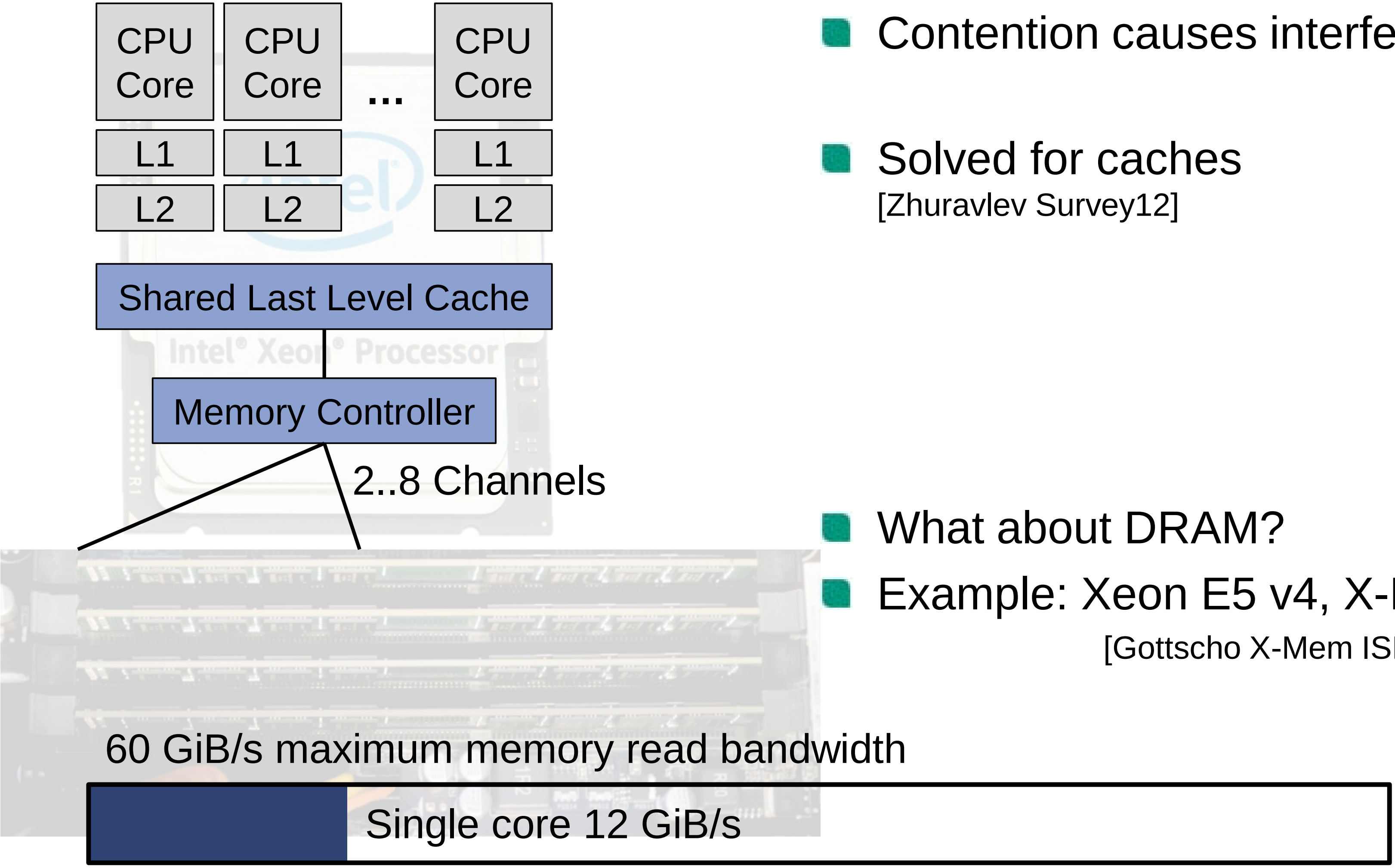
Software-Defined Physical Memory
Putting the OS in Control of DRAM

