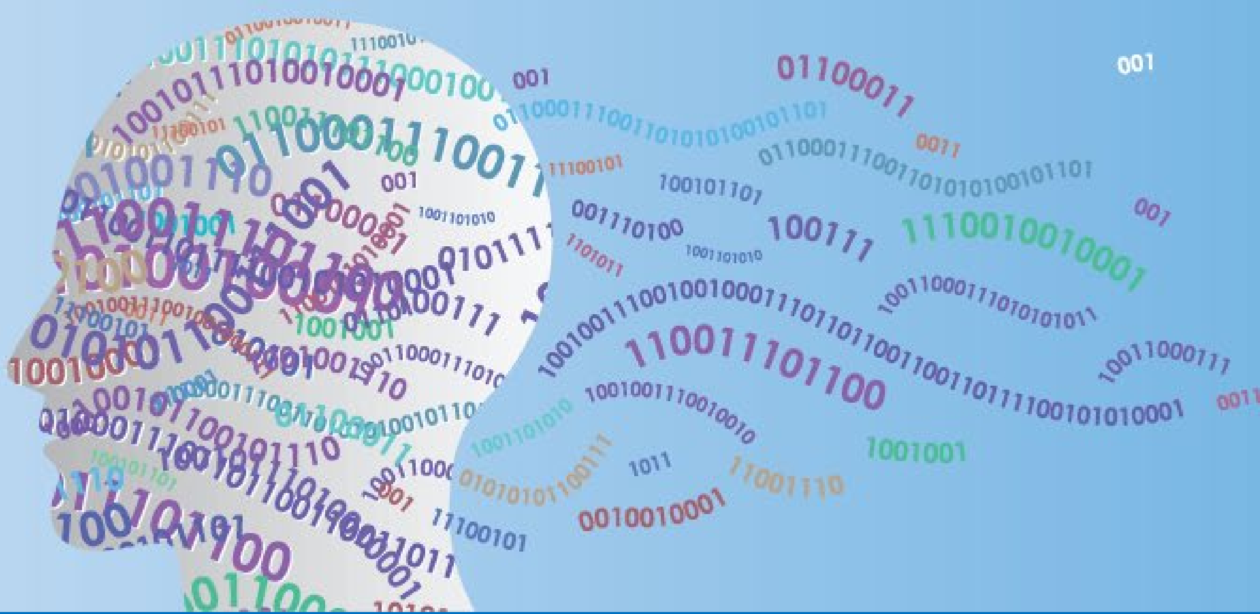
SNIA PERSISTENT MEMORY + COMPUTATIONAL STORAGE SUMMIT 2021

FROM DATACENTER TO EDGE : VIRTUAL EVENT
APRIL 21-22, 2021



Practical Computational Storage: Performance, Value, and Limitations

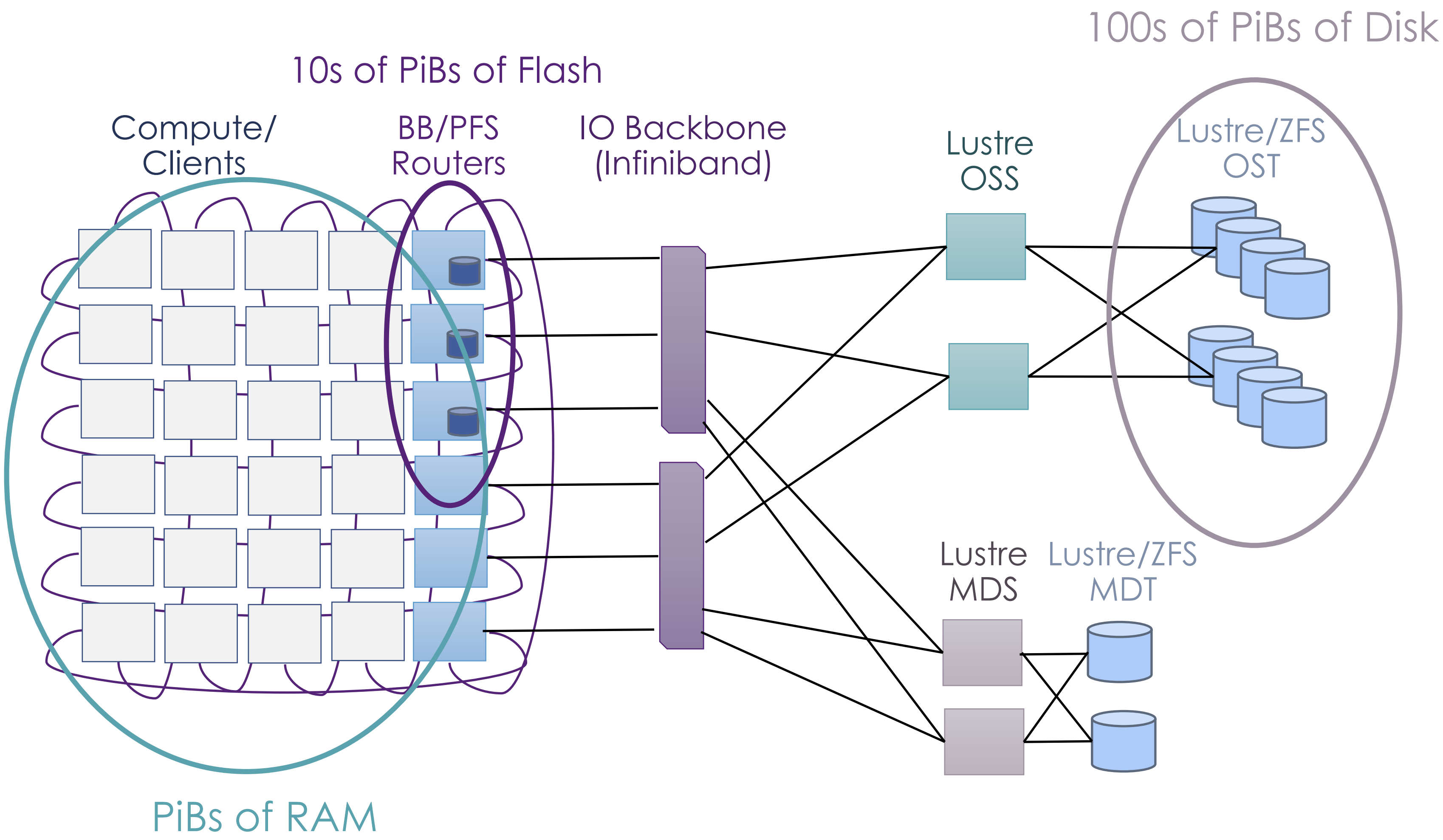
Brad Settlemyer, Sr Scientist, Los Alamos National Laboratory

