low-Overhead, Locality-Aware Processing-in-Memory Architecture," ISCA 2015.
- Lee+, "Decoupled Direct Memory Access: Isolating CPU and IO Traffic by Leveraging a Dual-Data-Port DRAM," PACT 2015.
- Seshadri+, "Gather-Scatter DRAM: In-DRAM Address Translation to Improve the Spatial Locality of Non-unit Strided Accesses," MICRO 2015.
- Lee+, "Simultaneous Multi-Layer Access: Improving 3D-Stacked Memory Bandwidth at Low Cost," TACO 2016.
- Hassan+, "ChargeCache: Reducing DRAM Latency by Exploiting Row Access Locality," HPCA 2016.
- Chang+, "Low-Cost Inter-Linked Subarrays (LISA): Enabling Fast Inter-Subarray Data Migration in DRAM," HPCA 2016.
- Chang+, "Understanding Latency Variation in Modern DRAM Chips: Experimental Characterization, Analysis, and Optimization," SIGMETRICS 2016.
- Khan+, "PARBOR: An Efficient System-Level Technique to Detect Data Dependent Failures in DRAM," DSN 2016.
- Hsieh+, "Transparent Offloading and Mapping (TOM): Enabling Programmer-Transparent Near-Data Processing in GPU Systems," ISCA 2016.
- Hashemi+, "Accelerating Dependent Cache Misses with an Enhanced Memory Controller," ISCA 2016.
- Boroumand+, "LazyPIM: An Efficient Cache Coherence Mechanism for Processing-in-Memory," IEEE CAL 2016.
- Pattnaik+, "Scheduling Techniques for GPU Architectures with Processing-In-Memory Capabilities," PACT 2016.
- Hsieh+, "Accelerating Pointer Chasing in 3D-Stacked Memory: Challenges, Mechanisms, Evaluation," ICCD 2016.
- Hashemi+, "Continuous Runahead: Transparent Hardware Acceleration for Memory Intensive Workloads," MICRO 2016.
- Khan+, "A Case for Memory Content-Based Detection and Mitigation of Data-Dependent Failures in DRAM," IEEE CAL 2016.
- Hassan+, "SoftMC: A Flexible and Practical Open-Source Infrastructure for Enabling Experimental DRAM Studies," HPCA 2017.
- Mutlu, "The RowHammer Problem and Other Issues We May Face as Memory Becomes Denser," DATE 2017.
- Lee+, "Design-Induced Latency Variation in Modern DRAM Chips: Characterization, Analysis, and Latency Reduction Mechanisms," SIGMETRICS 2017.
- Chang+, "Understanding Reduced-Voltage Operation in Modern DRAM Devices: Experimental Characterization, Analysis, and Mechanisms," SIGMETRICS 2017.
- Patel+, "The Reach Profiler (REAPER): Enabling the Mitigation of DRAM Retention Failures via Profiling at Aggressive Conditions," ISCA 2017.
- Seshadri and Mutlu, "Simple Operations in Memory to Reduce Data Movement," ADCOM 2017.
- Liu+, "Concurrent Data Structures for Near-Memory Computing," SPAA 2017.
- Khan+, "Detecting and Mitigating Data-Dependent DRAM Failures by Exploiting Current Memory Content," MICRO 2017.
- Seshadri+, "Ambit: In-Memory Accelerator for Bulk Bitwise Operations Using Commodity DRAM Technology," MICRO 2017.
- Kim+, "GRIM-Filter: Fast Seed Location Filtering in DNA Read Mapping Using Processing-in-Memory Technologies," BMC Genomics 2018.
- Kim+, "The DRAM Latency PUF: Quickly Evaluating Physical Unclonable Functions by Exploiting the Latency-Reliability Tradeoff in Modern DRAM Devices," HPCA 2018.
- Boroumand+, "Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks," ASPLOS 2018.
- Das+, "VRL-DRAM: Improving DRAM Performance via Variable Refresh Latency," DAC 2018.
- Ghose+, "What Your DRAM Power Models Are Not Telling You: Lessons from a Detailed Experimental Study," SIGMETRICS 2018.
- Kim+, "Solar-DRAM: Reducing DRAM Access Latency by Exploiting the Variation in Local Bitlines," ICCD 2018.
- Wang+, "Reducing DRAM Latency via Charge-Level-Aware Look-Ahead Partial Restoration," MICRO 2018.
- Kim+, "D-RaNGe: Using Commodity DRAM Devices to Generate True Random Numbers with Low Latency and High Throughput," HPCA 2019.
- Singh+, "NAPEL: Near-Memory Computing Application Performance Prediction via Ensemble Learning," DAC 2019.
- Ghose+, "Demystifying Workload-DRAM Interactions: An Experimental Study," SIGMETRICS 2019.
- Patel+, "Understanding and Modeling On-Die Error Correction in Modern DRAM: An Experimental Study Using Real Devices," DSN 2019.
- Boroumand+, "CoNDA: Efficient Cache Coherence Support for Near-Data Accelerators," ISCA 2019.
- Hassan+, "CROW: A Low-Cost Substrate for Improving DRAM Performance, Energy Efficiency, and Reliability," ISCA 2019.
- Mutlu and Kim, "RowHammer: A Retrospective," TCAD 2019.
- Mutlu+, "Processing Data Where It Makes Sense: Enabling In-Memory Computation," MICPRO 2019.
- Seshadri and Mutlu, "In-DRAM Bulk Bitwise Execution Engine," ADCOM 2020.
- Koppula+, "EDEN: Energy-Efficient, High-Performance Neural Network Inference Using Approximate DRAM," MICRO 2019.
- Avoid DRAM:
 - Seshadri+, "The Evicted-Address Filter: A Unified Mechanism to Address Both Cache Pollution and Thrashing," PACT 2012.
 - Pekhimenko+, "Base-Delta-Immediate Compression: Practical Data Compression for On-Chip Caches," PACT 2012.
 - Seshadri+, "The Dirty-Block Index," ISCA 2014.
 - Pekhimenko+, "Exploiting Compressed Block Size as an Indicator of Future Reuse," HPCA 2015.
 - Vijaykumar+, "A Case for Core-Assisted Bottleneck Acceleration in GPUs: Enabling Flexible Data Compression with Assist Warps," ISCA 2015.
 - Pekhimenko+, "Toggle-Aware Bandwidth Compression for GPUs," HPCA 2016.