Marius Hillenbrand | Frank Bellosa

GI Fachgruppe Betriebssysteme – Frühjahrstreffen 2017

OPERATING SYSTEMS GROUP, DEPARTMENT OF INFORMATICS

Shared Resources in Multicore Processors



- Contention causes interference

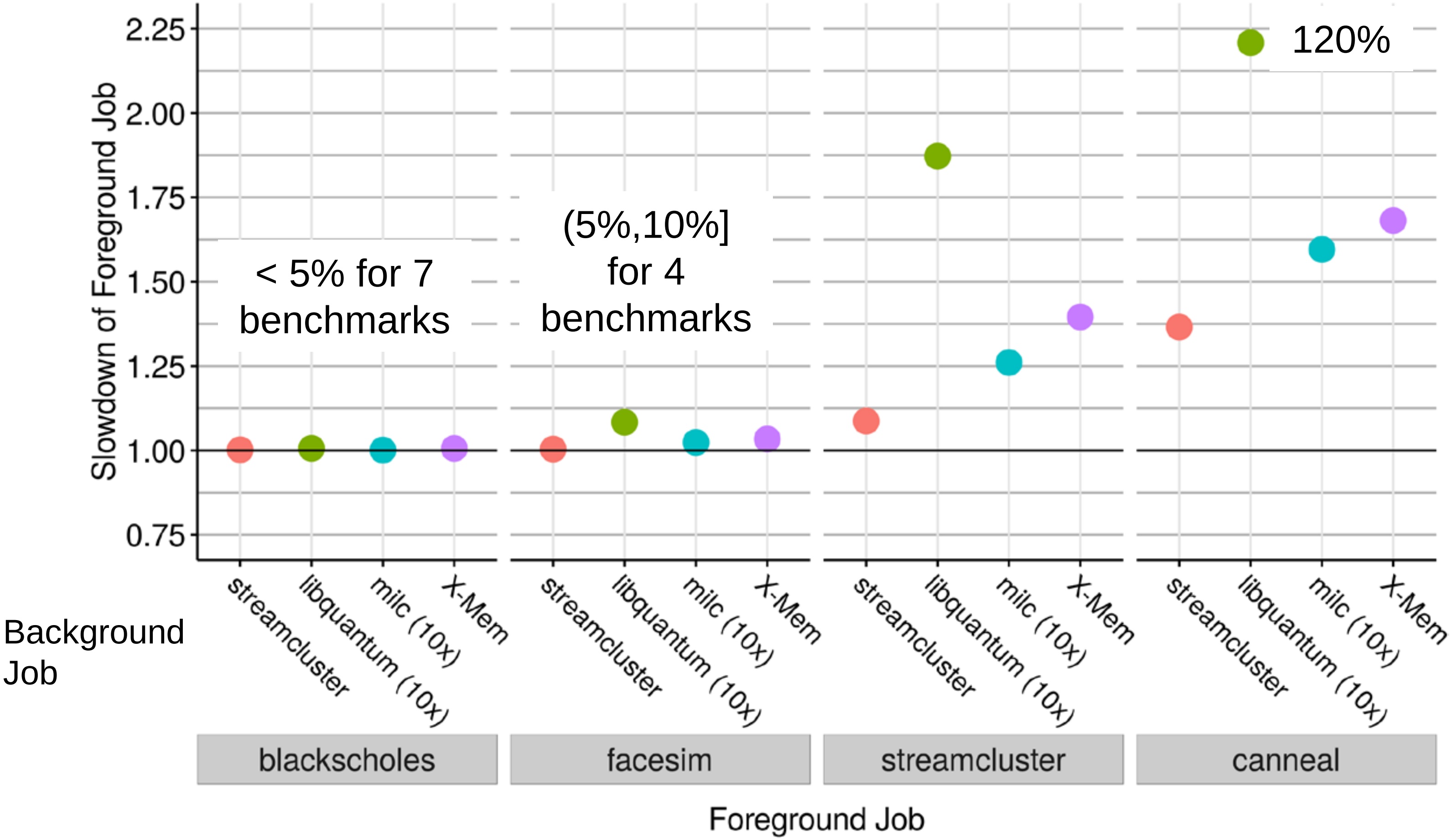
- Solved for caches
  [Zhuravlev Survey12]

- What about DRAM?
- Example: Xeon E5 v4, X-Mem
  [Gottscho X-Mem ISPASS16]

CPU: Intel Press Kit, DRAM: own

Experiment: DRAM Interference

- Partition caches and cores to expose memory interference



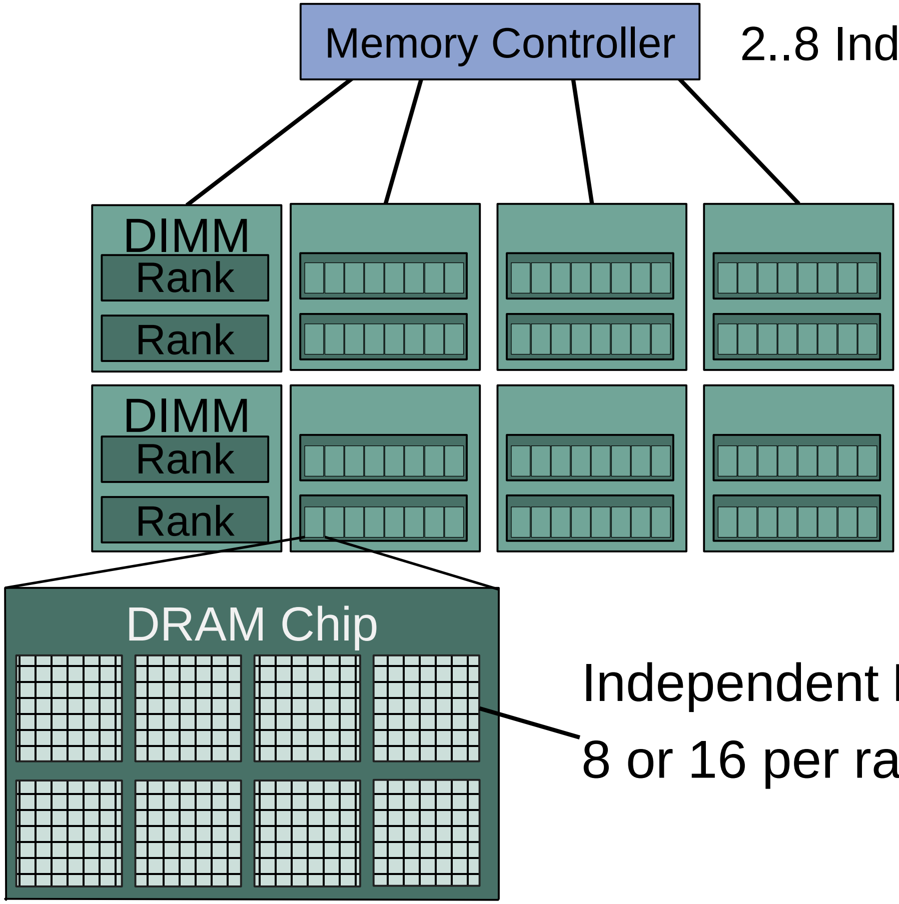
< 5% for 7 benchmarks

(5%,10%] for 4 benchmarks

120%

➢ Magnitude of slowdown is unpredictable

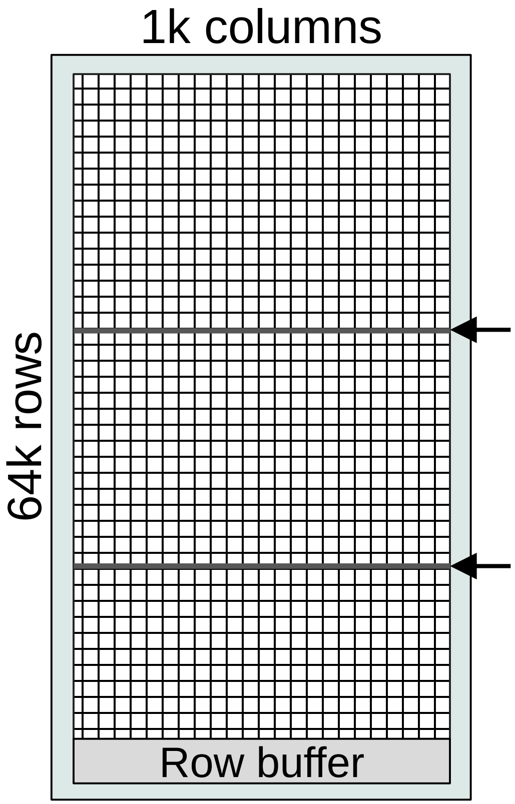
DRAM Parallelism [Jacob Memory07]



