Thoughts on Memory

Prof. Bruce Jacob
Keystone Professor & Director of Computer Engineering Program
Electrical & Computer Engineering
University of Maryland at College Park

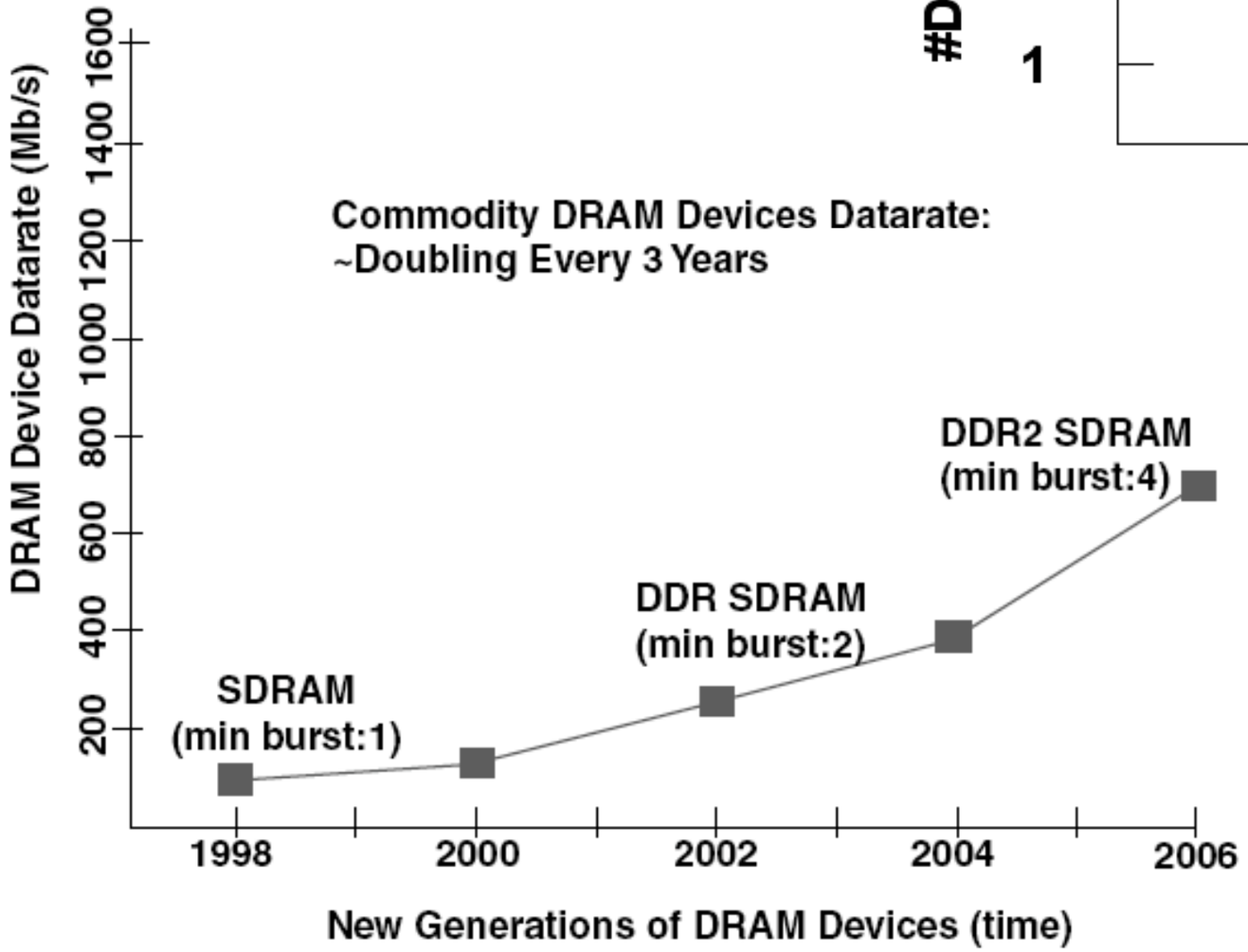
Main problem: We don’t understand it very well

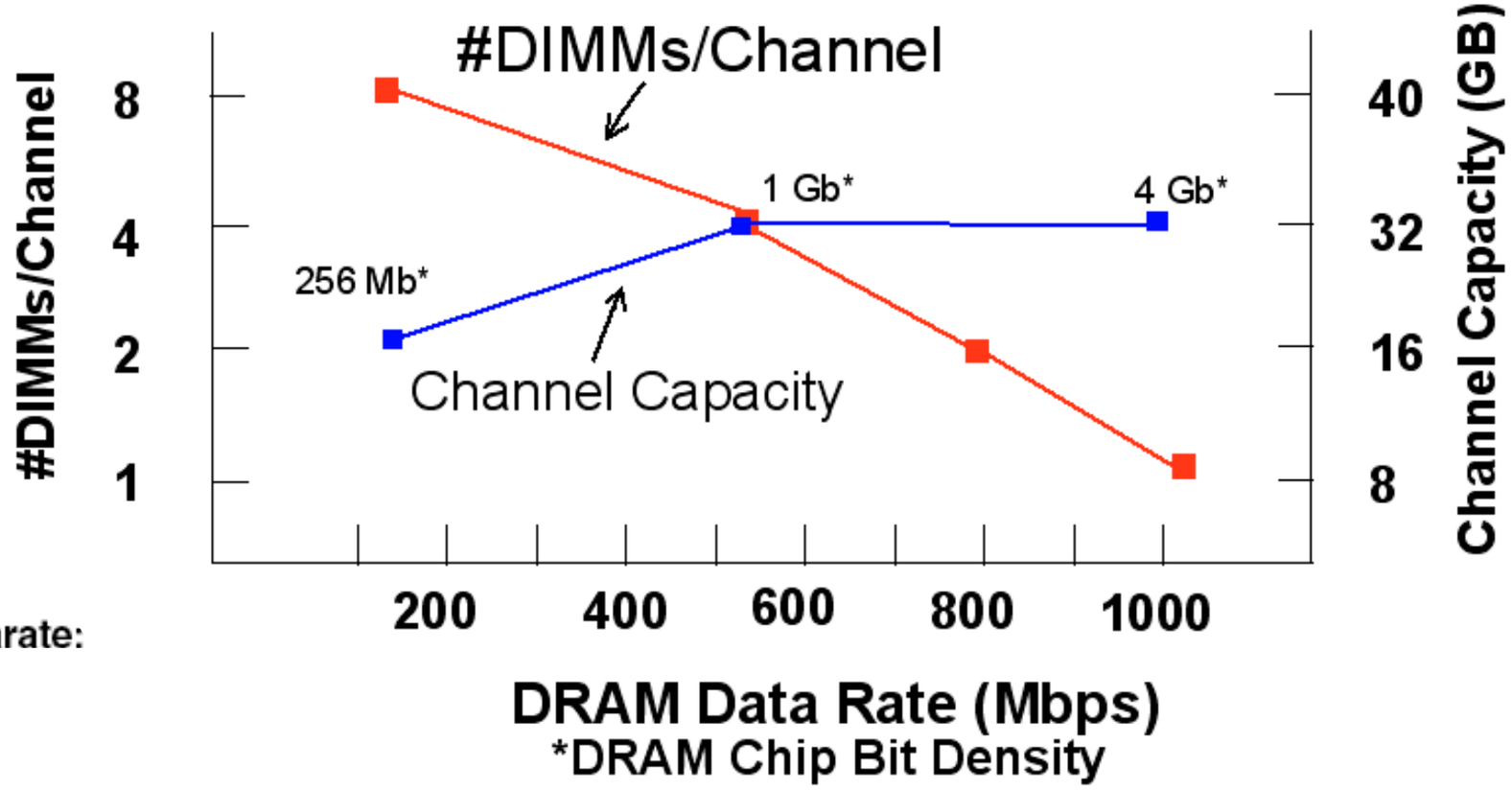
How it is represented

```
if (cache_miss(addr)) {
   cycle_count += DRAM_LATENCY;
}
```