Solution 2: Emerging Memory Technologies

- Some emerging **resistive** memory technologies seem more scalable than DRAM (and they are non-volatile)

- Example: Phase Change Memory

- Data stored by changing phase of material
- Data read by detecting material's resistance
- Expected to scale to 9nm (2022 [ITRS 2009])
- Prototyped at 20nm (Raoux+, IBM JRD 2008)
- Expected to be denser than DRAM: can store multiple bits/cell

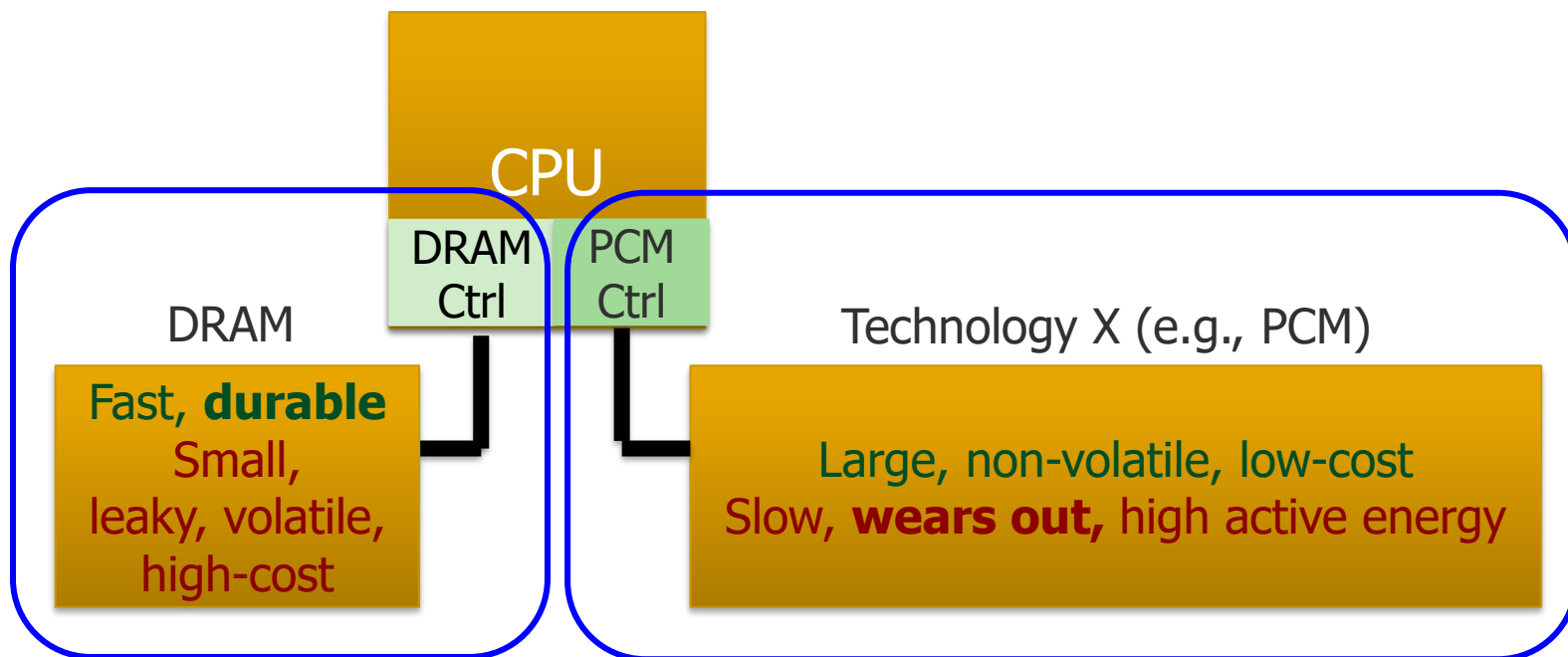


- But, emerging technologies have (many) shortcomings
 - Can they be enabled to replace/augment/surpass DRAM?

Solution 2: Emerging Memory Technologies

- Lee+, “[Architecting Phase Change Memory as a Scalable DRAM Alternative](#),” ISCA’09, CACM’10, IEEE Micro’10.
- Meza+, “[Enabling Efficient and Scalable Hybrid Memories](#),” IEEE Comp. Arch. Letters 2012.
- Yoon, Meza+, “[Row Buffer Locality Aware Caching Policies for Hybrid Memories](#),” ICCD 2012.
- Kultursay+, “[Evaluating STT-RAM as an Energy-Efficient Main Memory Alternative](#),” ISPASS 2013.
- Meza+, “[A Case for Efficient Hardware-Software Cooperative Management of Storage and Memory](#),” WEED 2013.
- Lu+, “[Loose Ordering Consistency for Persistent Memory](#),” ICCD 2014.
- Zhao+, “[FIRM: Fair and High-Performance Memory Control for Persistent Memory Systems](#),” MICRO 2014.