Outline

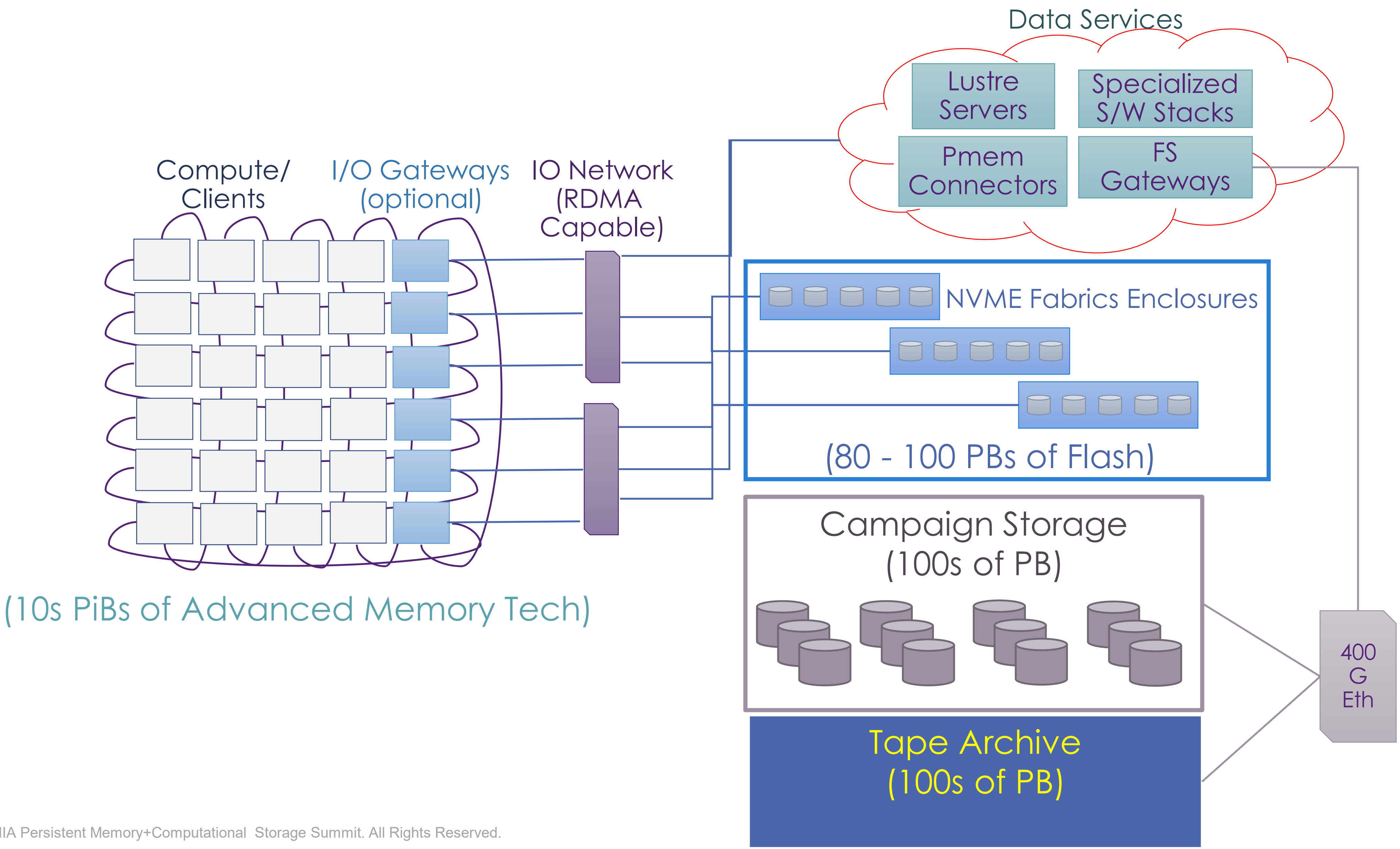
- A brief introduction to HPC Storage
- What does it mean for computational storage to be practical?
  - Problems CS* addresses, deployment
- What does performance for computational storage mean?
  - Does relocating a processor from a motherboard to an SSD really help?
- How can we evaluate the value of computational storage?
  - Processors here, processors there, what's the difference …
- What are the practical limitations?
  - Deployment considerations, CS* as a service vs CS