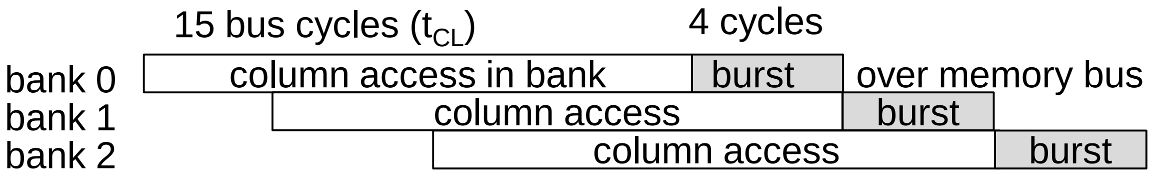
2..8 Independent **Channels**

Independent **Banks**
8 or 16 per rank

DRAM Operation & Interference



- Row hit



15 bus cycles (t_{CL})

4 cycles

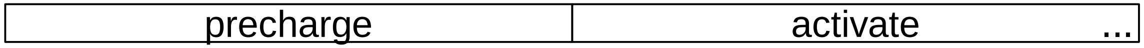
bank 0 column access in bank burst over memory bus

bank 1 column access burst

bank 2 column access burst

  - ➢ Want parallelism for performance

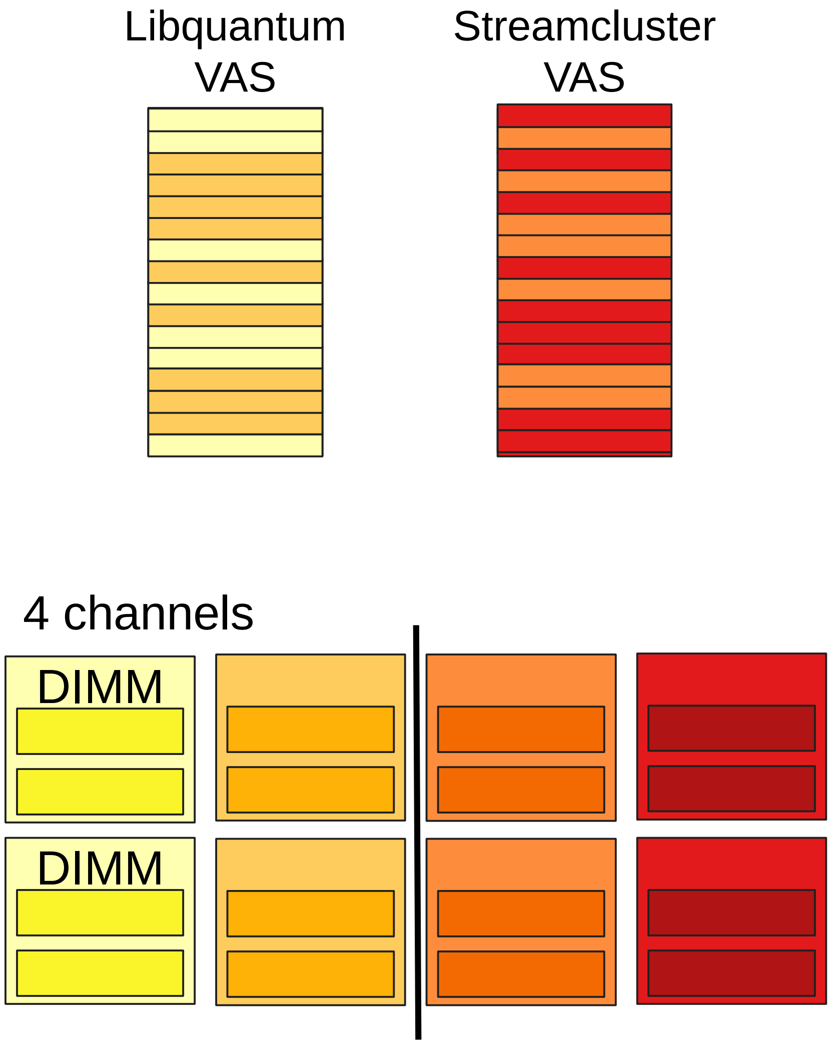
- Row miss – cycle to other row



precharge activate ...

  - ~3x latency
  - ➢ Sharing reduces locality, induces slowdown

Mitigation: Partitioning

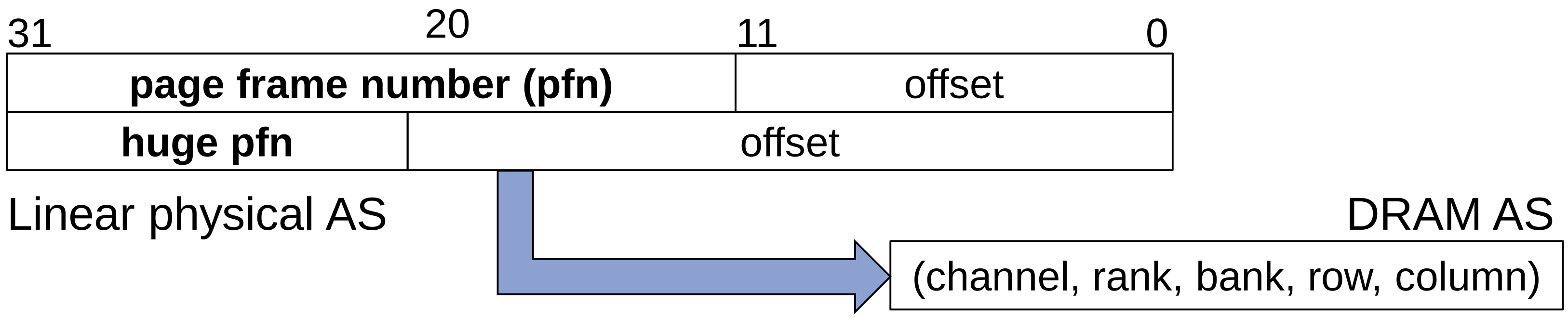


- Page placement is long-term scheduling
  - Permission to send read/write requests to DRAM banks/channels