even in simulators with “cycle accurate” memory systems—no lie

Problem: Capacity

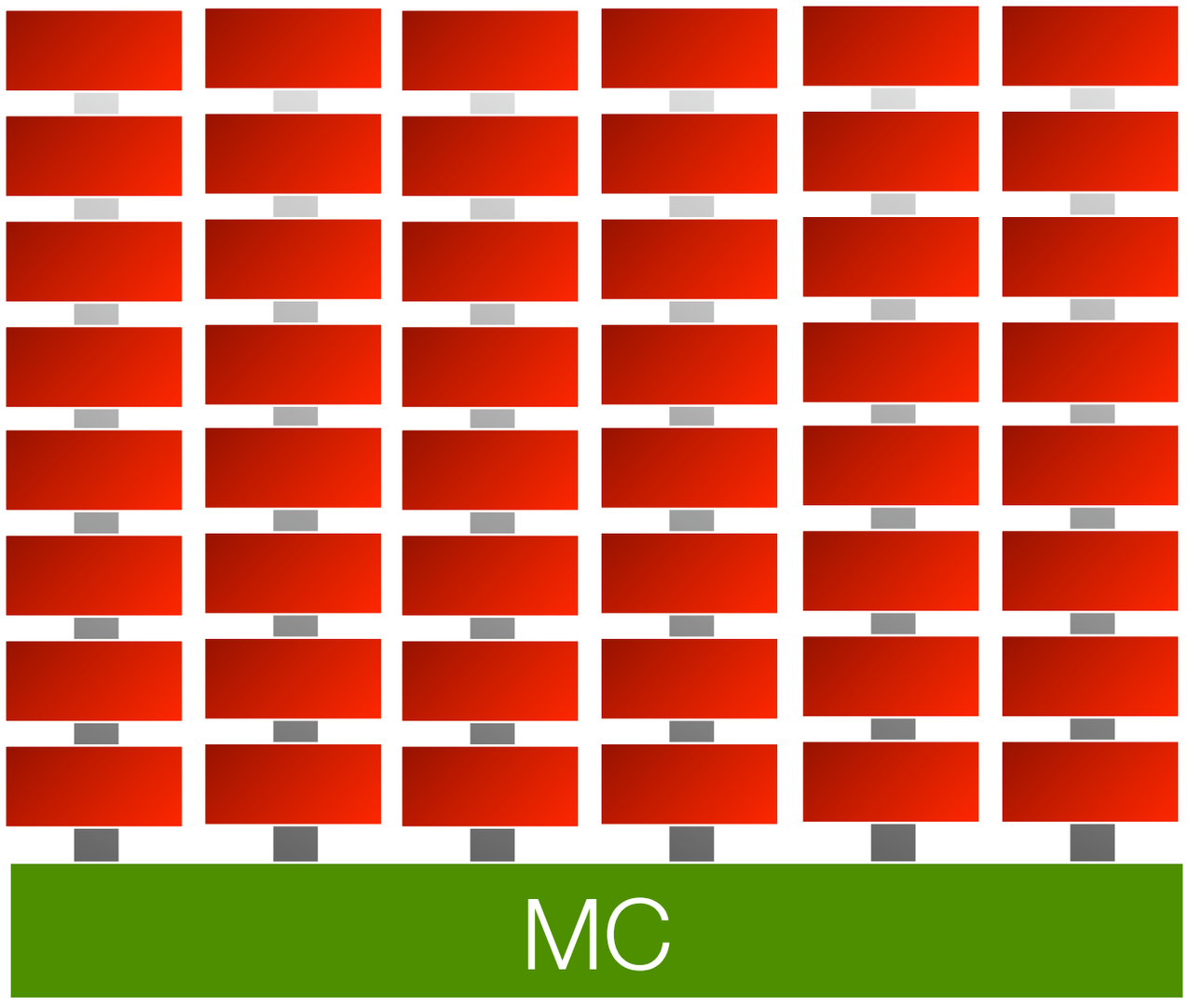




Problem: Capacity

JEDEC DDRx

~10W/DIMM, ~20W total



FB-DIMM

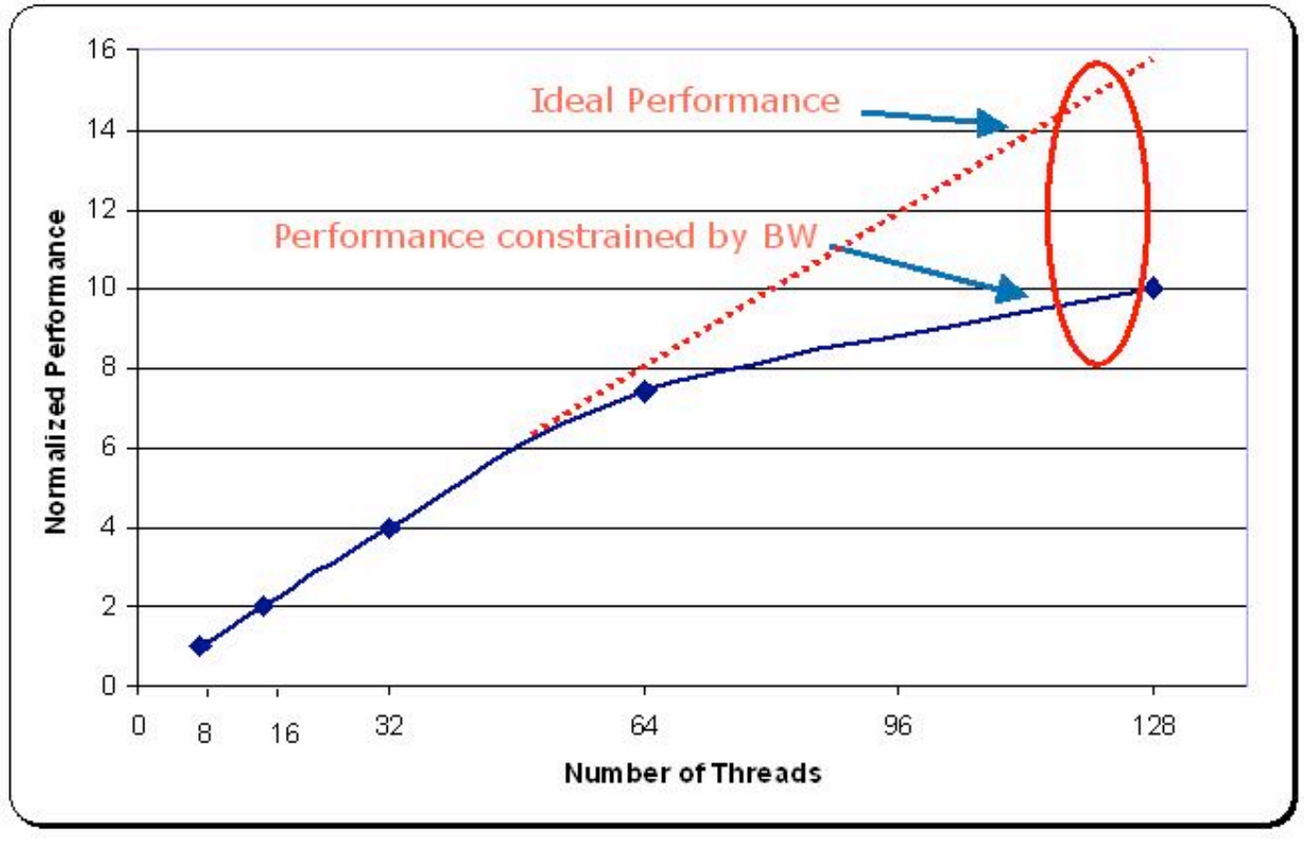
~10W/DIMM, ~300W total

Problem: Bandwidth

- Like capacity, primarily a power and heat issue: