Creating a Framework for Computational Storage

Nick Adams

Platform Storage Architect, Intel

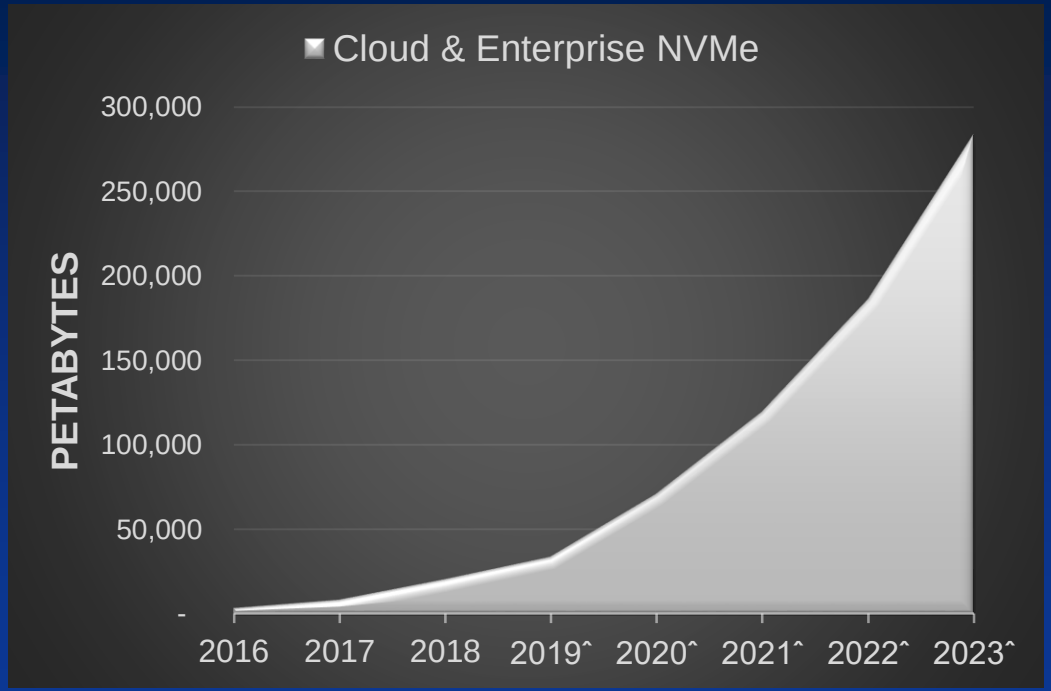
Agenda

- Why do we need Computational Storage?
- Innovation and Opportunity
- Looking Ahead
- Call to Action

Why do we need Computational Storage?

Flash Memory Summit

Huge Influx of Data



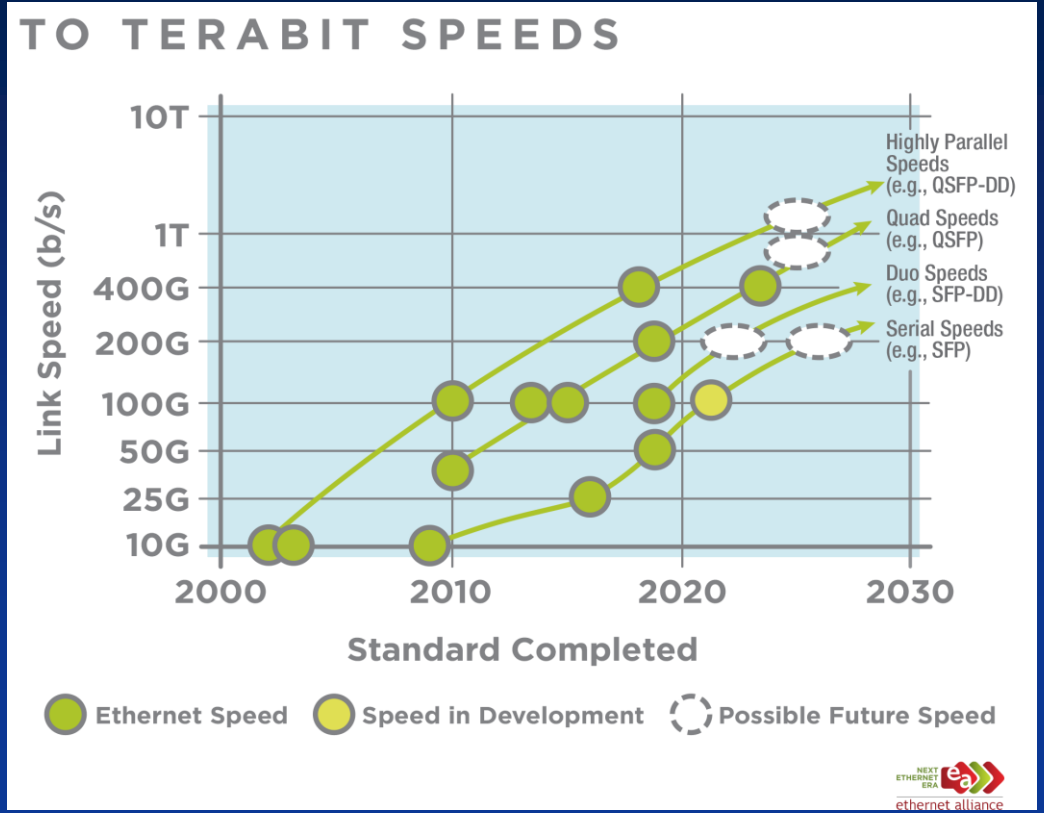
Nearly ***exponential*** growth in flash storage capacity!

