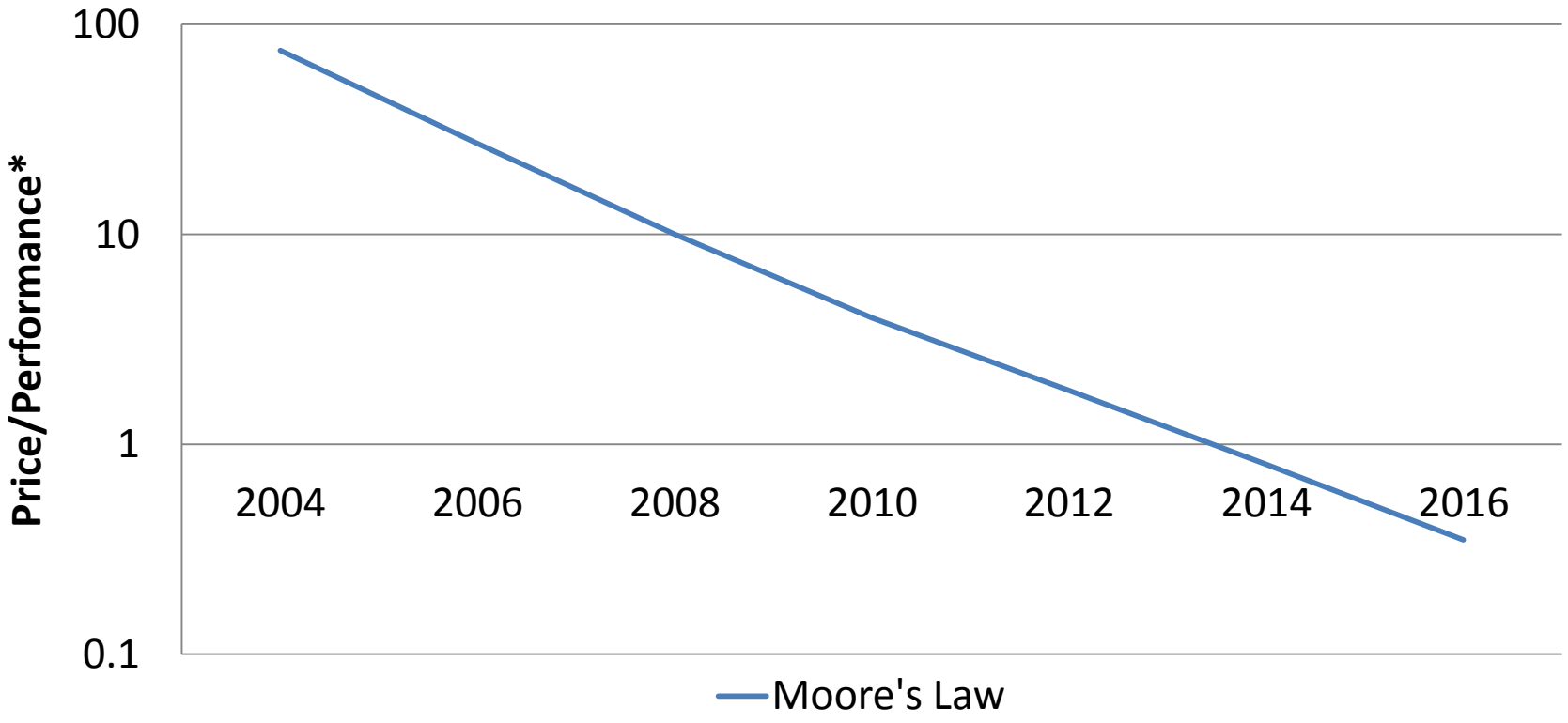
12th ANNUAL WORKSHOP 2016

OPENPOWER-BASED OPEN HYBRID COMPUTING

Building the Ecosystem for a Flexible Heterogeneous Compute Architecture