can get more BW by adding busses, but they need to be narrow & thus fast. Fast = hot.

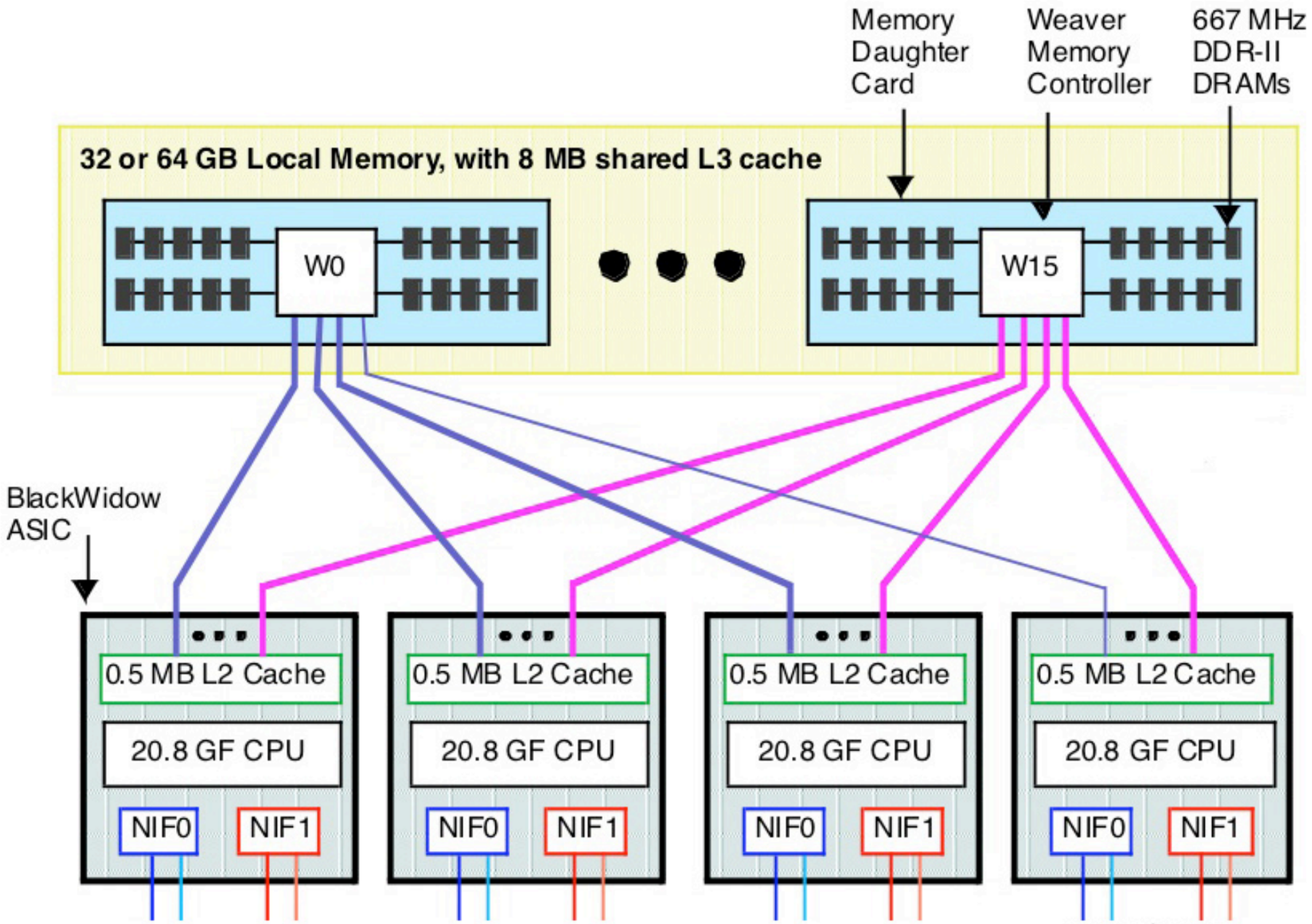
- Required BW per core is roughly 1 GB/s, and cores per chip is increasing

- Graph: Thread-based load (SPECjbb), memory set to 52GB/s sustained
  … cf. 32-core Sun Niagara: saturates at 25.6 GB/s