- Partitioning
  - Page coloring [Liedtke CacheRT97]
  - Channels [Muralidhara Chan11]
  - Banks [Liu BPM14]
  - Control parallelism
  - Isolation maintains locality

Partitioning – DRAM Address Mapping

- OS page placement assigns channels and banks



31 20 11 0

page frame number (pfn) | offset

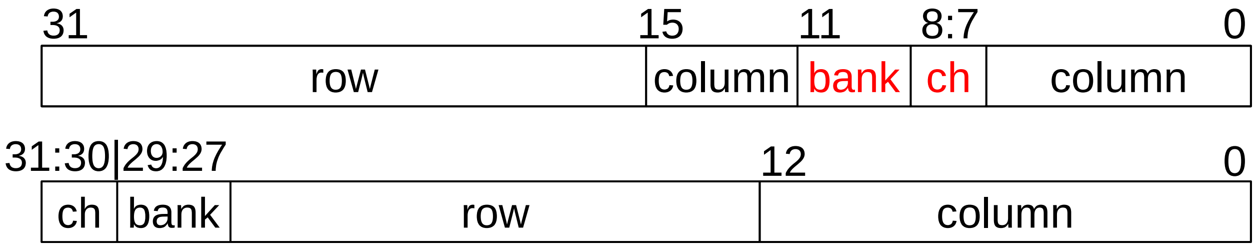
huge pfn | offset

Linear physical AS

DRAM AS

(channel, rank, bank, row, column)

- DRAM address mapping scheme [Jacob Memory07, 13.3]
  - Configured at boot time



31 15 11 8:7 0

row | column | bank | ch | column

31:30|29:27 12 0

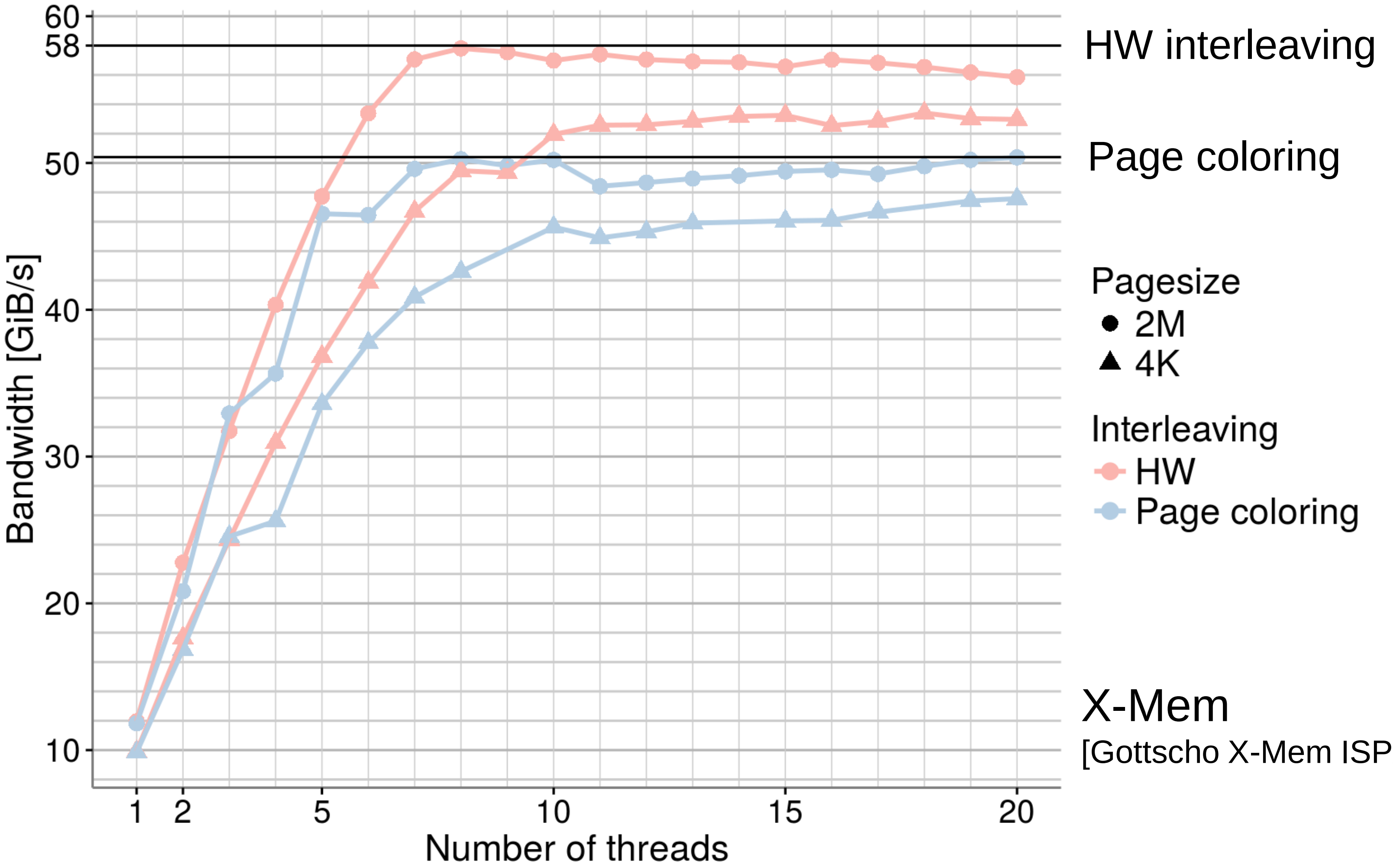
ch | bank | row | column

channel & bank interleaving

Non-interleaved (+page coloring)

➢ Need to reconfigure address mapping to enable partitioning (BIOS setup)

Slowdown from Partitioning



X-Mem
[Gottscho X-Mem ISPASS16]