- Yoon, Meza+, “[Efficient Data Mapping and Buffering Techniques for Multi-Level Cell Phase-Change Memories](#),” TACO 2014.
- Ren+, “[ThyNVM: Enabling Software-Transparent Crash Consistency in Persistent Memory Systems](#),” MICRO 2015.
- Chauhan+, “[NVMove: Helping Programmers Move to Byte-Based Persistence](#),” INFLOW 2016.
- Li+, “[Utility-Based Hybrid Memory Management](#),” CLUSTER 2017.
- Yu+, “[Banshee: Bandwidth-Efficient DRAM Caching via Software/Hardware Cooperation](#),” MICRO 2017.
- Tavakkol+, “[MQSim: A Framework for Enabling Realistic Studies of Modern Multi-Queue SSD Devices](#),” FAST 2018.
- Tavakkol+, “[FLIN: Enabling Fairness and Enhancing Performance in Modern NVMe Solid State Drives](#),” ISCA 2018.
- Sadrosadati+. “[LTRF: Enabling High-Capacity Register Files for GPUs via Hardware/Software Cooperative Register Prefetching](#),” ASPLOS 2018.
- Salkhordeh+, “[An Analytical Model for Performance and Lifetime Estimation of Hybrid DRAM-NVM Main Memories](#),” TC 2019.
- Wang+, “[Panthera: Holistic Memory Management for Big Data Processing over Hybrid Memories](#),” PLDI 2019.
- Song+, “[Enabling and Exploiting Partition-Level Parallelism \(PALP\) in Phase Change Memories](#),” CASES 2019.
- Liu+, “[Binary Star: Coordinated Reliability in Heterogeneous Memory Systems for High Performance and Scalability](#),” MICRO’19.