Problem: Bandwidth

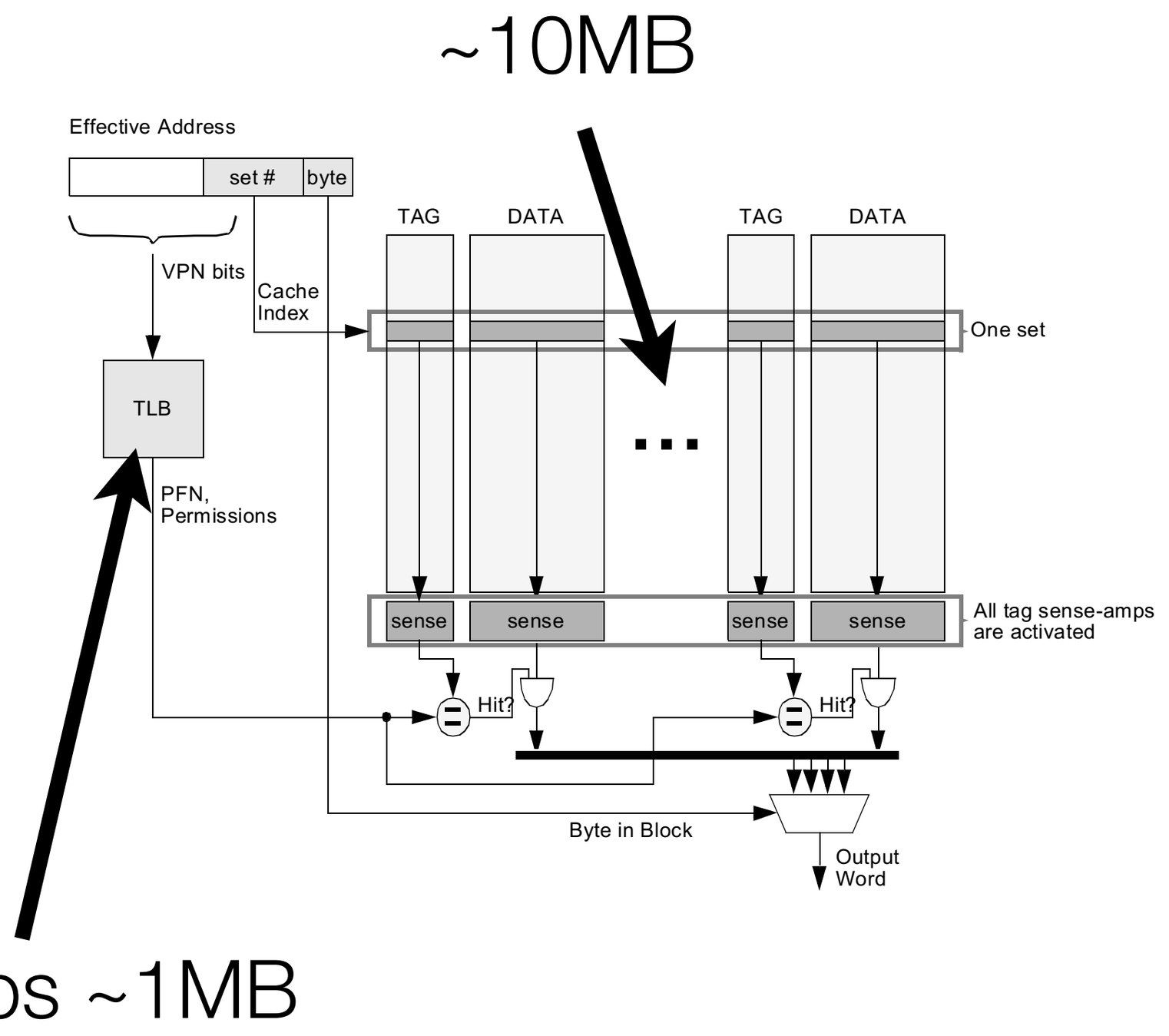
Sometimes bandwidth is everything ...