A Brief Overview of HPC Storage

A Simplified HPC Platform



A Next Generation HPC Platform



HPC Storage Challenges

- Petascale (and now Exascale) leverage **scale-out** to achieve high performance
  - CPU frequency gains are long gone (2002 P4 was 3GHz – just like today)
  - From 1,000s of cores to 1,000,000s of cores
  - A few exceptions:
    - HCA latencies have decreased (but switch latencies not as much)
    - GPUs brought fast context switches (but require thread scale-out)
  - While benchmark FLOPs increased, efficiency has worsened!
- **How do you solve a fixed-size problem faster?**
- **How do you make a storage system faster without making it bigger?**

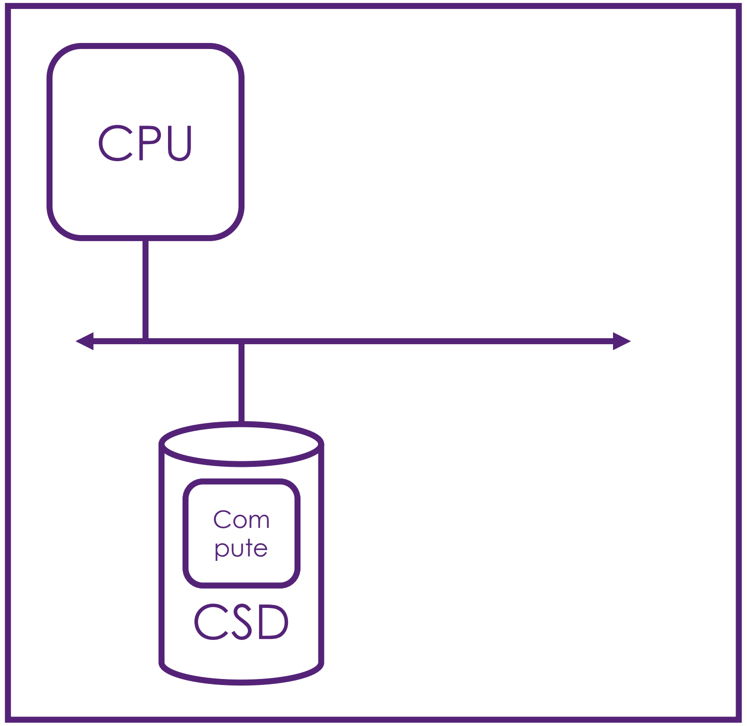
Fast Data Services

Computational Storage as an enabling technology

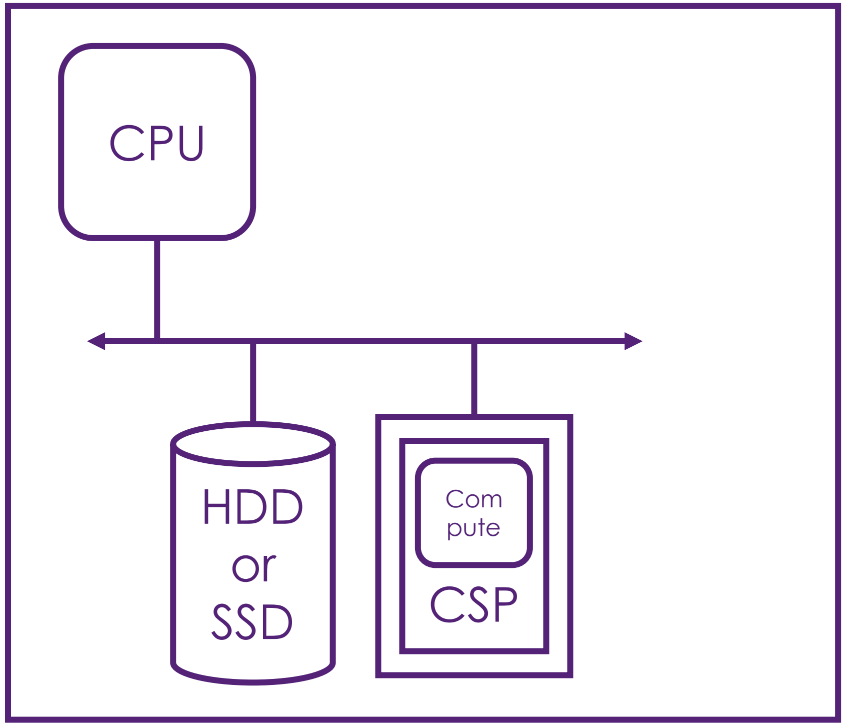
The Hard Thing about Strong Scaling

- Limitations of weak-scaling
  - We want to solve existing problems faster, not just make them bigger
  - What if I want to store a 100GB file as quickly as possible?
    - With more than a few thousand disks you become latency/layout bound …
    - And with NVME SSDs you just move to the next bottleneck, memory bandwidth
- **The pursuit of one time step functions**
  - Synchronous runtimes -> Asynchronous runtimes
  - DDR -> HBM
  - Which brings us to Computational Storage …