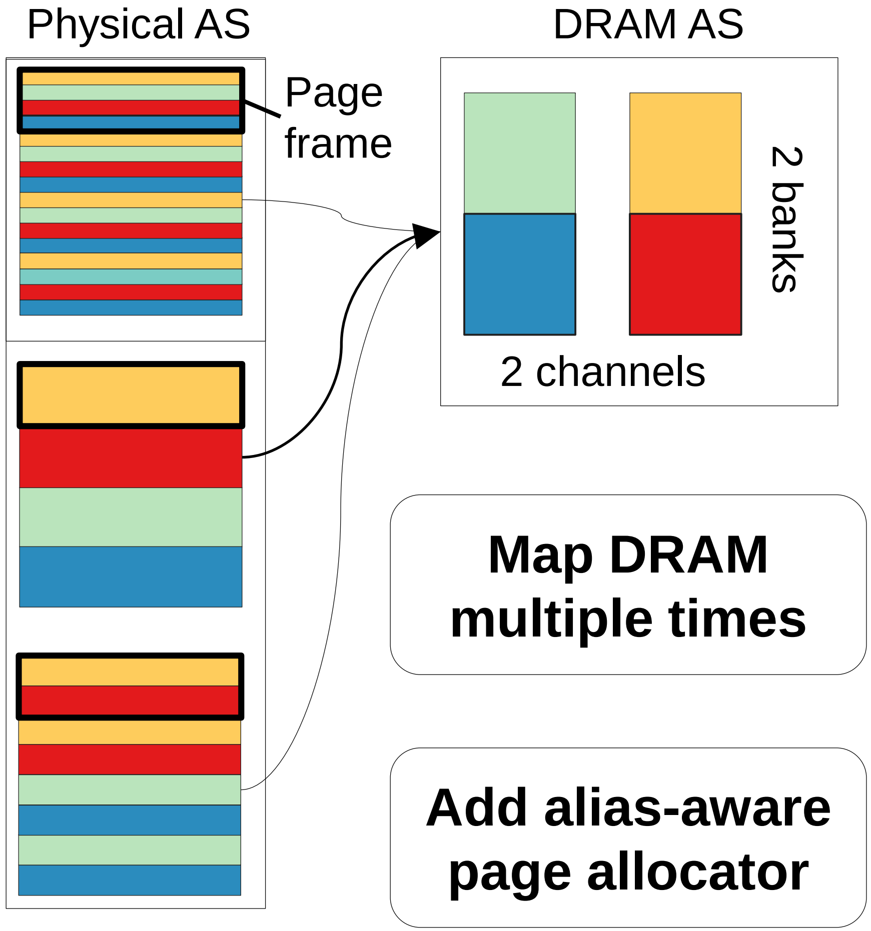
- ➢ Avoid slowdown when not needed, thus reconfigure at run time

