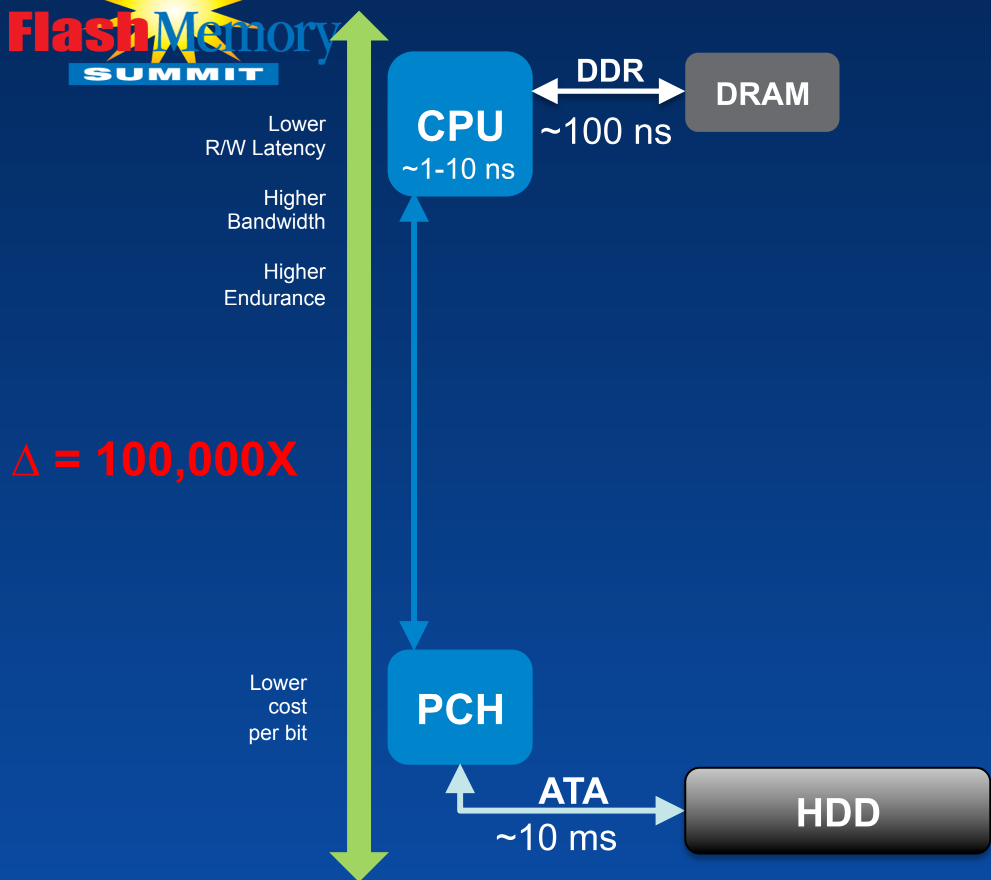
Building Datacenter Infrastructure Using Persistent Memory