ˆ Projections provided by Forward Insights Q2'19

Improved Network Bandwidth

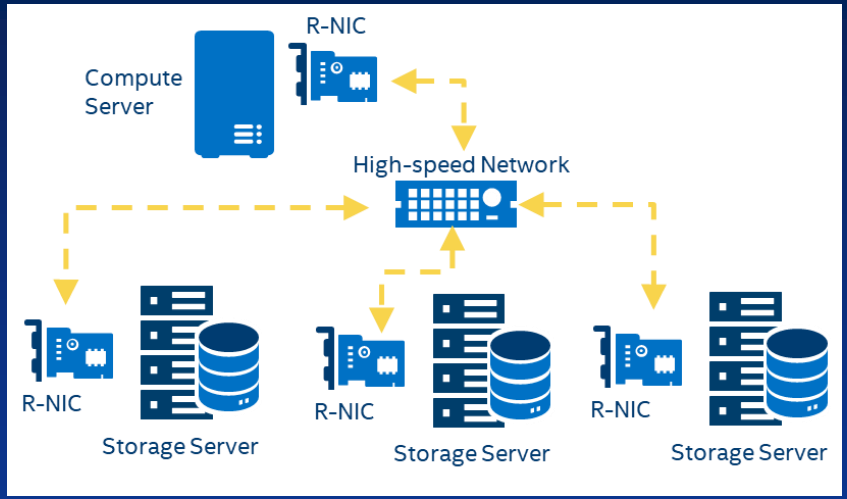
Significant network bandwidth improvements…

but still lagging storage capacity.



Continued growth of the Datacenter

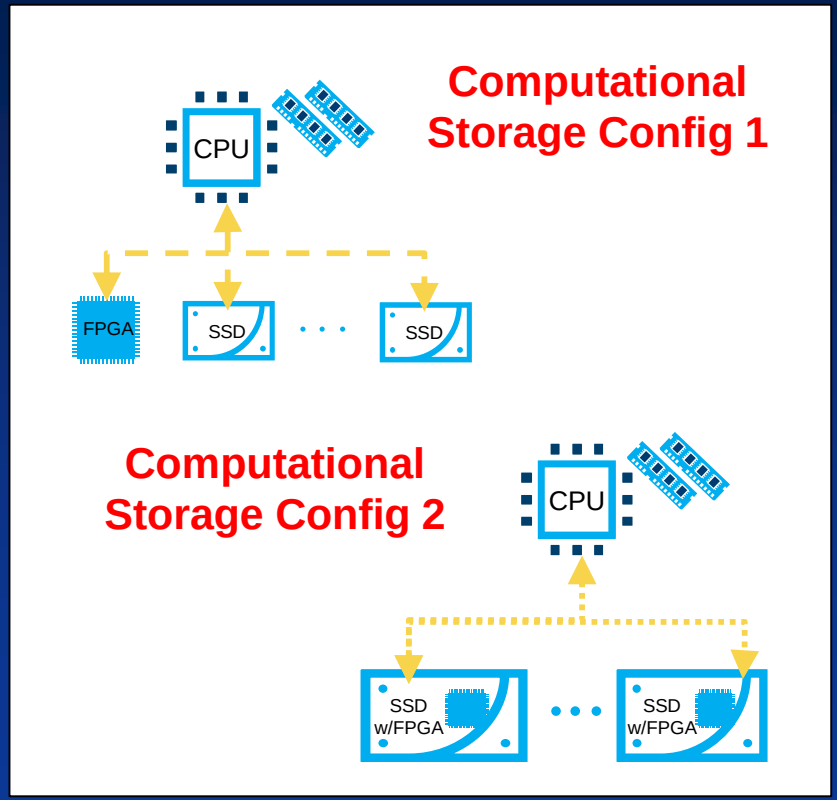
- So all that Data must be moved & processed.
- The same as we've always done?
- Maybe not…