DRAM Address Mapping Aliases

- Bank + channel interleaving
  - Max parallelism
  - No isolation

- Linear
  - Channel and bank partitioning
  - Minimum parallelism

- Bank interleaving
  - Channel partitioning
  - Bank parallelism



Map DRAM multiple times

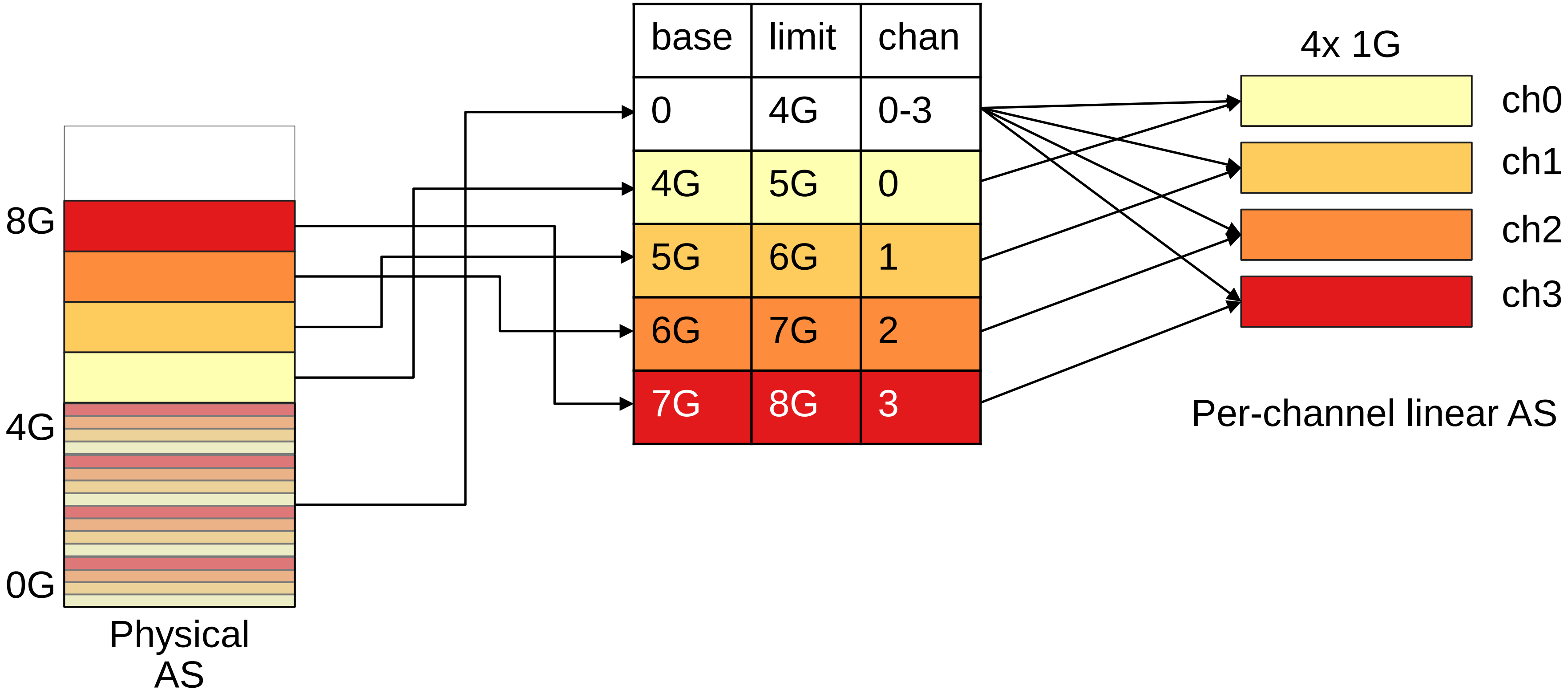
Add alias-aware page allocator

➢ Dynamically choose performance or isolation at run time

Channel Mapping Aliases

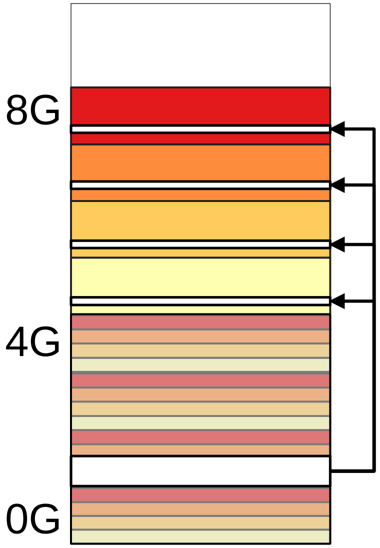
- Reconfigurable address translation (Intel Xeon, AMD Athlon/Opteron)

[SongPIkit16] [Xeon7500] [AMD15h30h]



base	limit	chan
0	4G	0-3
4G	5G	0
5G	6G	1
6G	7G	2
7G	8G	3

Alias-Aware Memory Management

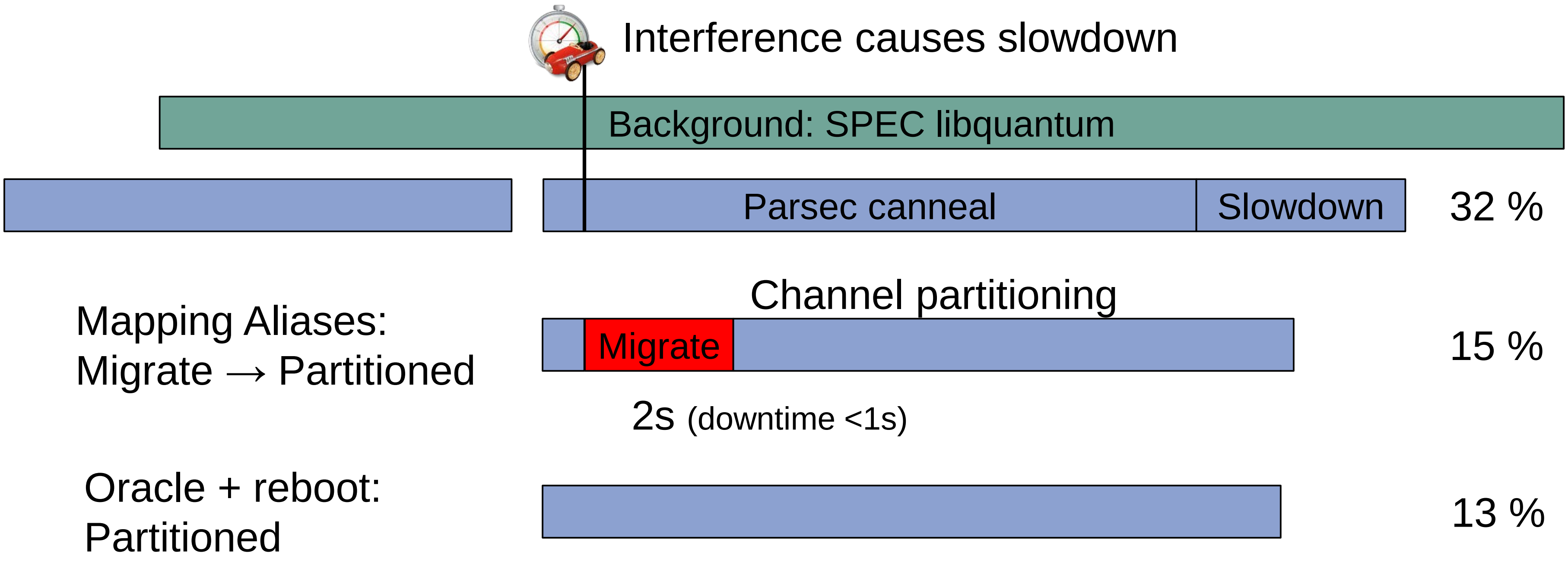


- Page coloring
  - Large regions
  - Utilize NUMA support in OS
- Binding processes to mapping scheme and channel
  - Aliases and channels are ~NUMA memory nodes
- Avoiding conflicts
  - Same DRAM behind corresponding physical regions
  - Memory hotplugging sets conflicting regions offline
- Migrating processes
  - NUMA memory policy and page migration
  - Cache coherence

Evaluation: On-Demand Partitioning

- Scenario: Compute cluster node
- Interleaved address mapping

AMD Athlon X4 880K
Linux 4.4.36
Cache & core partitioning



Interference causes slowdown

Background: SPEC libquantum

Parsec canneal | Slowdown — 32 %

Channel partitioning

Mapping Aliases:
Migrate → Partitioned

Migrate — 15 %

2s (downtime <1s)

Oracle + reboot:
Partitioned

13 %

➢ Dynamic reconfiguration provides effective isolation and reduces slowdown

Conclusion

- DRAM performance interference
  - Slowdown depends on workload combination
  - Not known in advance

- Partitioning introduces unavoidable overhead
  - Disables interleaving
  - Reduces memory parallelism

- DRAM mapping aliases offer the OS a choice at runtime
  - Isolation or sharing
  - Integrated with memory management

  - Performance of interleaved address mapping
  - On-demand partitioning

References

[Zhuravlev Survey12] Sergey Zhuravlev, Juan Carlos Saez, Sergey Blagodurov, Alexandra Fedorova, and Manuel Prieto. *Survey of Scheduling Techniques for Addressing Shared Resources in Multicore Processors*. ACM Computing Surveys 45, 1, Article 4 (December 2012)