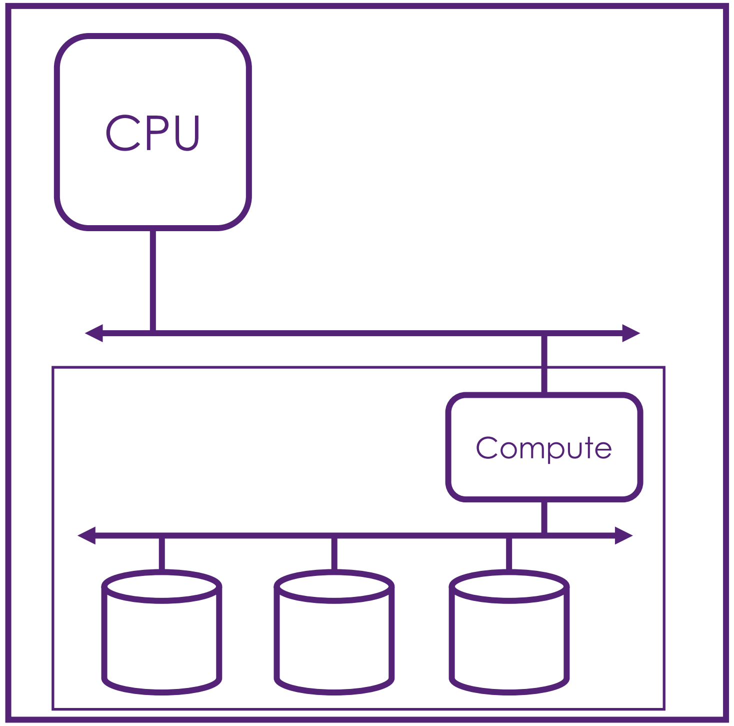
Computational Storage Examples



Computational Storage Device



Computational Storage Processor

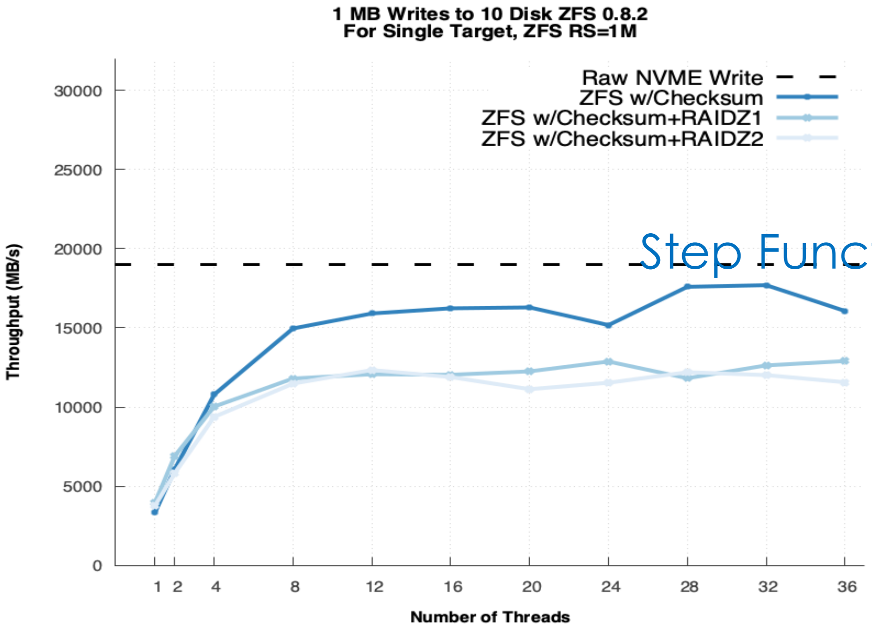


Computational Storage Array

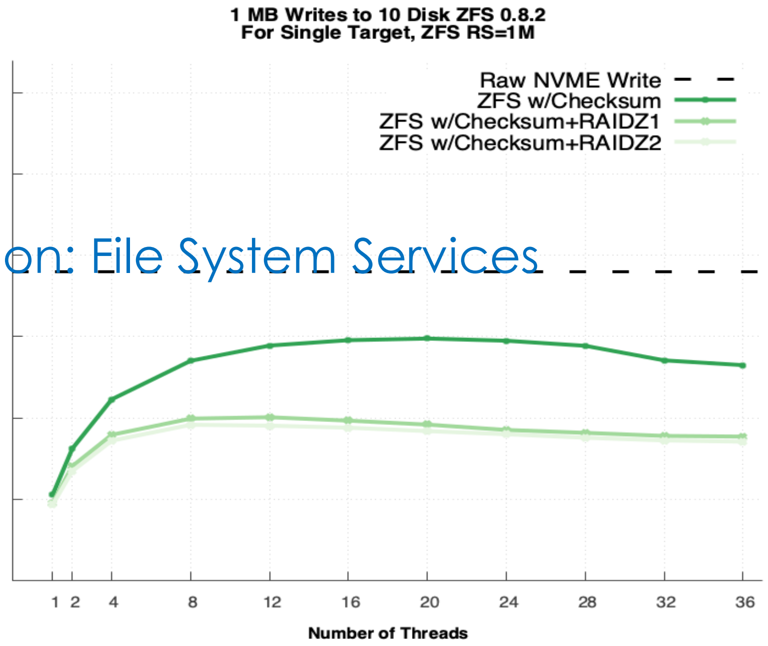
Enabling New Step Functions

- Computational Storage is not in and of itself a step function
  - High-quality, line-rate compression is a step function
  - High-quality, low-overhead index creation is a step function