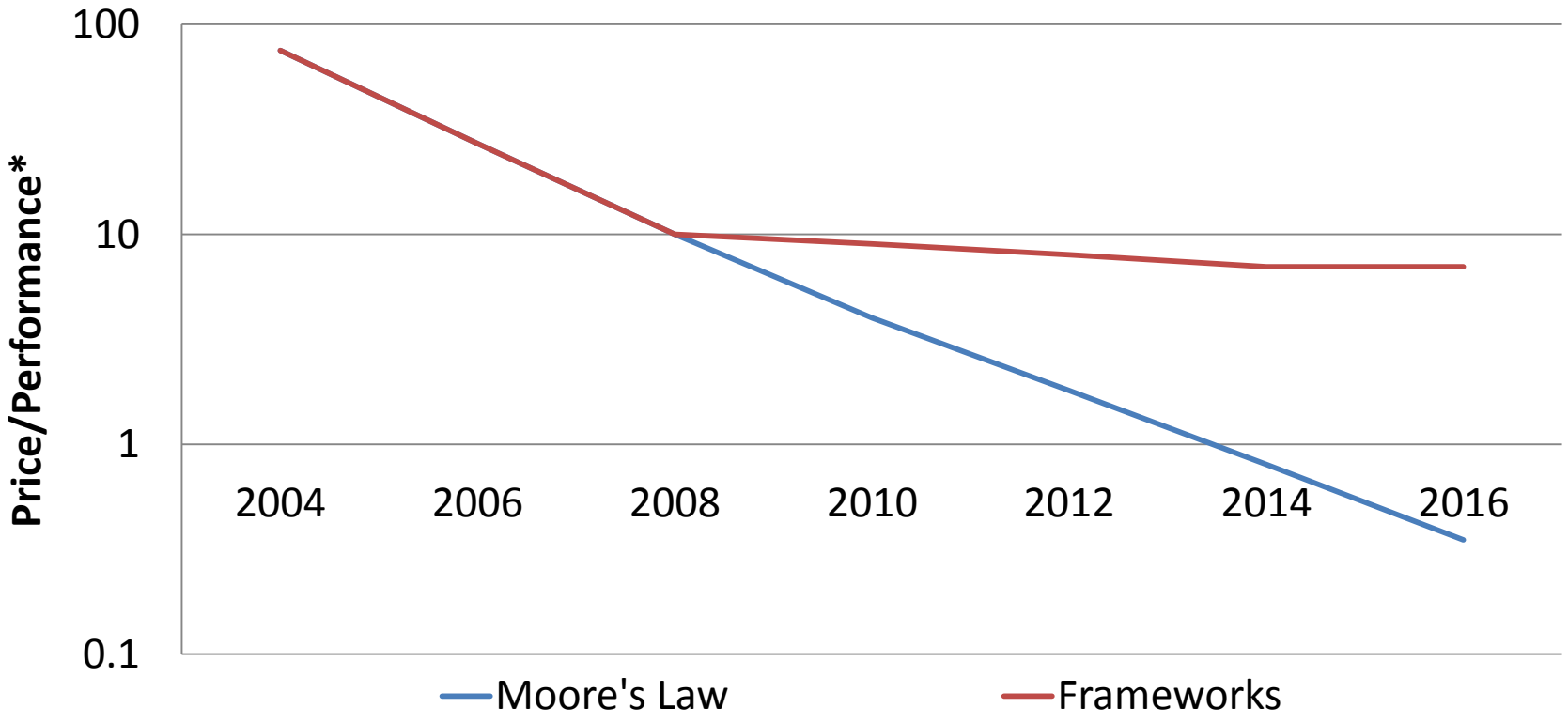
Bernard Metzler, IBM Zurich Research

INDUSTRY TRENDS: TOWARDS WORKFLOW OPTIMIZED SYSTEM



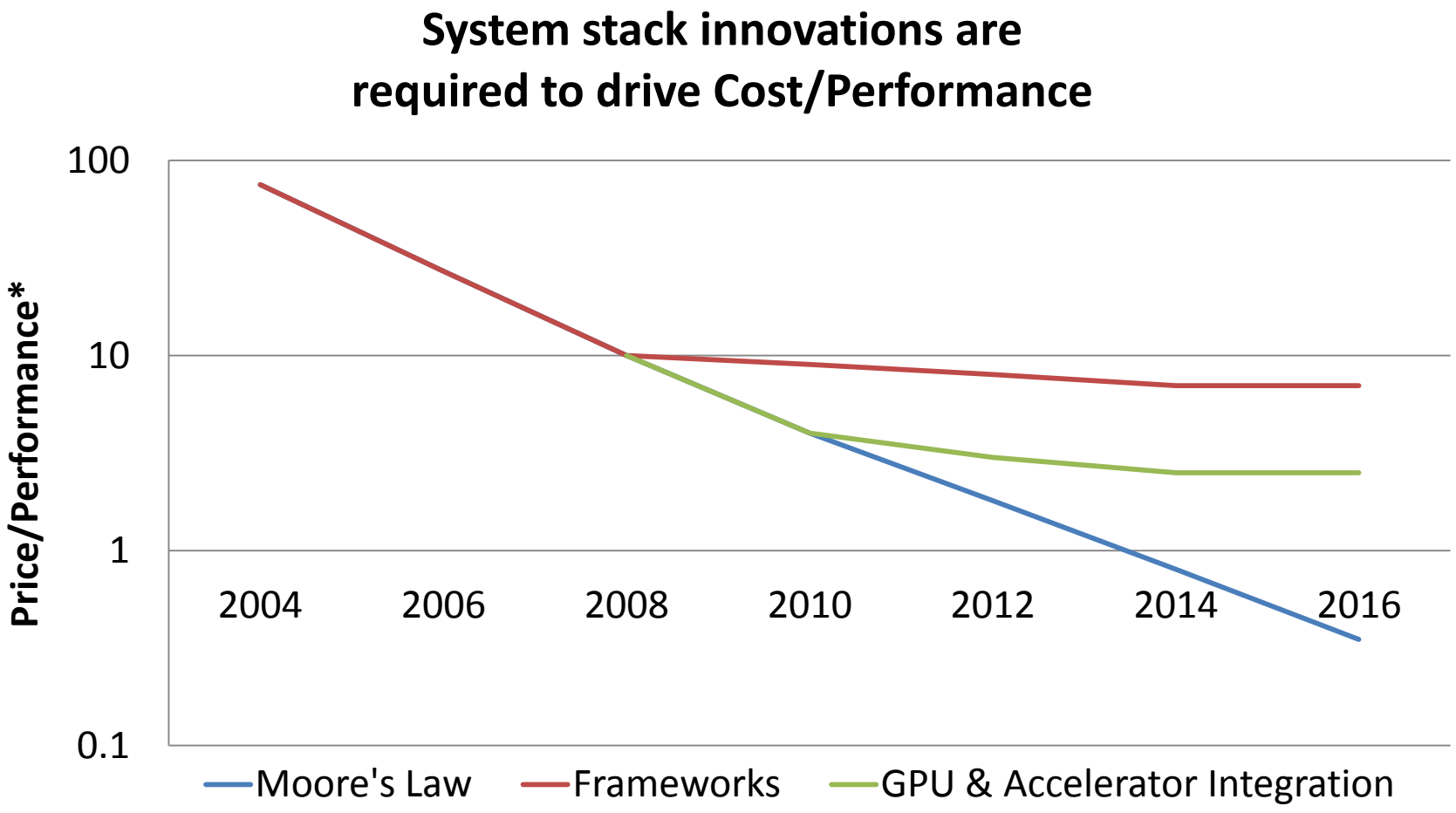
(*schematic diagram)