[X-Mem ISPASS16] Mark Gottscho, Sriram Govindan, Bikash Sharma, Mohammed Shoaib, and Puneet Gupta*. X-Mem: A Cross-Platform and Extensible Memory Characterization Tool for the Cloud*. IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS). Uppsala, Sweden. April 2016

[Intel SDM] Intel Corp. *Intel 64 and IA-32 Architectures Software Developer's Manual*, September 2016, Order Number 325462-060US

[Jacob Memory07] Bruce Jacob, Spencer Ng, and David Wang. *Memory Systems: Cache, Dram, Disk*. 2007. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[Rixner Sched00] Scott Rixner, William J. Dally, Ujval J. Kapasi, Peter Mattson, and John D. Owens. *Memory Access Scheduling*. In Proceedings of the 27th Annual International Symposium on Computer Architecture (ISCA '00). ACM, 2000

[Mutlu Survey14] Onur Mutlu and Lavanya Subramanian. *Research Problems and Opportunities in Memory Systems.* Supercomputing Froniers and Innovations: an International Journal 1, 3 2014

References (2)

[PesslDRAMA16] Peter Pessl, Daniel Gruss, Clémentine Maurice, Michael Schwarz, Stefan Mangard. *DRAMA: Exploiting DRAM Addressing for Cross-CPU Attacks.* 25th Usenix Security Symposium, 2016

[SongPIkit16] Wonjun Song, Hyunwoo Choi, Junhong Kim, Eunsoo Kim, Yongdae Kim, and John Kim. PIkit: A New Kernel-Independent Processor-Interconnect Rootkit. 25th Usenix Security Symposium, 2016

[Xeon7500] Intel Corp. *Intel Xeon Processor 7500 Series Datasheet,* Volume 2, March 2010

[AMD15h30h] Advanced Micro Devices, Inc. *BIOS and Kernel Developer's Guide (BKDG) for AMD Family 15h Models 30h-3Fh Processors,* 49125 Rev 3.06 - February 10, 2015

[Liedtke CacheRT97] Jochen Liedtke, Hermann Haertig, and Michael Hohmuth. *OS-Controlled Cache Predictability for Real-Time Systems*. In Proceedings of the 3rd IEEE Real-Time Technology and Applications Symposium (RTAS '97). IEEE Computer Society, 1997

References (3)

[Ebrahimi FST10] Eiman Ebrahimi, Chang Joo Lee, Onur Mutlu, and Yale N. Patt. *Fairness via Source Throttling: a Configurable and High-Performance Fairness Substrate for Multi-Core Memory Systems.* In Proceedings of the Fifteenth Edition of ASPLOS on Architectural Support for Programming Languages and Operating Systems (ASPLOS XV). ACM, 2010

[Yun MemG13] Heechul Yun, Yao Gang, Rodolfo Pellizzoni, Marco Caccamo, and Lui Sha, *MemGuard: Memory Bandwidth Reservation System for Efficient Performance Isolation in Multi-core Platforms*, IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS), April, 2013

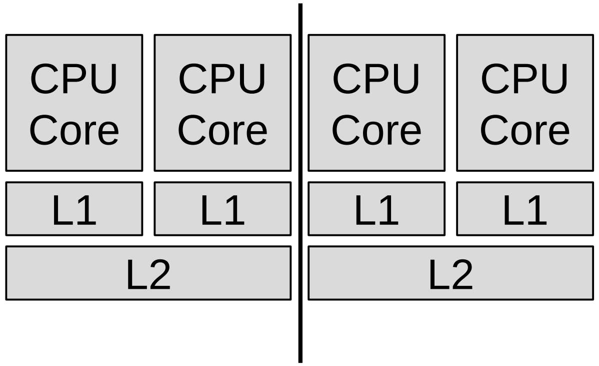
[Zhang HWET09] Xiao Zhang, Sandhya Dwarkadas, and Kai Shen. *Hardware Execution Throttling for Multi-Core Resource Management*. In Proceedings of the 2009 USENIX Annual Technical Conference (ATC'09). USENIX Association, 2009

[Muralidhara Chan11] Sai Prashanth Muralidhara, Lavanya Subramanian, Onur Mutlu, Mahmut Kandemir, and Thomas Moscibroda. *Reducing Memory Interference in Multicore Systems via Application-Aware Memory Channel Partitioning*. In Proceedings of the 44th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO-44). ACM, 2011

[Liu BPM14] Lei Liu, Zehan Cui, Yong Li, Yungang Bao, Mingyu Chen, and Chengyong Wu. *BPM/BPM+: Software-based Dynamic Memory Partitioning Mechanisms for Mitigating DRAM Bank-/Channel-Level Interferences in Multicore Systems*. *ACM* Transactions on Architecture and Code Optimization (TACO) 11, 1, Article 5 (February 2014)