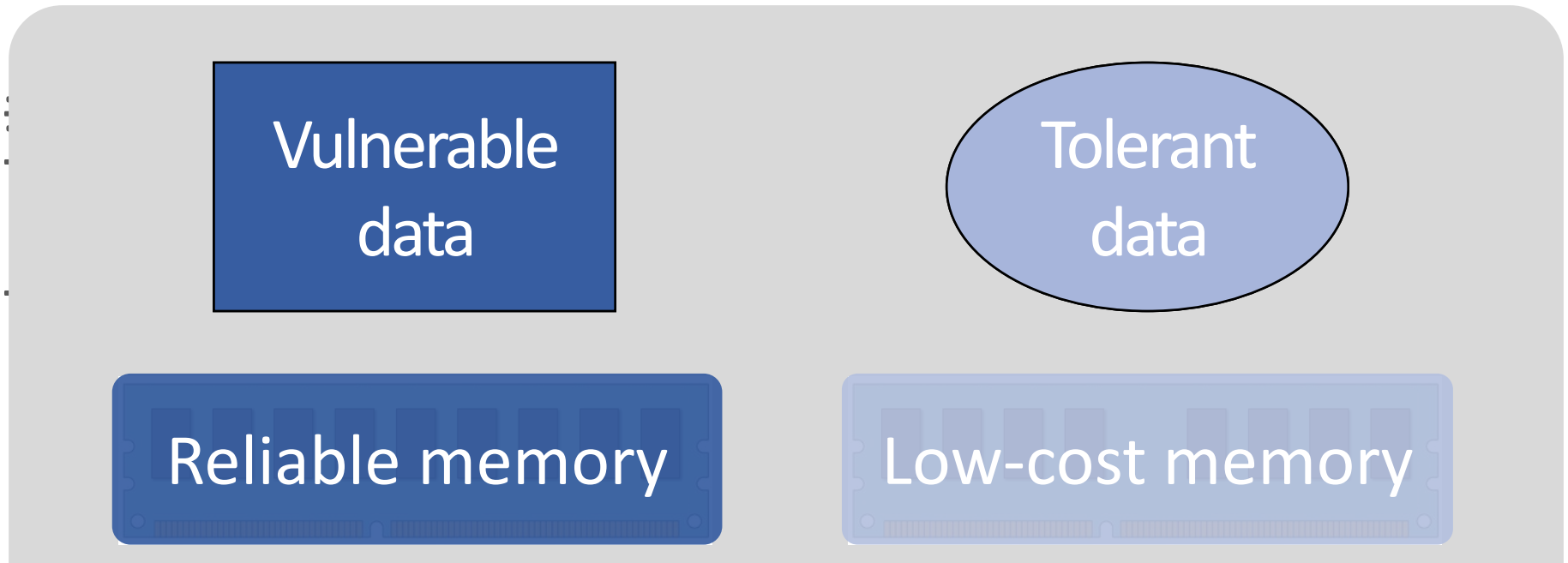
Combination: Hybrid Memory Systems



Hardware/software manage data allocation and movement
to achieve the best of multiple technologies

Meza+, "Enabling Efficient and Scalable Hybrid Memories," IEEE Comp. Arch. Letters, 2012.
Yoon, Meza et al., "Row Buffer Locality Aware Caching Policies for Hybrid Memories," ICCD
2012 Best Paper Award.