Forum R-21 – Persistent Memory

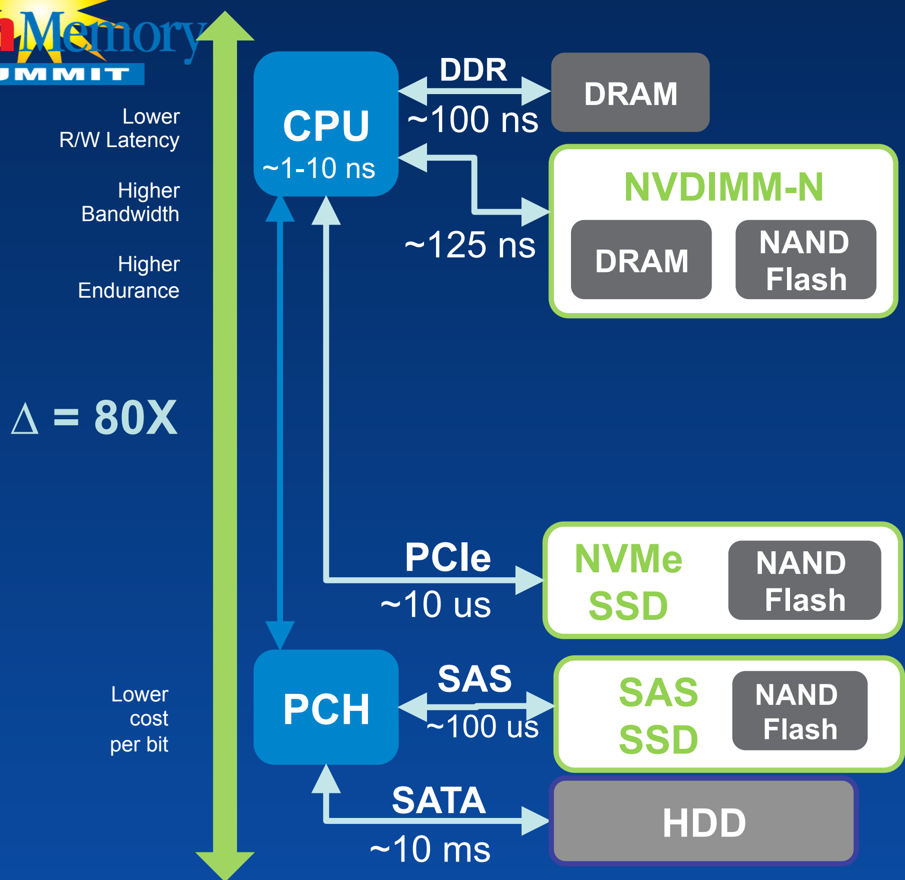
Rob Peglar
Senior VP & CTO
Symbolic IO

The Past: Nonvolatile Memories in Server Architectures



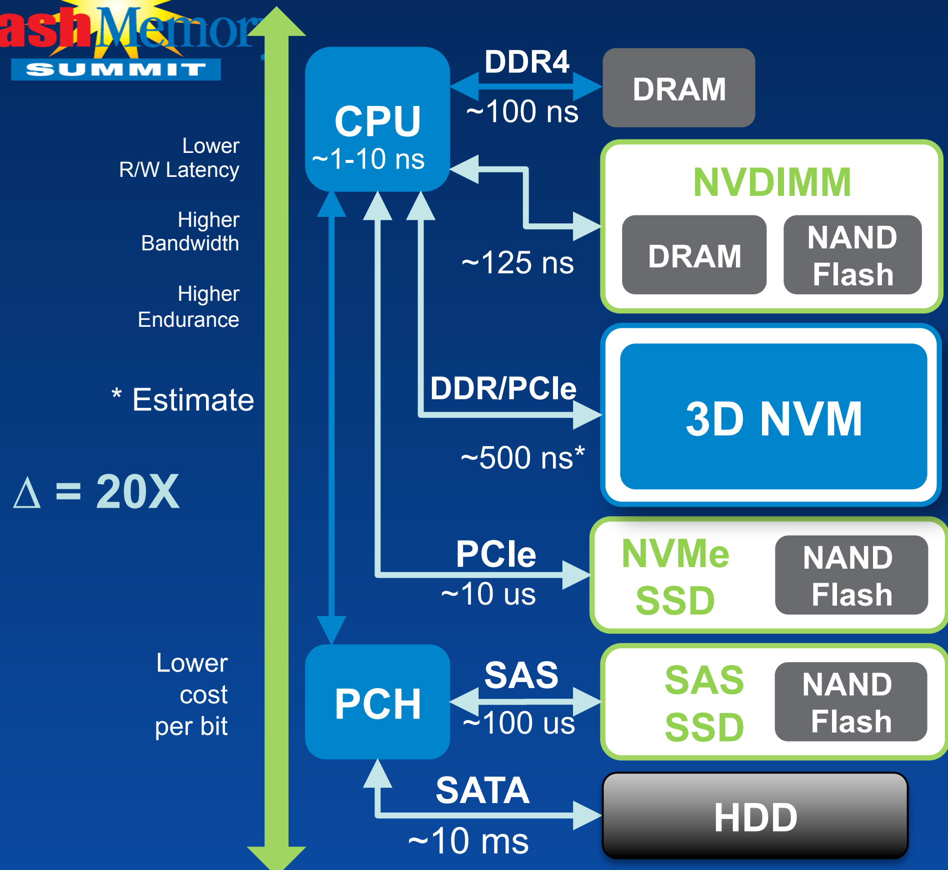
- For decades we've had two primary types of memories in computers: DRAM and Hard Disk Drive (HDD)
- DRAM was fast and volatile and HDDs were slower, but nonvolatile (aka persistent)
- Data moves from the HDD to DRAM over a bus where it is the fed to the processor
- The processor writes the result in DRAM and then it is stored back to disk to remain for future use
- ATA HDD is 100,000 times slower than DRAM (!!!)