Innovation & Opportunity

The Beginning: Storage-centric Offload

- RAID, Erasure Coding, Compression Offload
- Move infrastructure processing from the Host to the Computational Storage Device

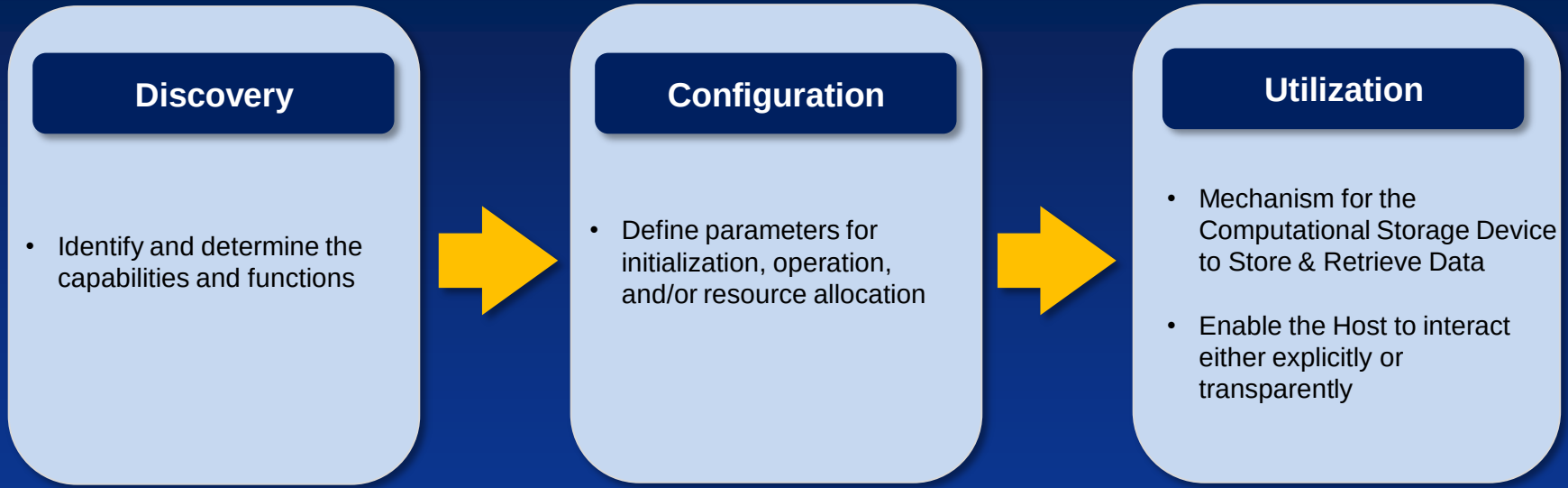


Establishing an Ecosystem

- Create a Context for communication
  - Common Definitions – CSD, CSP & CSA
    - Computational Storage Drive (CSD)
    - Computational Storage Processor (CSP)
    - Computational Storage Array (CSA)
  - Common Framework
    - Leverage an already existing ecosystem, NVMe
    - Work with the NVMe standard to enable OS ecosystems
    - Minimize or eliminate changes to application level SW

Today: Custom offerings.
Standardizing now.