  - And many more …
- **In general**
  - **CSDs deliver benefit when data can be semantically interpreted at the device**
  - **CSPs deliver benefit by providing efficient processing capabilities along data path**
  - **CSAs deliver benefit for complex computational storage services**
- **But I said practical, so let's be specific …**

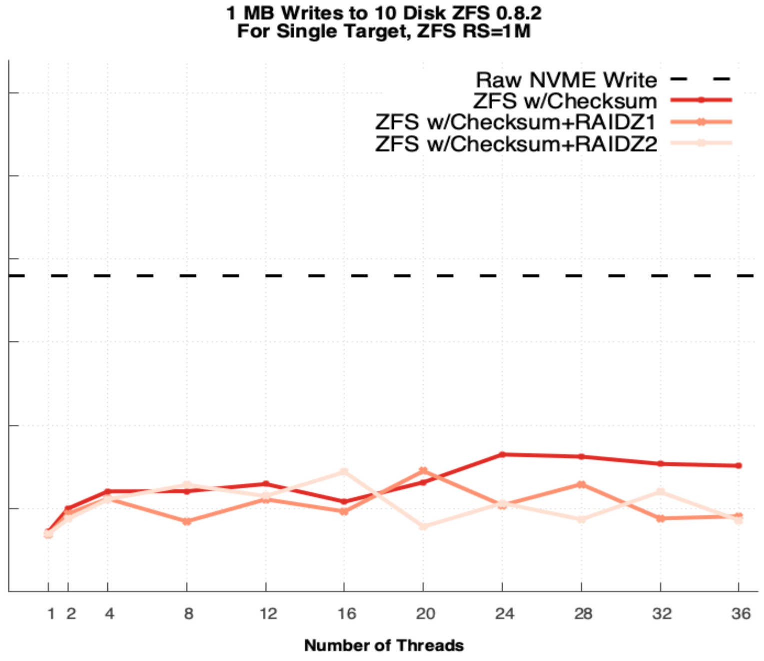
Step Function: File System Services



Intel Platinum (Dual Socket)



AMD EPYC (2^{nd} Gen)