INDUSTRY TRENDS: TOWARDS WORKFLOW OPTIMIZED SYSTEM



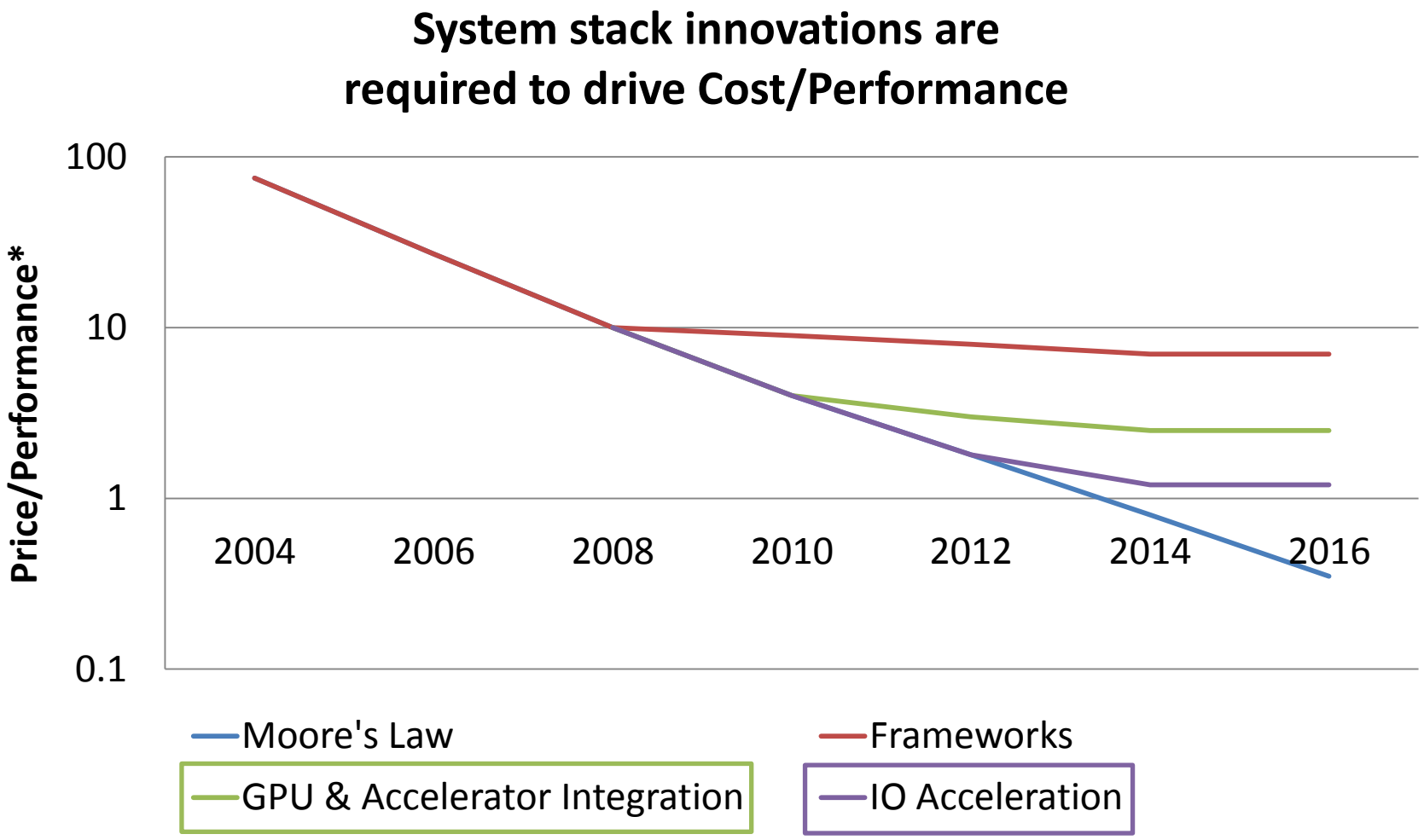
(*schematic diagram)

INDUSTRY TRENDS: TOWARDS WORKFLOW OPTIMIZED SYSTEM



(*schematic diagram)

INDUSTRY TRENDS: TOWARDS WORKFLOW OPTIMIZED SYSTEM



(*schematic diagram)

OPENPOWER CONSORTIUM

- **Collaboration around Power Architecture products**
- **Open technical membership organization, founded in 2013**
- **IBM is opening up technology around Power Architecture on a liberal license:**
  - Processor specifications
  - Firmware
  - Software
- **Member companies:**
  - Can customize POWER CPU processors and system platforms
  - Custom systems for large/warehouse data centers
  - Custom workload acceleration through **GPUs, ASICs, FPGAs, advanced I/O**
  - Over 190 members as of now
- **Goal: Create an ecosystem of HW and SW development to drive innovation in HPC and cloud computing**
  - Allow for workload optimized solutions
  - Rely on open, stable spec's and interfaces
  - Agree on a common architecture, but avoid vendor lock-in

190+ MEMBERS


Implementation / HPC / Research
Software
System / Integration
I/O / Storage / Acceleration
Boards / Systems
Chip / SOC

OPENPOWER
I/O, ACCELERATOR AND GPU TECHNOLOGY ROADMAP

Mellanox Interconnect

Connect-IB
FDR InfiniBand
PCIe Gen3

ConnectX-4
EDR InfiniBand
CAPI over PCIe Gen3

ConnectX-5
Next-Gen InfiniBand
Enhanced CAPI over PCIe Gen4

NVIDIA GPUs

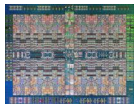
Kepler
PCIe Gen3

Pascal
NVLink

Volta
Enhanced NVLink

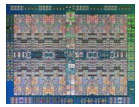
IBM CPUs

POWER8



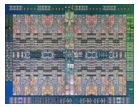
OpenPower
CAPI Interface

POWER8'