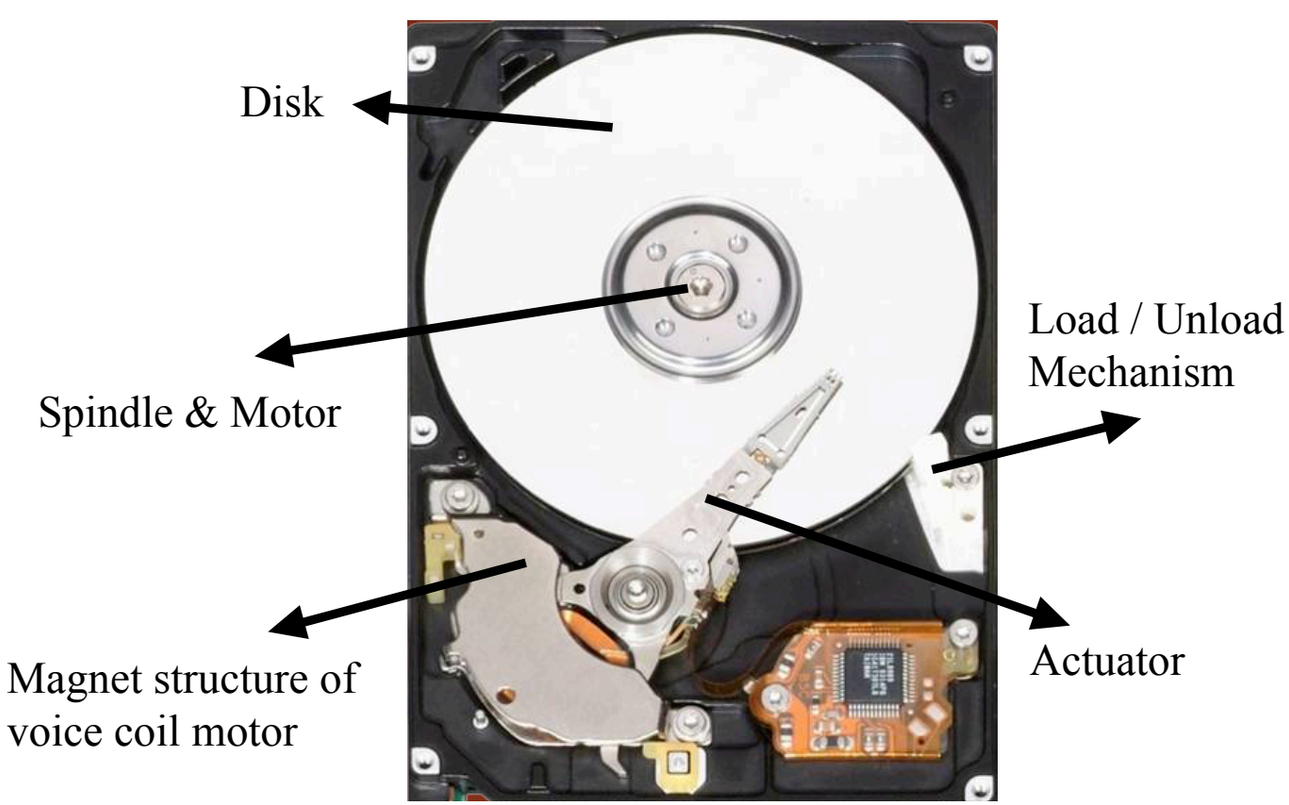
Cray Black Widow memory system

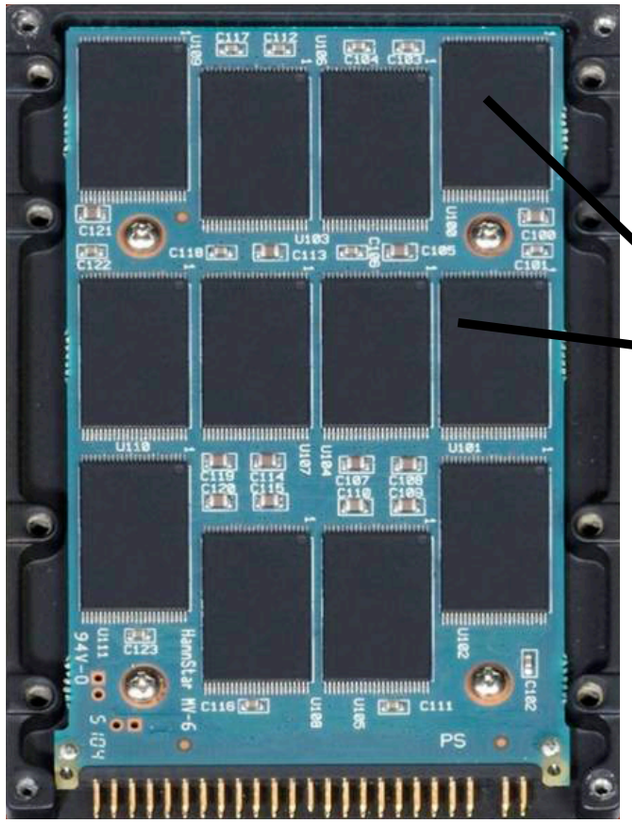
Problem: TLB Reach

- Doesn't scale at all (still small and not upgradeable)
- Currently accounts for **20+%** of system overhead
- Higher associativity (which offsets the TLB's small size) can create a power issue
- The TLB's "reach" is actually much worse than it looks, because of different access granularities



Trend: Disk, Flash, and other NV

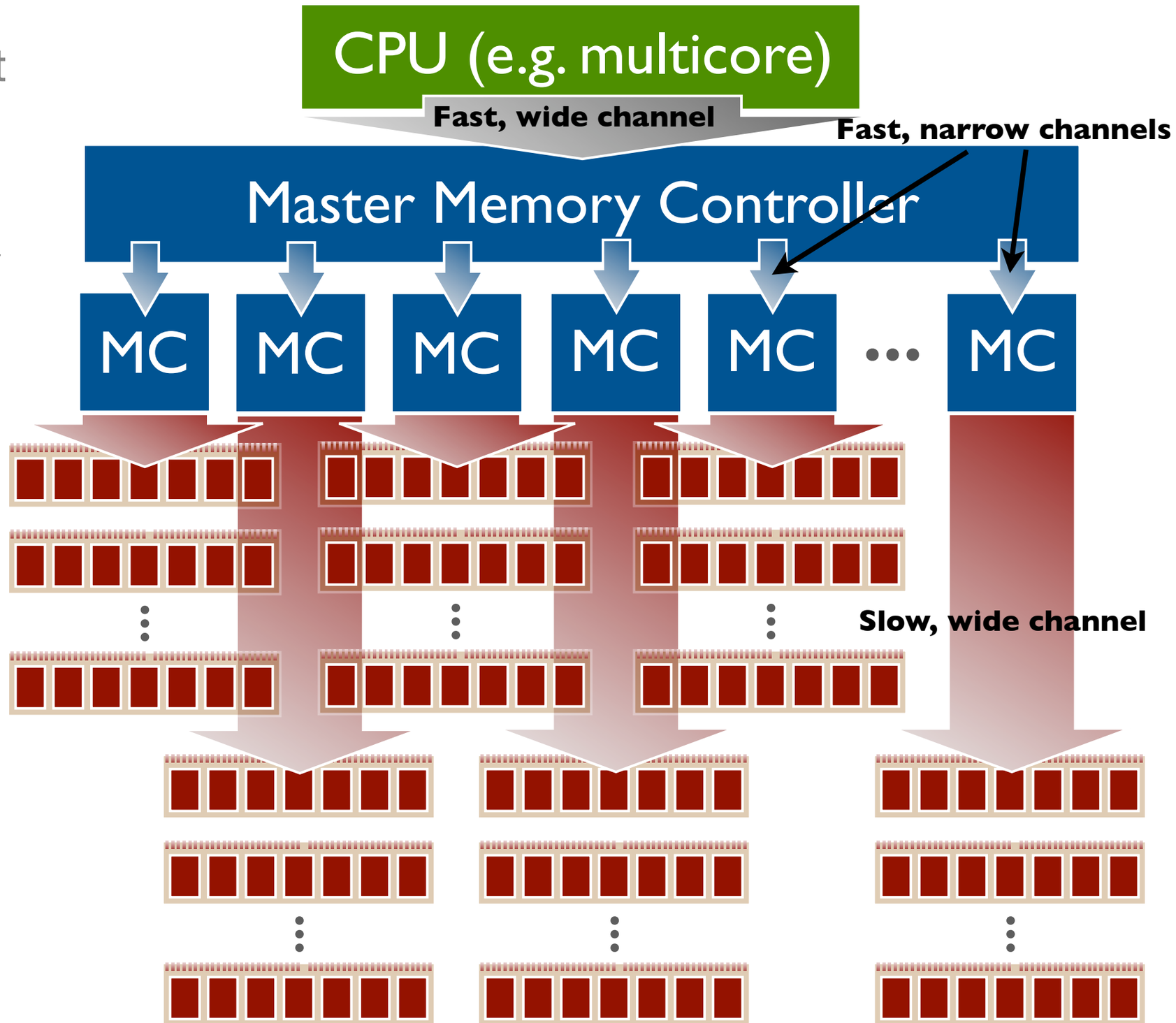




- Flash is currently eating Disk's lunch
- PCM is expected to eat Flash's lunch

Obvious Conclusions I

- Want capacity without sacrificing bandwidth
- **Need a new memory system architecture**
- This is coming (details will change, of course)