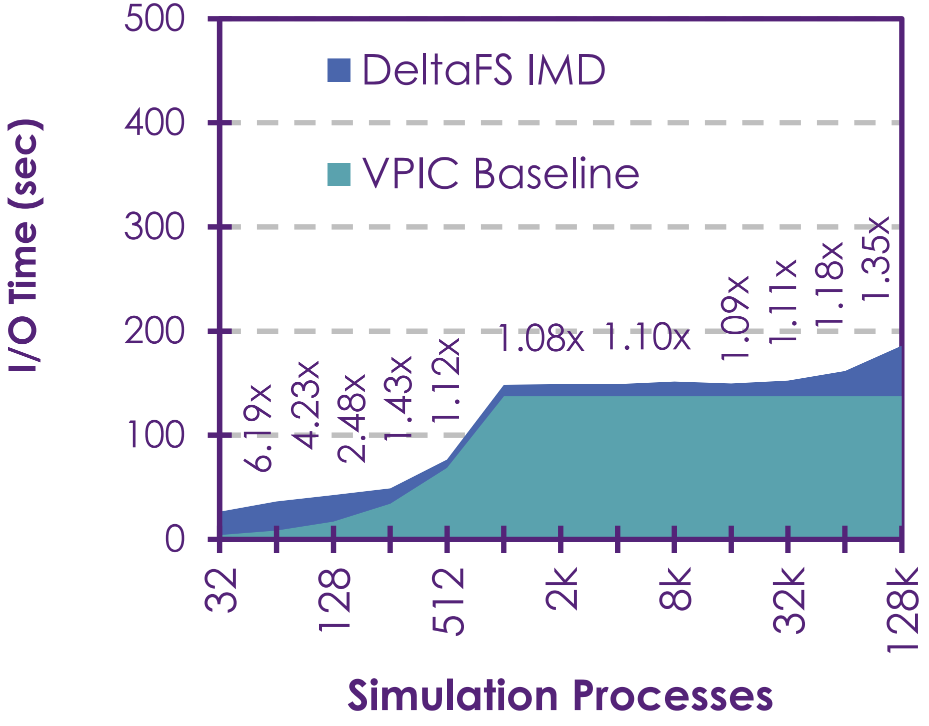
AMD EPYC (1^{st} Gen)

Step Function: File System Services

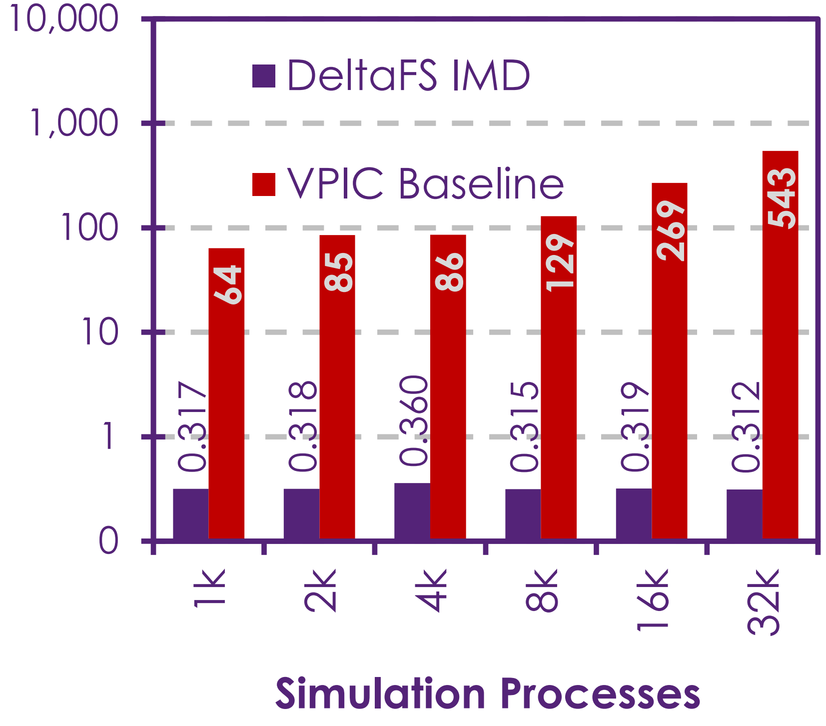
Step Function: File System Services Offload

- Increase compression rates from 1.06:1 -> 1.3:1 for scientific data
  - From 6% to 30% is a 5x improvement? No, that's not how math works
  - LANL is a bit unlucky we don't see more benefit
- Enable complex coding/decoding to protect against correlated failures
  - Dynamically choose which operations to accelerate
- Achieve higher per-server and per-device bandwidths
  - Approaching line-rate
- Lower server costs and quantities

Step Function: Near-Storage Analytics



Overheads for streaming index creation (IMD) versus no indexes



Query performance with indexes (IMD) versus no indexes

Step Function: Near-Storage Analytics

- Speedups for post-hoc analysis (1000x speedup demonstrated)
- Post-hoc index creation (speculative)
- Less reliance on massive compute tier as a large merge sort space
- More agile access to data (less time waiting in scheduler queue)