CAPI &
NVLink

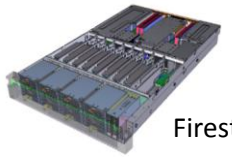
POWER9



Enhanced CAPI &
enhanced NVLink

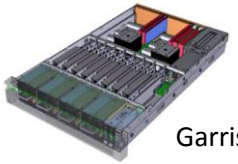
IBM Systems

2015



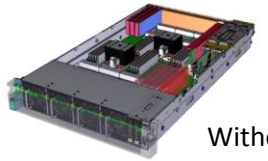
Firestone

2016



Garrison

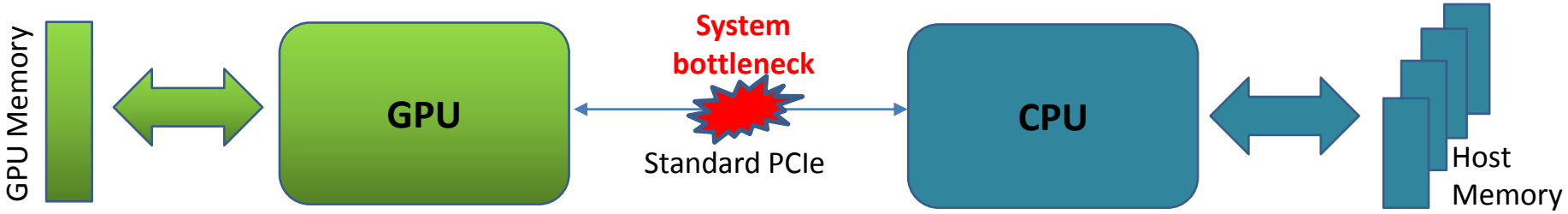
2017



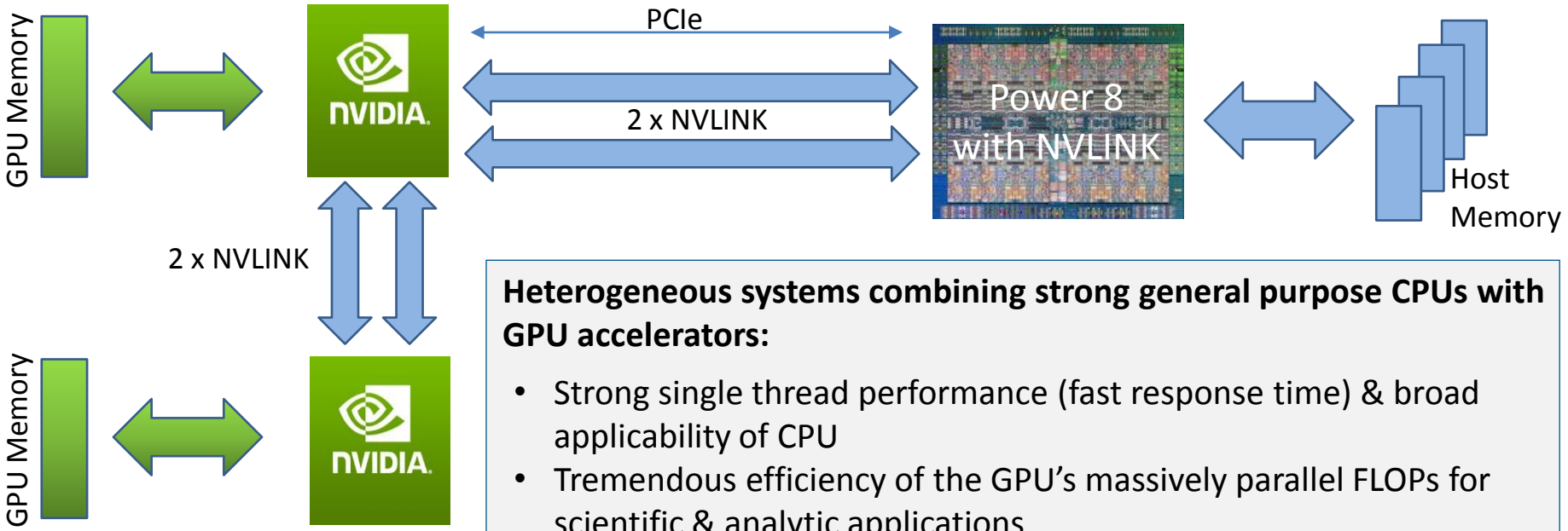
Witherspoon

EFFICIENT GPU INTEGRATION

Standard PCIe connected GPUs can't realize full potential due to communication bottlenecks



NVLINK eliminates this bottleneck, substantially increasing the GPU-CPU communication bandwidth



Heterogeneous systems combining strong general purpose CPUs with GPU accelerators:

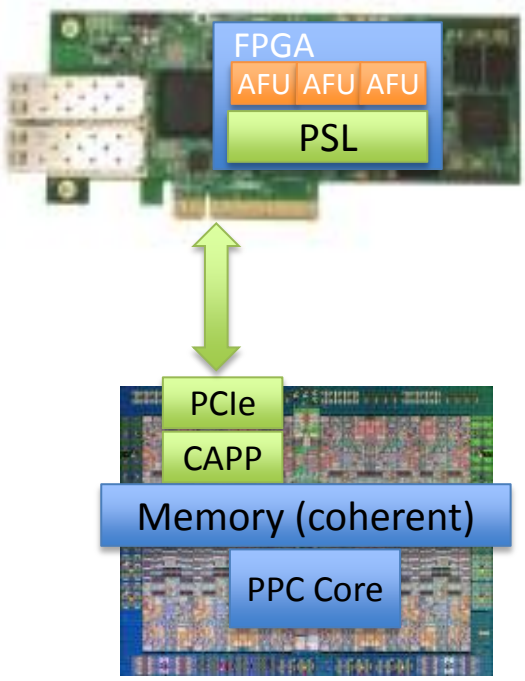
- Strong single thread performance (fast response time) & broad applicability of CPU
- Tremendous efficiency of the GPU's massively parallel FLOPs for scientific & analytic applications

WHY ACCELERATORS?