Establish Foundational Principles



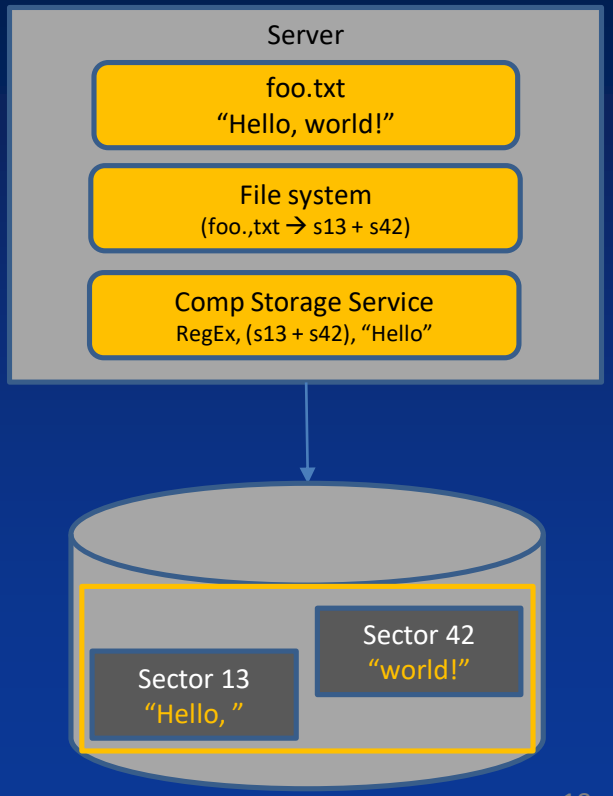
Additional Key Usage Models

- Structured Data Usages
  - Video Compression
  - Database Compaction
  - Regular Expressions (RegEx)

- Critical next step is to address Computational Storage with the context of the object at the Storage Device…

Challenges for Structured Data Usages

- Need more data to work with blocks
- Need access to additional context
  - Object Store, Key Value, another solution?
- Maybe we can make block storage *object aware* without…
  - Fundamentally changing the interface
  - Adding a lot of state and complexity
  - Ephemeral mapping of blocks to create an "object"

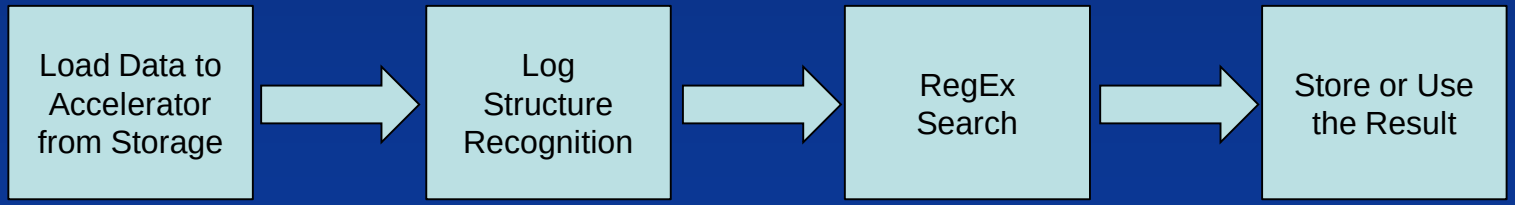


More Advanced Usage Models

- Computational Storage for AI & ML
  - Voice Recognition
  - Image Processing
  - Machine Translation
  - Unstructured Data (media streams & text)

Challenges for Unstructured Data

- Coordination of Acceleration
  - AI/ML usages quickly introduce the need to Chain, Pipeline &/or Graph the Inputs & Outputs
  - Discover Capabilities, Schedule the work, Harvest the results
- Accelerators will need direct access to Data regardless of medium
  - Via Memory Semantics – Needs to be defined
  - Consistent interface for both Storage and Persistent Memory