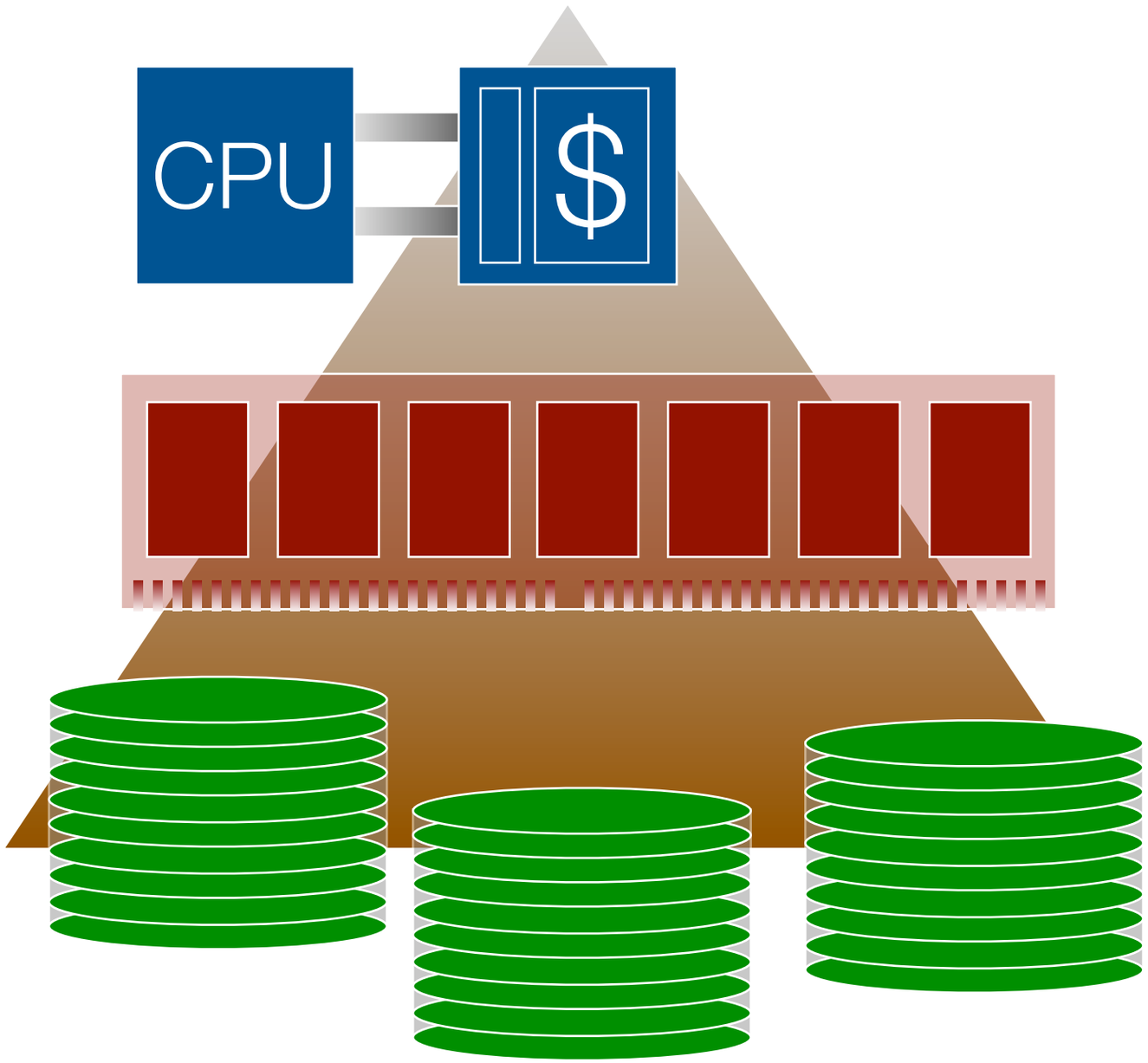
Obvious Conclusions II

- Flash/NV is inexpensive, is fast (rel. to disk), and has better capacity roadmap than DRAM

- **Make it a first-class citizen in the memory hierarchy**

- Access it via load/store interface, use DRAM to buffer writes, software management