- **Transistor Efficiency & Extreme Parallelism**
  - Bit-level operations
  - Variable-precision floating point
- **Power-Performance Advantage**
  - > 2x compared to Multi-core (MIC) or GPGPU
  - Unused LUTs are powered off
- **Technology Scaling better than CPU/GPU**
  - FPGAs are not frequency or power limited yet
  - 3D has great potential
- **Dynamic reconfiguration**
  - Flexibility for application tuning at run-time vs. compile-time
- **Additional advantages when FPGAs are network or storage connected**
  - allows network/storage as well as compute specialization
  - Near memory computation

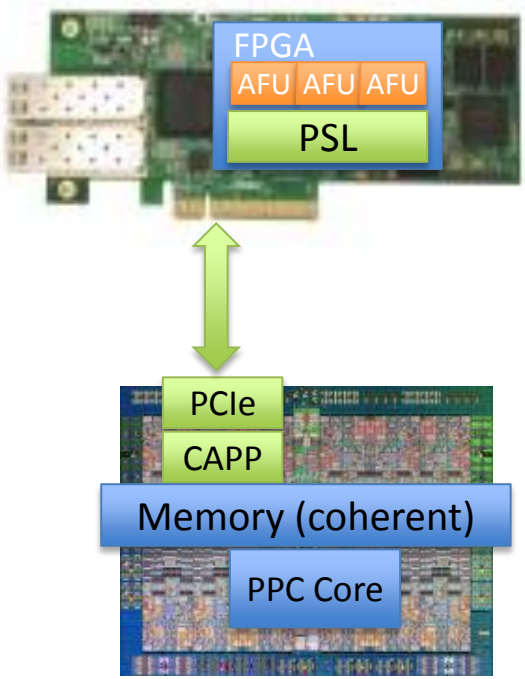
CAPI: COHERENT ACCELERATOR PROCESSOR INTERFACE

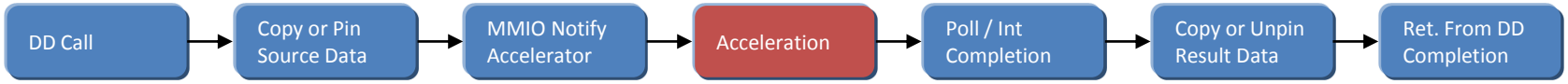
- **Customer defined acceleration within processor context**
- **Function based acceleration:**
  - Main application executed on host processor
  - Computational heavy functions offloaded
  - Single binary with both HW and SW version of function
    - Runs also w/o accelerator available
- **CAPI device: full peer to processor**
  - Enables coherent memory between AFU and application
    - Via PSL, AFU can ‘understand’ app’s VA
    - No need to pin application memory for I/O
    - AFU works within application context



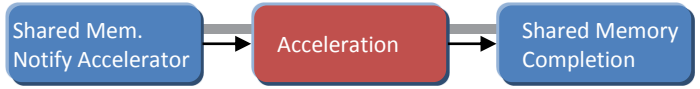
CAPI: COHERENT ACCELERATOR PROCESSOR INTERFACE

- **Customer defined acceleration within processor context**
- **Function based acceleration:**
  - Main application executed on host processor
  - Computational heavy functions offloaded
  - Single binary with both HW and SW version of function
    - Runs also w/o accelerator available
- **CAPI device: full peer to processor**
  - Enables coherent memory between AFU and application
    - Via PSL, AFU can 'understand' app's VA
    - No need to pin application memory for I/O
    - AFU works within application context