Evaluating Value

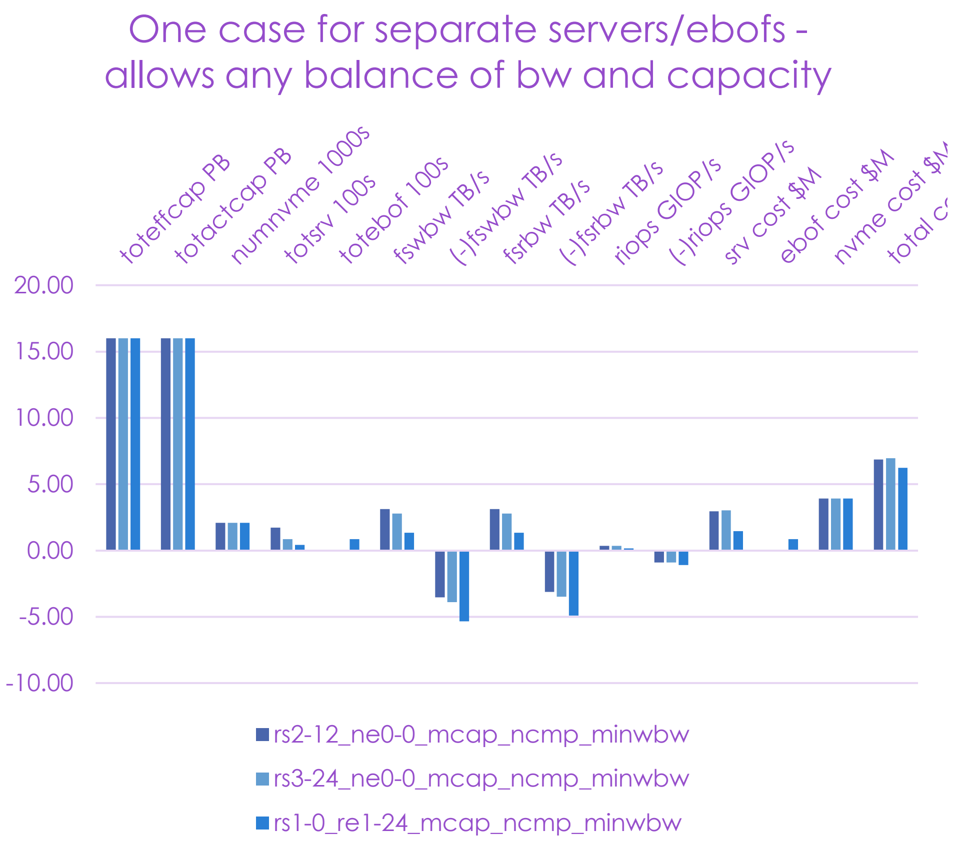
Computational Storage Value

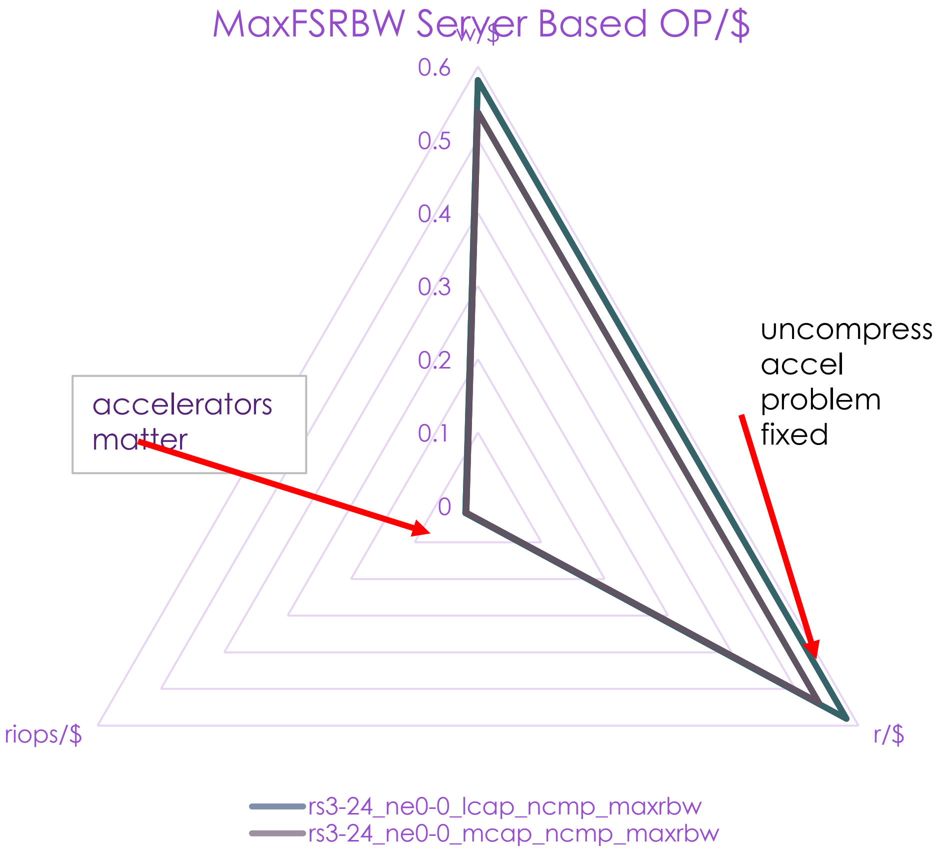
- Moving CPU to NIC/SSDs is not obviously better
  - And definitely not obviously cheaper
  - But it might be, how do we evaluate
- Two trends to consider
  - Specialized CPUs (custom ARM, RISCV, DPU, FPGA, etc) becoming common
  - Scaling with respect to servers not appropriate for all workloads
- New storage interfaces make system design harder
  - More degrees of freedom (no longer minimize number of servers)
  - With these interfaces we can minimize all services simultaneously and independently