- Probably reduces capacity pressure on DRAM system



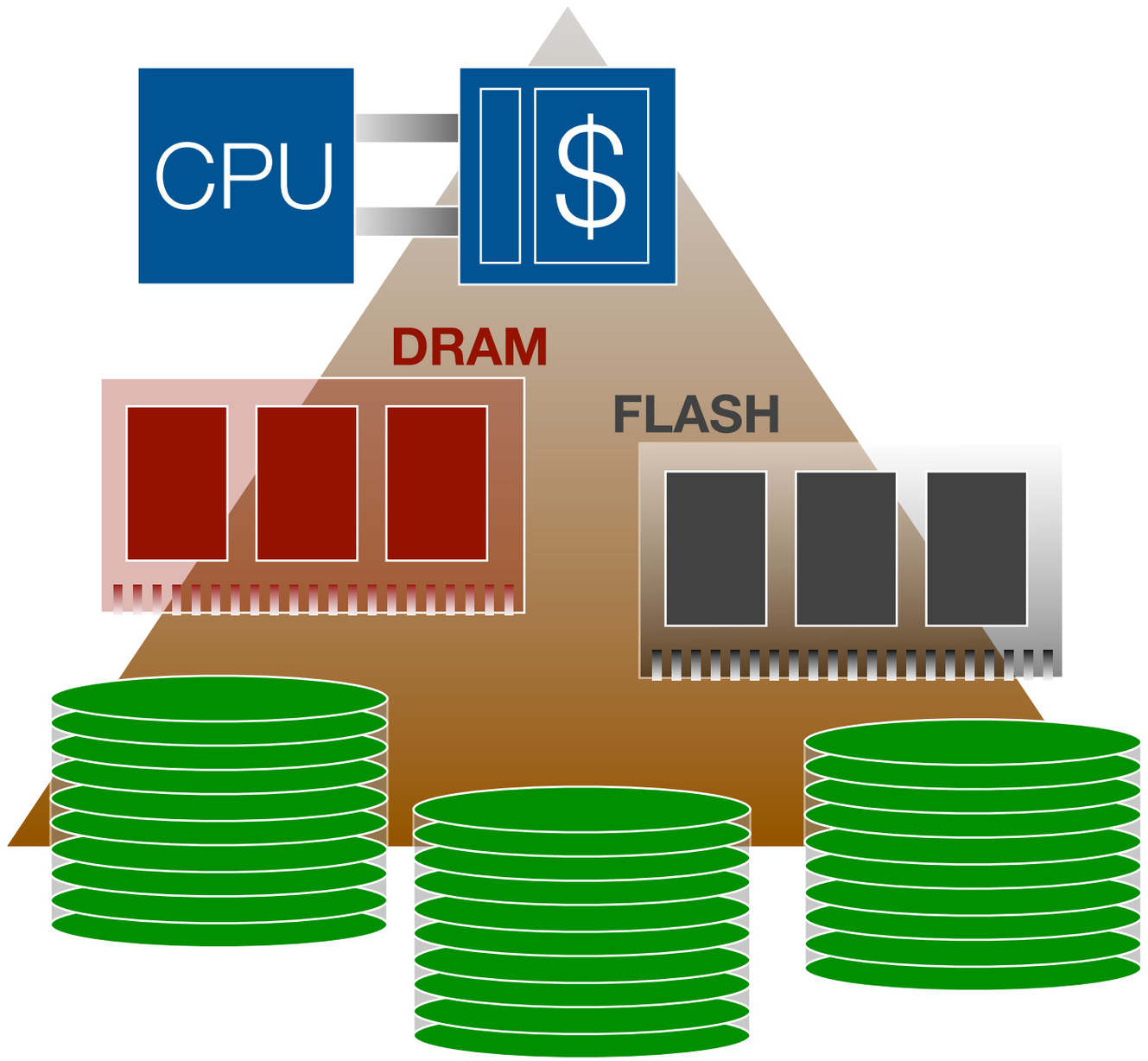
Obvious Conclusions II

- Flash/NV is inexpensive, is fast (rel. to disk), and has better capacity roadmap than DRAM

- **Make it a first-class citizen in the memory hierarchy**

- Access it via load/store interface, use DRAM to buffer writes, software management

- Probably reduces capacity pressure on DRAM system



Obvious Conclusions III

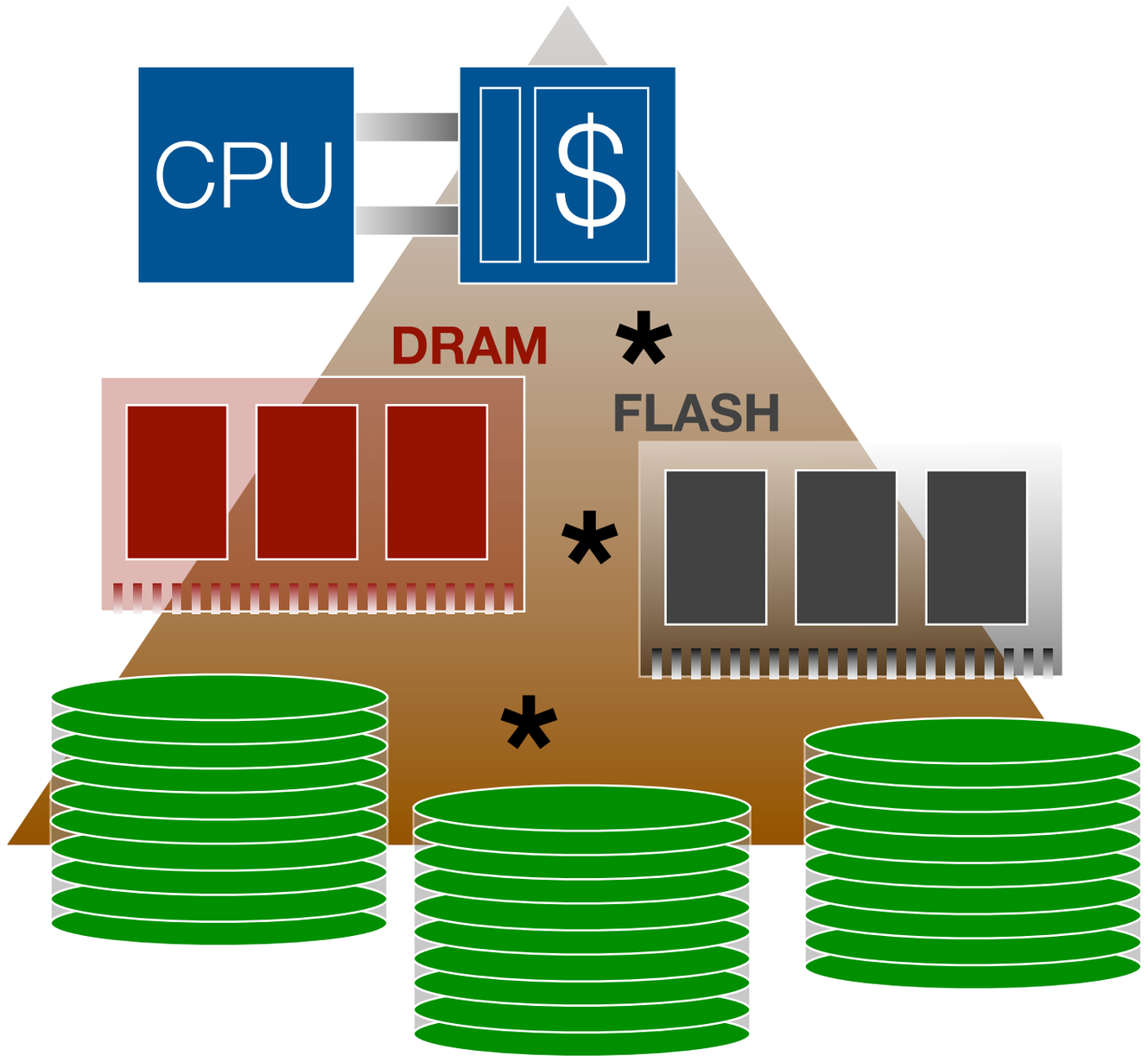
- Reduce translation overhead (both in performance and in power)