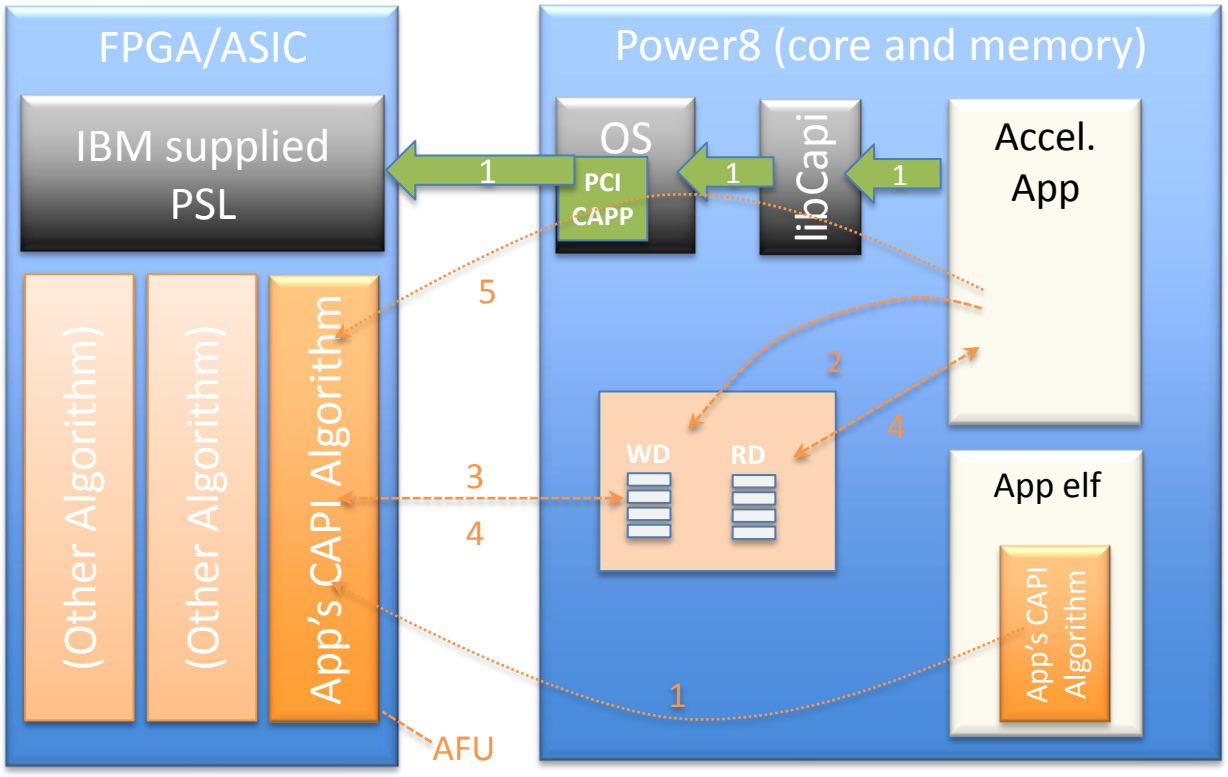


Typical I/O Model Flow



Flow with a Coherent Model

CAPI: ACCELERATOR OPERATION

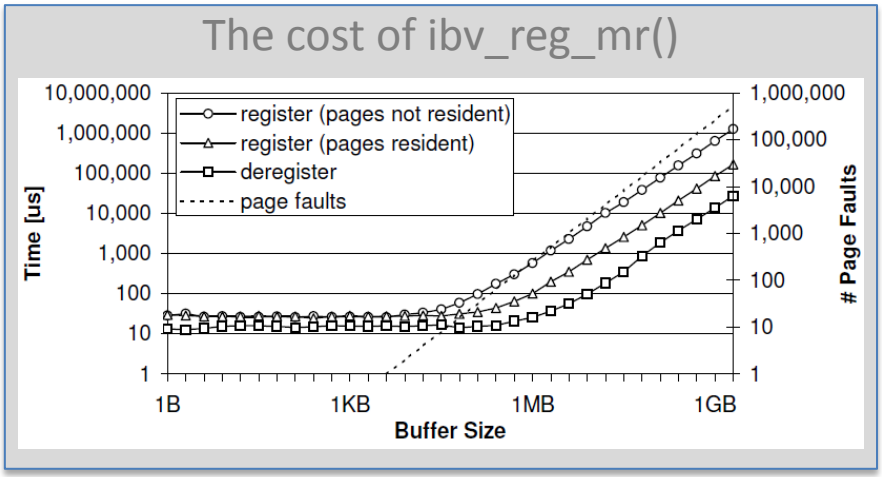


0. **OS is aware of CAPI device(s), libCapi in place, application bin contains algorithm**
1. **Algorithm gets transferred to FPGA**
2. **Shared context set up containing shared**
   - Work descriptors,
   - Result descriptors
   - Shared locks,
   - …
3. **CAPI algorithm starts working**
4. **Ongoing communication between app. and CAPI attached algorithm**
5. **Work completes, CAPI algorithm shutdown**

1. **Virtual addressing: no pinning, no copy**
2. **Coherent caching**
3. **Elimination of device driver: direct app – device communication**
4. **Efficient hybrid HW/SW co-design**

EFFICIENT RDMA: MEMORY REGISTRATION

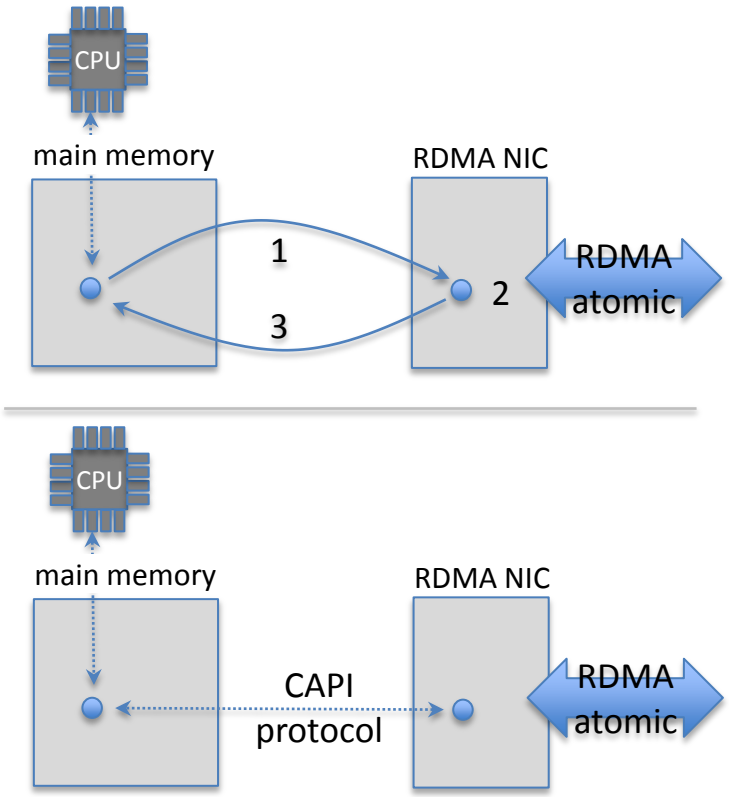
- **Classic RDMA communication buffer management**
  - Static communication buffer memory pinning for RDMA memory access
  - Adapter translates VA/PA on fast path
  - Potential waste of host memory resources
  - Time consuming for short lived communication contexts