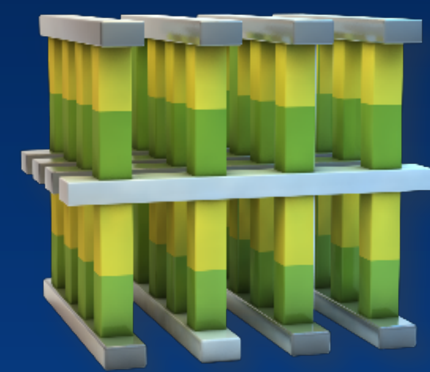
The Present: 2D Hybrid Memory Server Architectures



- System performance increased as the speed of both the interface and the memory accesses improved
- NAND Flash considerably improved the nonvolatile response time
- SAS and PCIe made further optimizations to the storage interface
- NVDIMM provides battery- or ultra-capacitor-backed DRAM, operating at near-DRAM speeds and retains data when power is removed
- NVMe transport provides efficient use of PCI-Express bus (queues, etc.)

The Future: 3D Nonvolatile Memories in Server Architectures





Courtesy Micron

- NVM technology provides the benefit in 'the middle' – reduces the gap
- Significantly faster than NAND Flash with much higher endurance
- Performance can be realized on PCIe or DDR buses – storage or memory
- Lower cost per bit than DRAM while being considerably more dense
  - Software-enabled via PMEM & others

The Inflection Point

- There is no question whatsoever that persistent memory changes compute
- But does it change storage?
  - Is persistent memory just faster storage for what we have?
- Should I just throw persistent memory ‘at the problem’?
- This technique is currently being used in SSDs

- Throw NVMe at the problem – faster transport, less overhead, more queues, etc.
- Throw dense 3D NAND flash at the problem – 512TB in 3U – save W,BTU,RU
- That’s all well and good – BUT …

Solve the Weiji

- We have a weiji on our hands 危機, translated, 'critical point'
- Instead of treating data like we have for ~60 years now – blocks – look at bits
  - Like DNA – order matters – only two base pairs (A+T, C+G) – adapts over time
- Translate (encode) the data into a better (space efficient, compute efficient, secure) form
- Use persistent memory to save metadata and translation (bit markers, instructors)
  - No disk necessary of any kind – SSD or otherwise
  - Takes only 4GB of DRAM to hold all possible combinations of 32-bit entities (2^32)
- It takes 14 bytes (13 letters and a blank) to represent the words 'critical point'
- It takes 4 bytes (2 bytes per symbol) to represent weiji
- The meaning to the end user is the same – so which is better to persist?
- CHANGE THE GAME – not the rules – compute in-memory using 10X the data

Building Datacenter Infrastructure Using Persistent Memory

Forum R-21 – Persistent Memory

Rob Peglar
Senior VP & CTO
Symbolic IO