- **Need an OS/arch redesign**

- Revisit superpages, multi-level TLBs

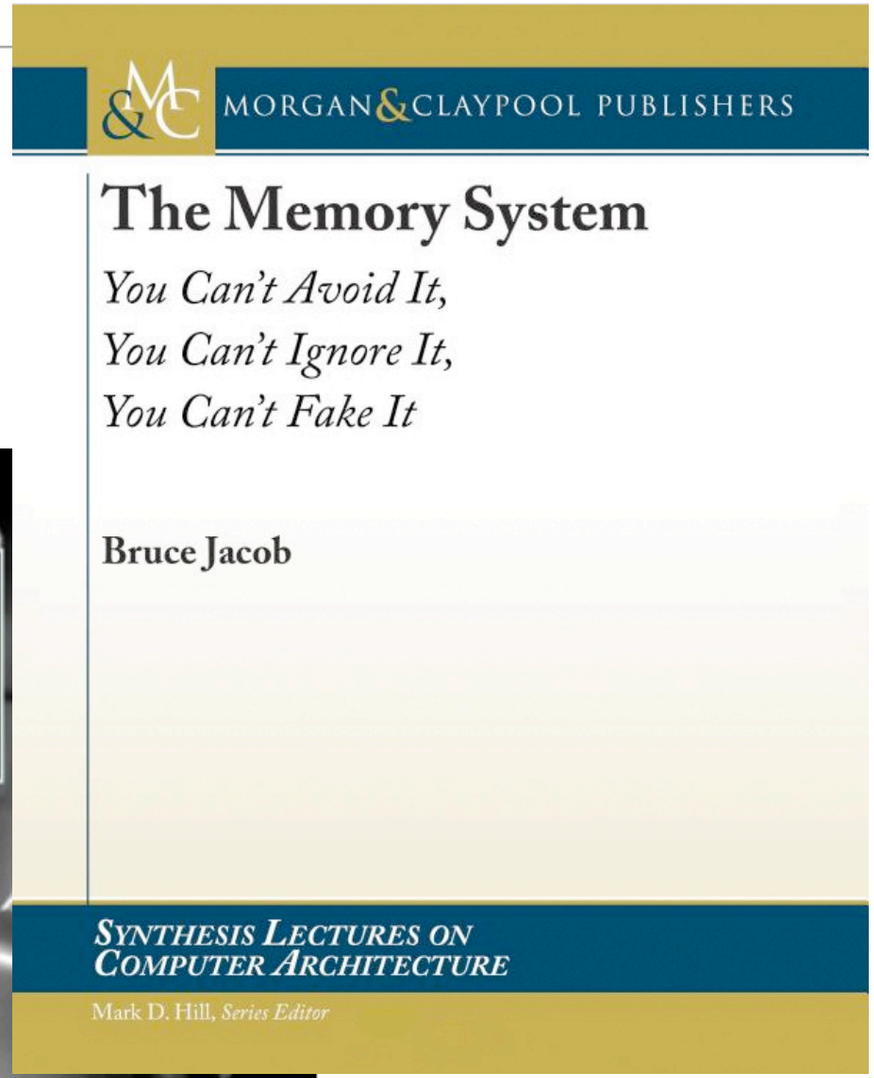
- Revisit SASOS concepts, *location of translation point/s*

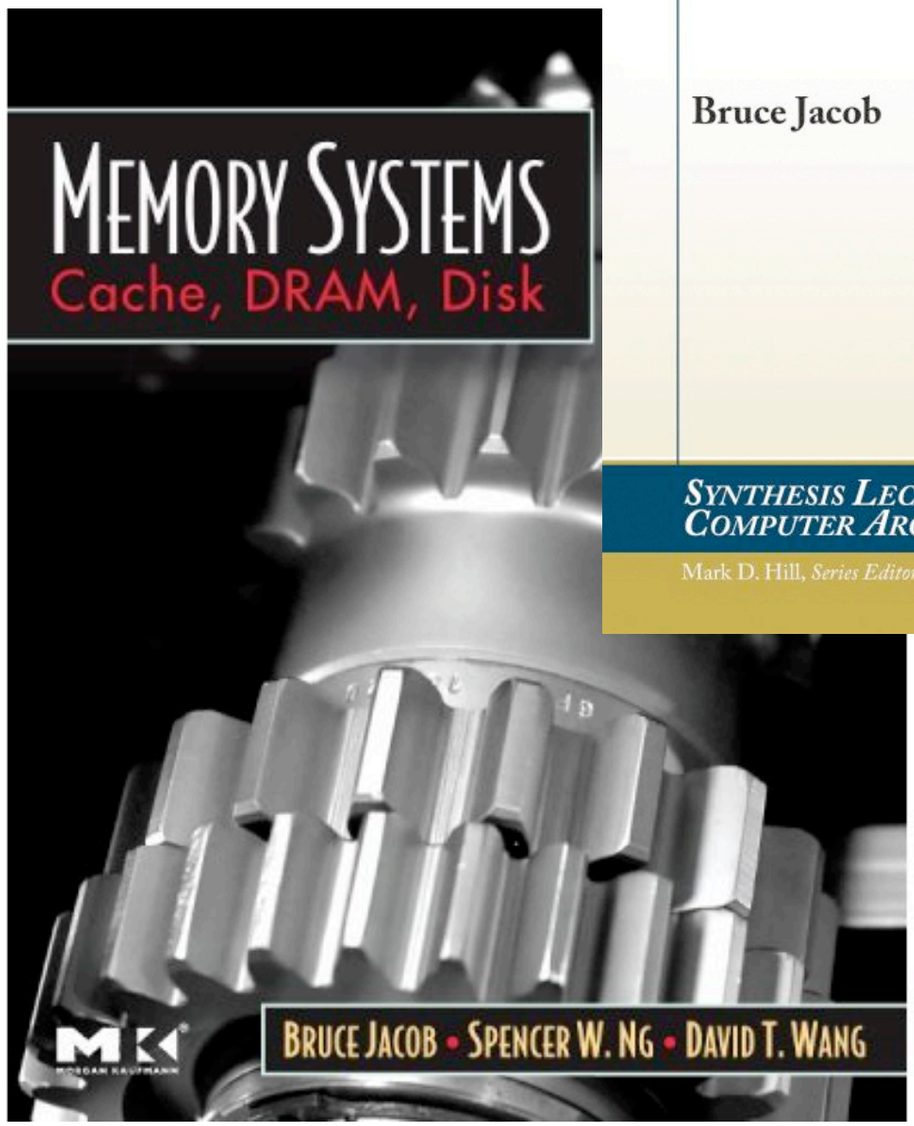
- Probably most suited for the high-end, at least initially



Acknowledgements & Shameless Plugs

- Much of this has appeared previously in our books, papers, etc.
  - The Memory System (You Can't Avoid It; You Can't Ignore It; You Can't Fake It). B. Jacob, with contributions by S. Srinivasan and D. T. Wang. ISBN 978-1598295870. Morgan & Claypool Publishers: San Rafael CA, 2009.
  - Memory Systems: Cache, DRAM, Disk. B. Jacob, S. Ng, and D. Wang, with contributions by S. Rodriguez. ISBN 978-0123797513. Morgan Kaufmann: San Francisco CA, 2007.
- Support from Intel, DoD, DOE, Sandia National Lab, Micron, Cypress Semiconductor





Questions?

(thank you for your kind indulgence)

Prof. Bruce Jacob
Keystone Professor & Director of Computer Engineering Program
Electrical & Computer Engineering
University of Maryland at College Park

blj@umd.edu
www.ece.umd.edu/~blj

… or just google “bruce jacob”