- **CAPI model**
  - No memory pinning needed
    - Adapter runs in user context and 'understands' VA
    - OS supports on the fly page-in if needed
    - No pinning of sparsely used memory
  - Memory registration
    - Communicates buffer boundaries to adapter
    - Used by adapter to enforce access protection
  - Elegant way of doing 'lazy memory registration'

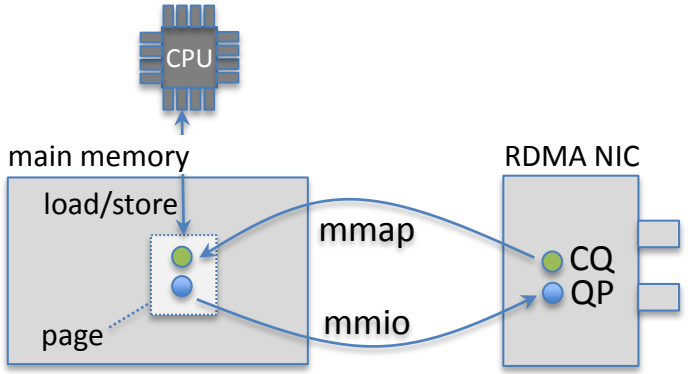
EFFICIENT RDMA: ATOMIC OPERATIONS

- **RDMA Atomics**
  - Atomic read-modify-write operations on small in memory objects, such as
    - Compare & Swap, Fetch & Add
  - Used for efficient distributed access control
    - Database, collective operations etc.
- **Classic Implementation**
  - Adapter fetches current object value via PCI transfer (1)
  - Adapter potentially sets new value (2) and writes back via PCI transfer (3)
- **CAPI implementation**
  - Adapter and host share PSL/CAPP managed consistent view on atomic object
  - Atomic object cached/in-situ access by adapter, PCI xfer only if local CPU accesses
  - Higher Atomic RDMA OP/s rate possible
  - Atomics to be moved to memory controller (enhanced CAPI)

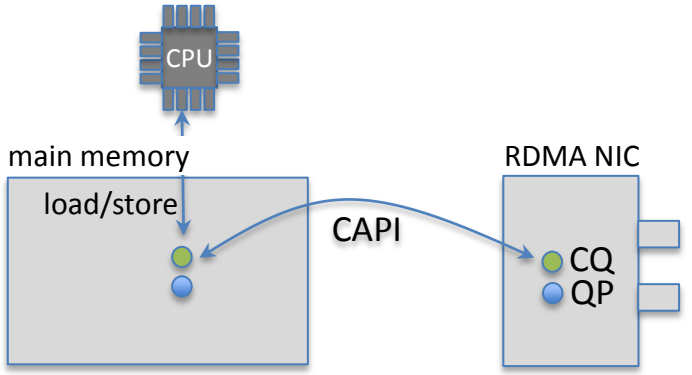


EFFICIENT RDMA: ENDPOINT RESOURCE ALLOCATION

- **Classic RDMA endpoint model**
  - Memory mapped endpoint resources
    - QP, CQ, RQ, …
    - Memory mapped doorbell register
  - Overhead from memory mapping (page pinning)
  - Overhead for short lived endpoints (endpoint setup time)