Computational Storage Value

- Consider the following LANL data services
  - File services with compression, decompression, checksum, erasure coding
    - Encryption, dedupe are also relevant
  - Key-value services supporting compaction, open queries, and closed queries
- How do we minimize cost for:
  - 100GB/s file write throughput
  - 100M 4K IOP reads
  - Critical rebuilds (simultaneous reads and writes)

Value Depends Upon Price

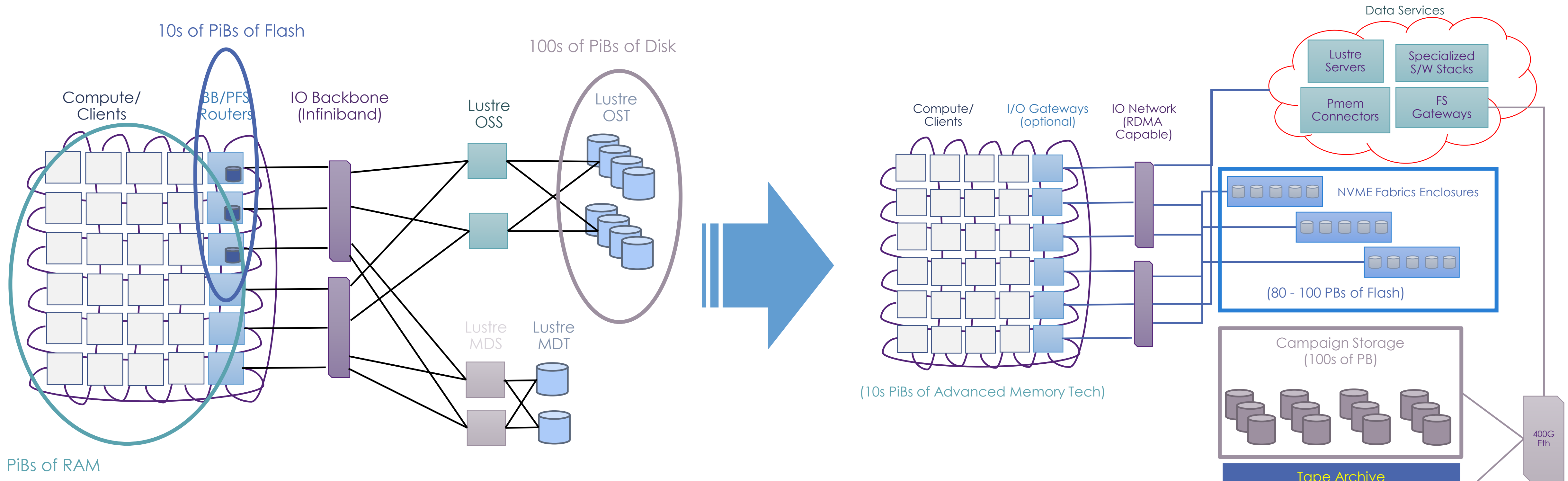




Other Practical Consideration

The boring stuff

Moving From Current to Next



Challenges for New Storage Architectures

- Userspace access to storage devices
  - HPC networks have offered user-space networking (including RDMA) for decades
  - User-space access to storage will be similarly disruptive to HPC
  - How to accomplish this is still very unclear …
- Beyond block-level services
  - New interfaces such key-value, object, etc need to be grounded in user requirements
  - But

Challenges for New Storage Architectures

- New interfaces need to provide better services
  - Researchers have spent years exploring the tradeoffs within block-level services
  - New interfaces hold the promise of semantic interpretation of data
  - But reliability
- Beyond block-level services
  - New interfaces such key-value, object, etc need to be grounded in user requirements
  - But

Closing

Conclusions

- Computational storage offers performance benefits in several practical scenarios
  - Where data transformations limited by memory bandwidth
    - File processing pipelines, complex file formats, multi-pass algorithms
  - Where large degrees of data reduction possible
    - Highly selective queries, compression
  - These aren’t hypothetical benefits, we are actually achieving these benefits with computational storage!
- Computational Storage can deliver value to LANL use cases
  - More cost-effective way to purchase memory bandwidth
  - Eliminates server-mediated restrictions on storage access

Conclusions

- Value is governed by interest
    - Understand which metrics you care about, and design to those metrics
- Computational Storage Skepticism
    - Often hear about past failures of active storage
    - Practical deployment problems do exist
    - But maybe this time is different …

Thank you

Please visit www.snia.org/pm-summit for presentations