Exploiting Memory Error Tolerance with Hybrid Memory Systems



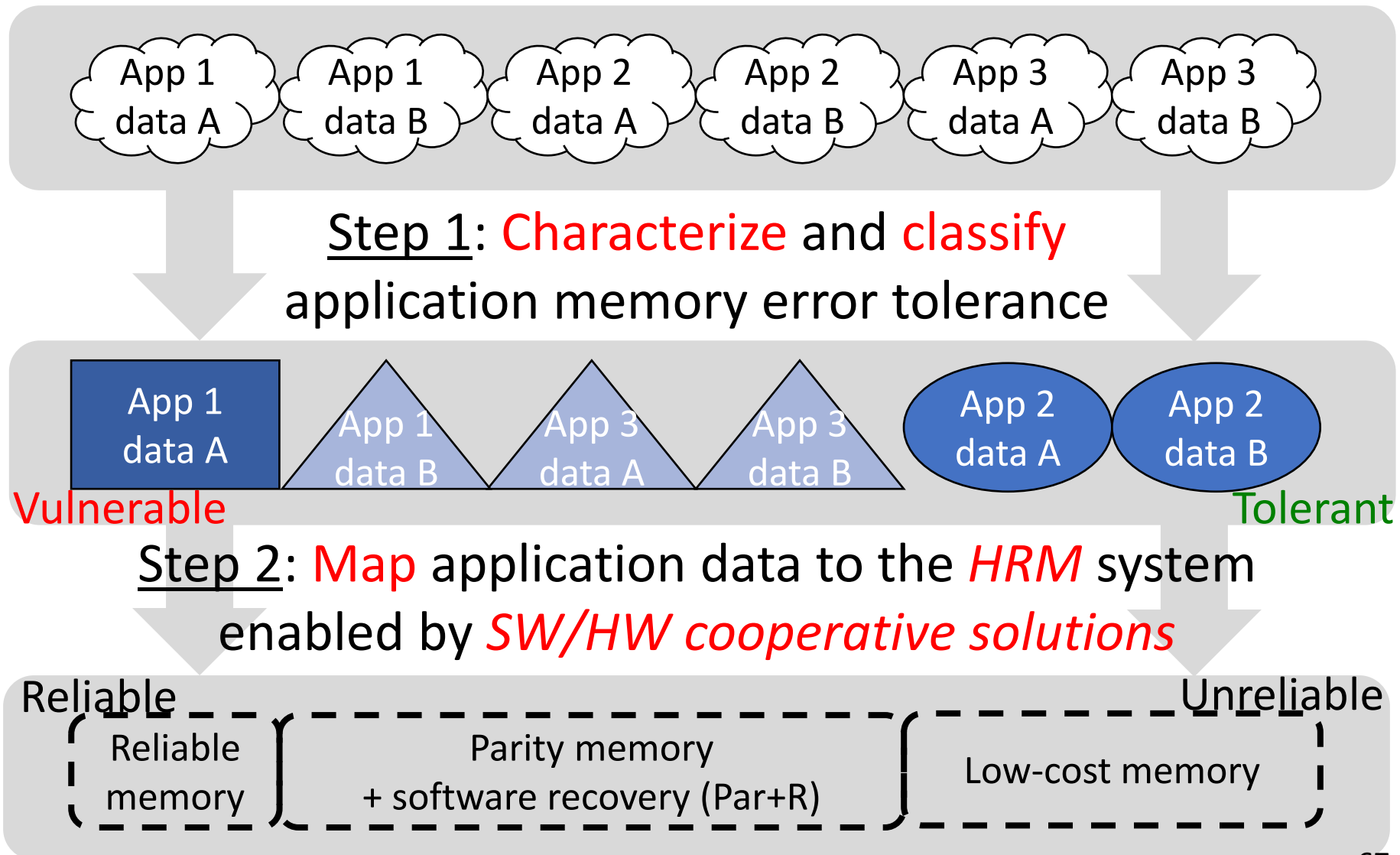
On Microsoft's Web Search workload

Reduces server hardware **cost** by **4.7 %**

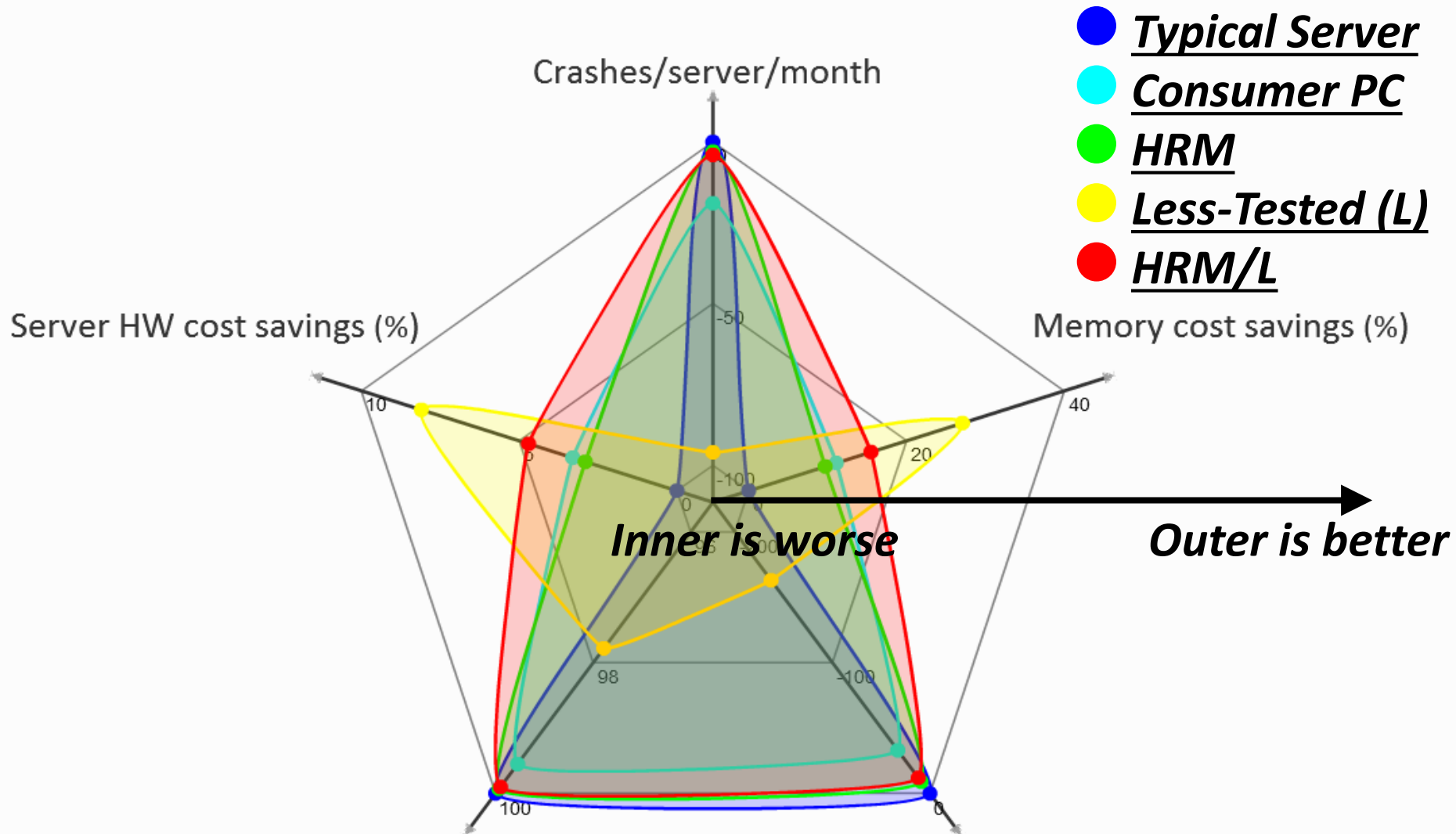
Achieves single server **availability** target of **99.90 %**

Heterogeneous-Reliability Memory [DSN 2014]

Heterogeneous-Reliability Memory



Evaluation Results



● ● Bigger area means better tradeoff

More on Heterogeneous Reliability Memory

- Yixin Luo, Sriram Govindan, Bikash Sharma, Mark Santaniello, Justin Meza, Aman Kansal, Jie Liu, Badriddine Khessib, Kushagra Vaid, and Onur Mutlu,
"Characterizing Application Memory Error Vulnerability to Optimize Data Center Cost via Heterogeneous-Reliability Memory"
Proceedings of the 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Atlanta, GA, June 2014. [[Summary](#)]
[[Slides \(pptx\)](#)] [[pdf](#)] [[Coverage on ZDNet](#)]

Characterizing Application Memory Error Vulnerability to Optimize Datacenter Cost via Heterogeneous-Reliability Memory

Yixin Luo Sriram Govindan* Bikash Sharma* Mark Santaniello* Justin Meza
Aman Kansal* Jie Liu* Badriddine Khessib* Kushagra Vaid* Onur Mutlu

Carnegie Mellon University, yixinluo@cs.cmu.edu, {meza, onur}@cmu.edu

*Microsoft Corporation, {srgovin, bsharma, marksan, kansal, jie.liu, bk Hessib, kvaid}@microsoft.com

An Orthogonal Issue: Memory Interference

- Problem: **Memory interference between cores is uncontrolled**
 - unfairness, starvation, low performance
 - **uncontrollable, unpredictable, vulnerable system**
 - Solution: **QoS-Aware Memory Systems**
 - Hardware designed to provide a configurable fairness substrate
 - Application-aware memory scheduling, partitioning, throttling
 - Software designed to configure the resources to satisfy different QoS goals
 - QoS-aware memory systems can provide predictable performance and higher efficiency
-

Strong Memory Service Guarantees

- Goal: Satisfy performance/SLA requirements in the presence of shared main memory, heterogeneous agents, and hybrid memory/storage
- Approach:
 - Develop techniques/models to accurately estimate the performance loss of an application/agent in the presence of resource sharing
 - Develop mechanisms (hardware and software) to enable the resource partitioning/prioritization needed to achieve the required performance levels for all applications
 - All the while providing high system performance
- Subramanian et al., “MISE: Providing Performance Predictability and Improving Fairness in Shared Main Memory Systems,” HPCA 2013.
- Subramanian et al., “The Application Slowdown Model,” MICRO 2015.

End of Backup Slides