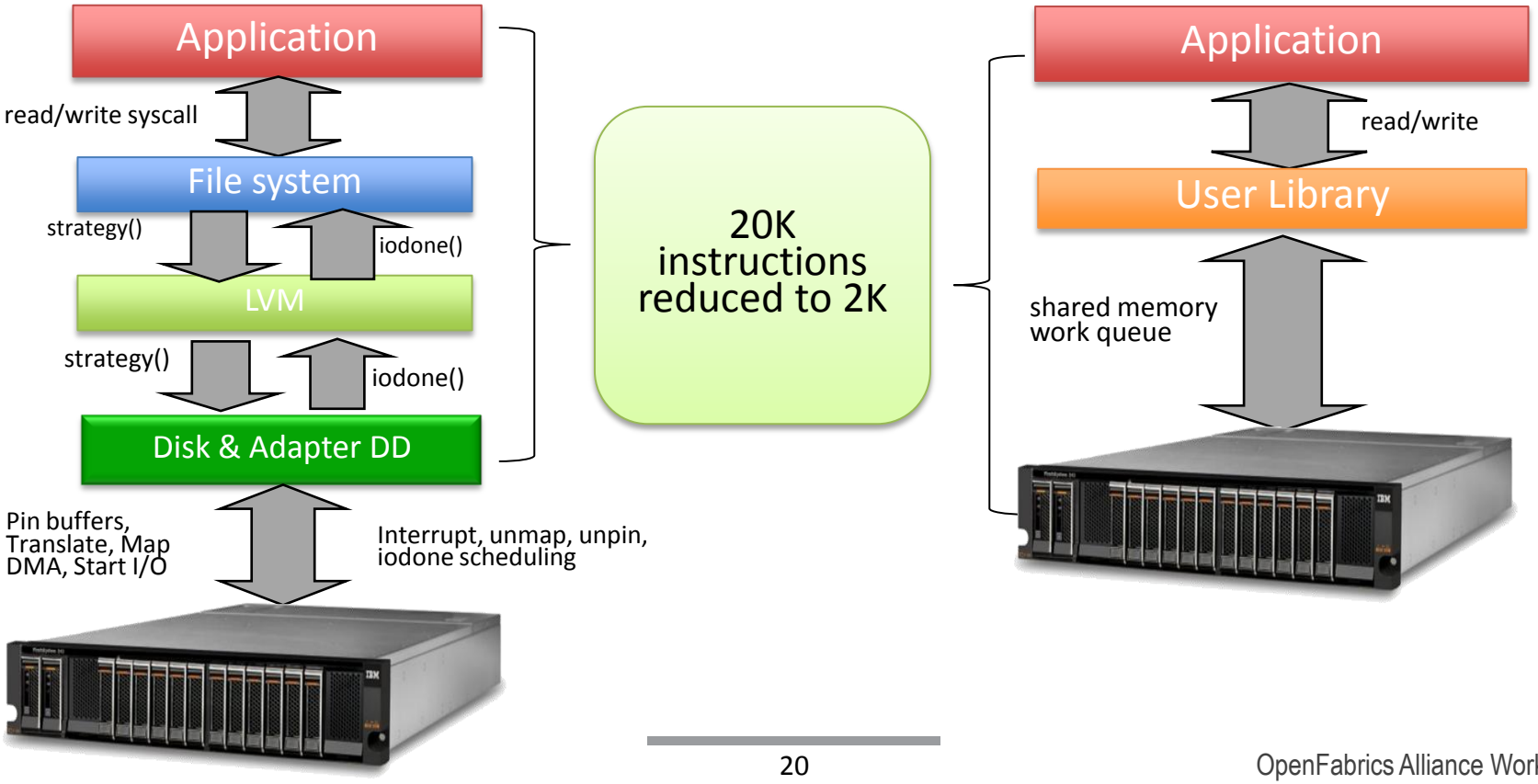


- **Using CAPI model**
  - Application transparently shares endpoint resources with adapter
  - CPU and adapter updates to work queues and other structures are kept coherent
  - Simplifies user library



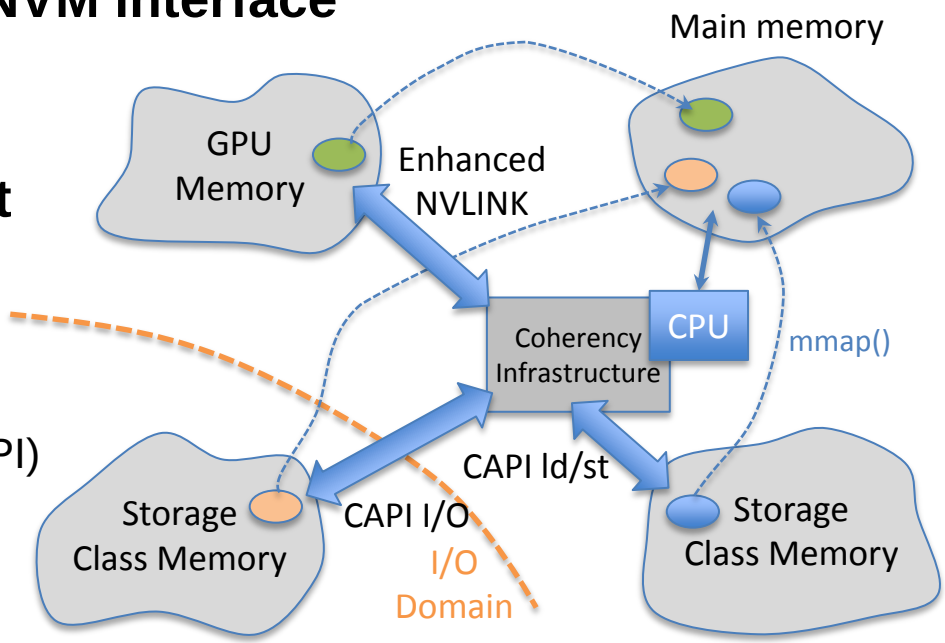
CAPI AND NVM INTEGRATION

- **TMS Flash attached to POWER8 via CAPI FC coherent adapter**
- **Read/Write from application to significantly shorten code path**
- **5 x the bandwidth per core at half the latency**



COHERENT NVM INTEGRATION - OUTLOOK

- **CAPI I/O model will benefit from higher P9 CAPI bandwidth**
- **Emerging new non-volatile memory technologies**
  - Higher endurance, lower latency
  - Justify closer integration with CPU
  - Complement asynchronous I/O model
- **Introduction of true load/store NVM interface**
  - Use enhanced CAPI attached memory
  - CPU load/store to NVM
- **Results in 3 models of coherent access to mmap()'d memory:**
  - GPU Memory (enhanced NVLINK)
  - NV Memory async. I/O access (CAPI)
  - NV Memory ld/st access (enhanced CAPI)



SUMMARY AND OUTLOOK

- **Accelerator and GPU integration**
  - Dictated by technology trends
  - Efficient integration means:
    - Flexible software/accelerator co-design
    - Maintaining coherency
    - IO balanced, high sustained bandwidth
- **OpenPOWER**
  - Industry consortium building open ecosystem around POWER technology
  - Aiming at efficient Accelerator and GPU integration
  - Coherent Accelerator Processor Interface (CAPI) and NVLINK
- **OpenPOWER based efficient OpenFabrics RDMA stack**
  - Lazy memory registration
  - RDMA endpoint management
  - Coherent RDMA Atomics
  - Efficient NVM integration
- **Outlook: hybrid computer systems**
  - Programmable near-memory accelerators
    - → see 2016 OpenPOWER Summit talk “Programmable Near-Memory Acceleration on ConTutto”

OPENFABRICS
ALLIANCE

12th ANNUAL WORKSHOP 2016

THANK YOU