Evaluation Setup

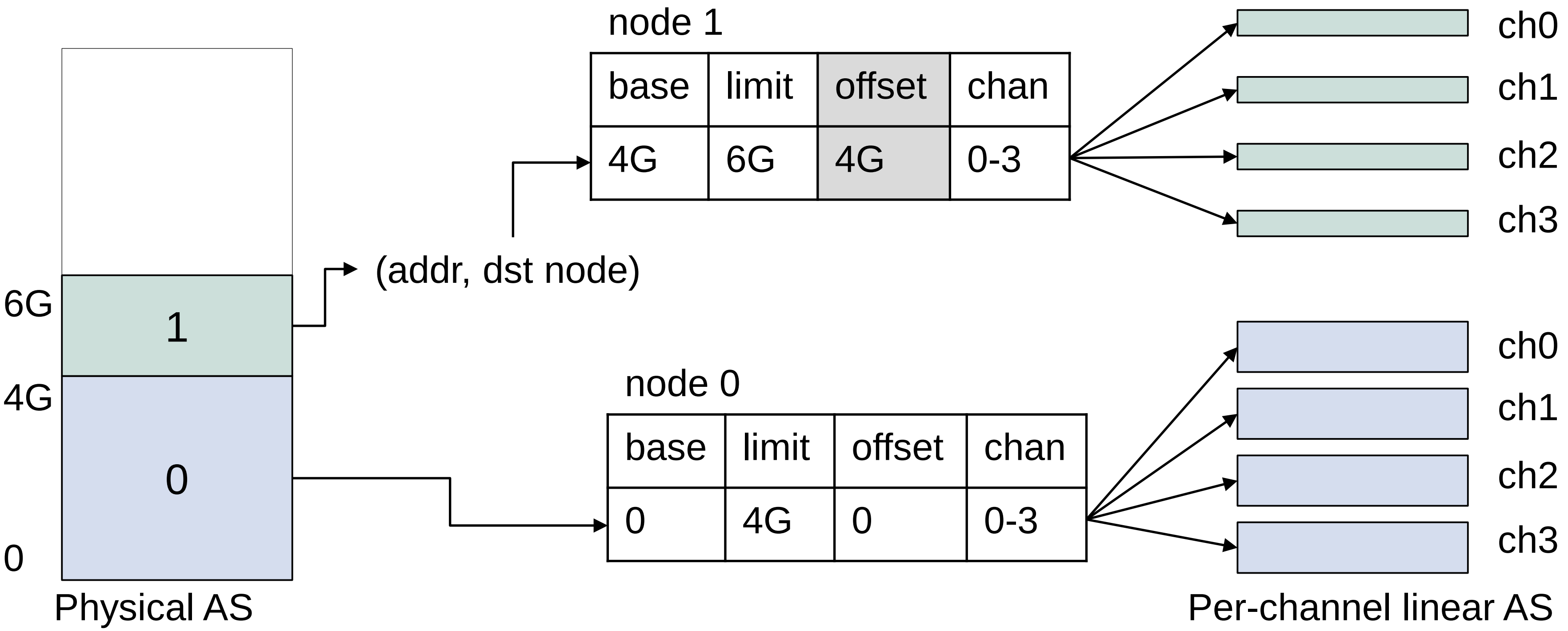
- AMD Athlon X4 880K *Steamroller* [AMD15h30h]
- 32 GiB dual-channel DDR3 DRAM
  - Channel-interleaved alias
  - Linear alias
- Linux 4.4.36 + modifications
- Core and cache partitioning



Implementation: Conventional Mapping

- 3-stage address translation (Intel Xeon, AMD Athlon/Opteron)
  1. Source NUMA routing
  2. Target address decoder
  3. (DRAM address decoder)

[SongPIkit16] [Xeon7500] [AMD15h30h]



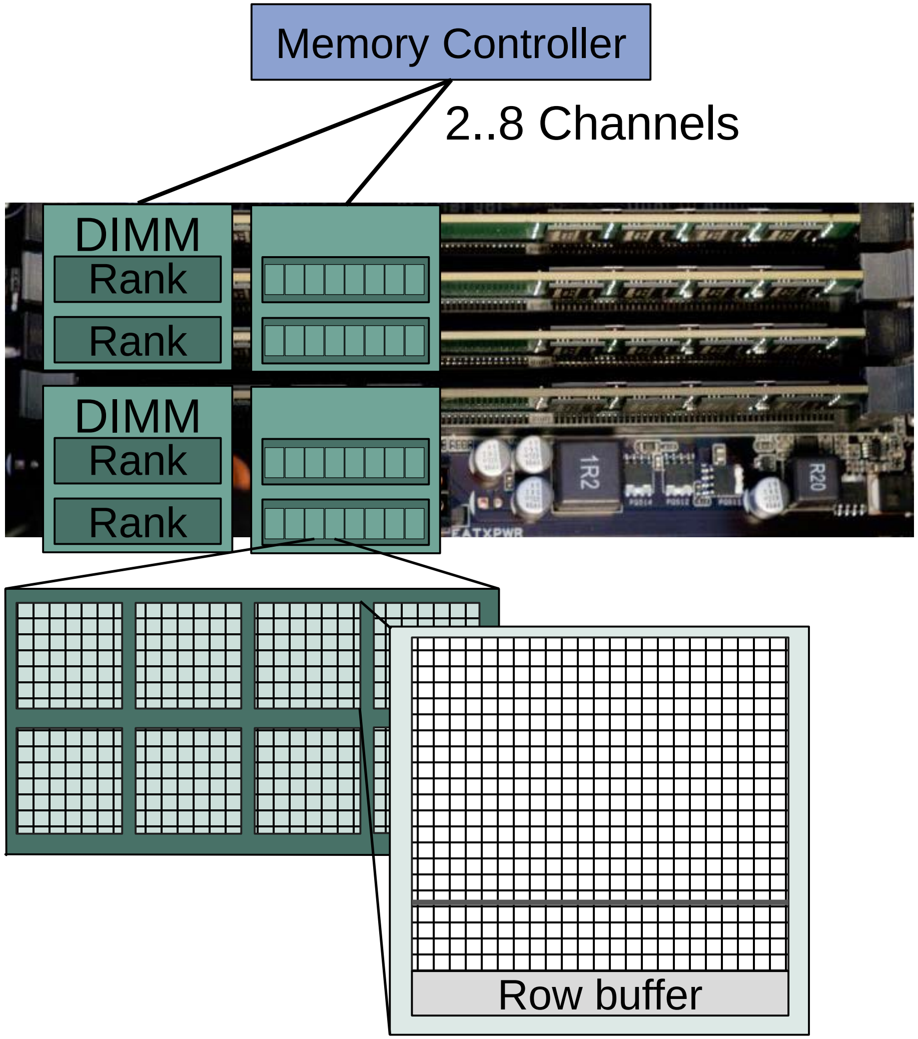
node 1

base	limit	offset	chan
4G	6G	4G	0-3

node 0

base	limit	offset	chan
0	4G	0	0-3

DRAM Structure



- Hierarchy of parallel resources
- Memory Channel
  - Command & address / data bus
  - Set of DIMMs
- Rank
  - Set of chips (8/9)
  - Addressed as a unit
  - 1-2 per DIMM
- Bank
  - 2-dimensional DRAM array
  - 8/16 per rank

[Jacob Memory07]

➢ Memory parallelism from independent banks and channels