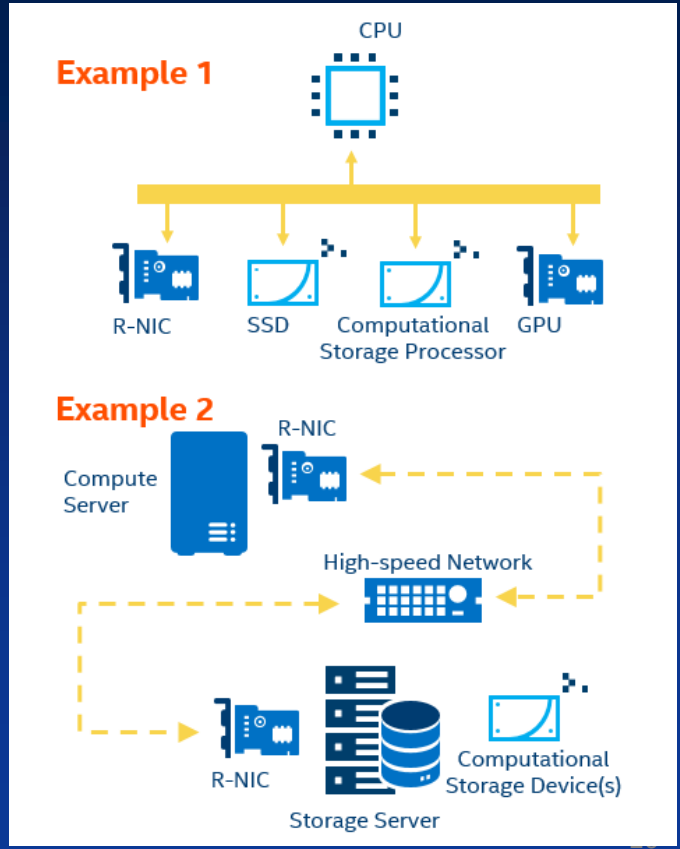


Looking Ahead:

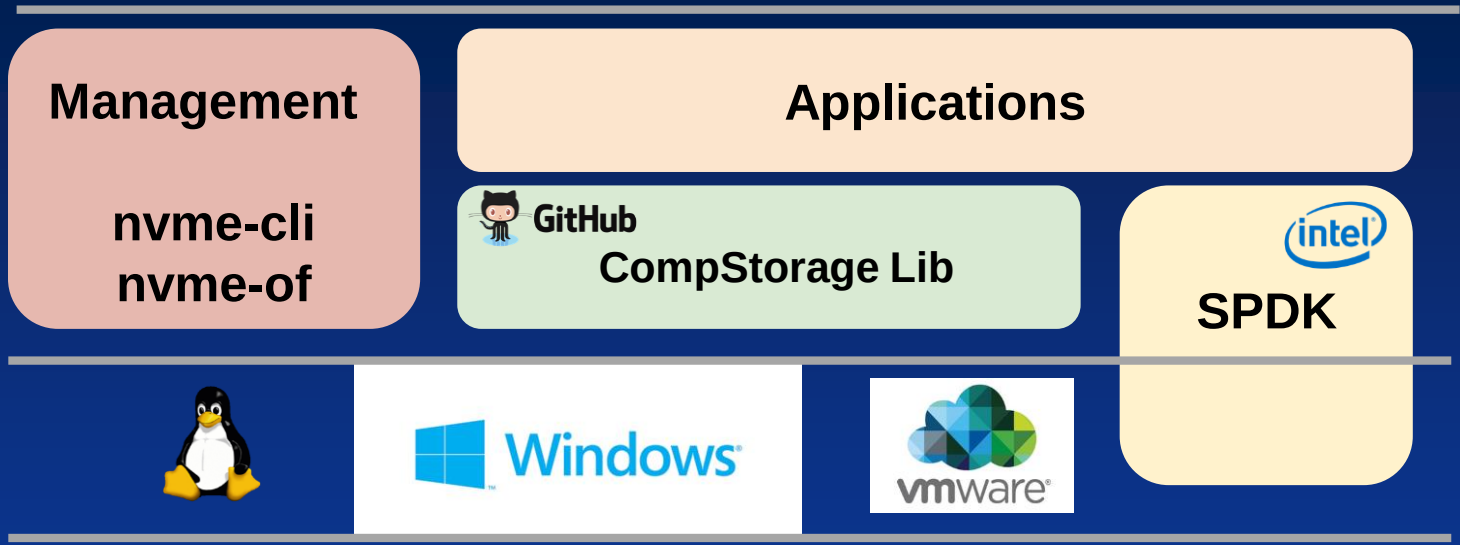
How do we ensure Computational Storage can scale?

Scaling from Hardware Perspective

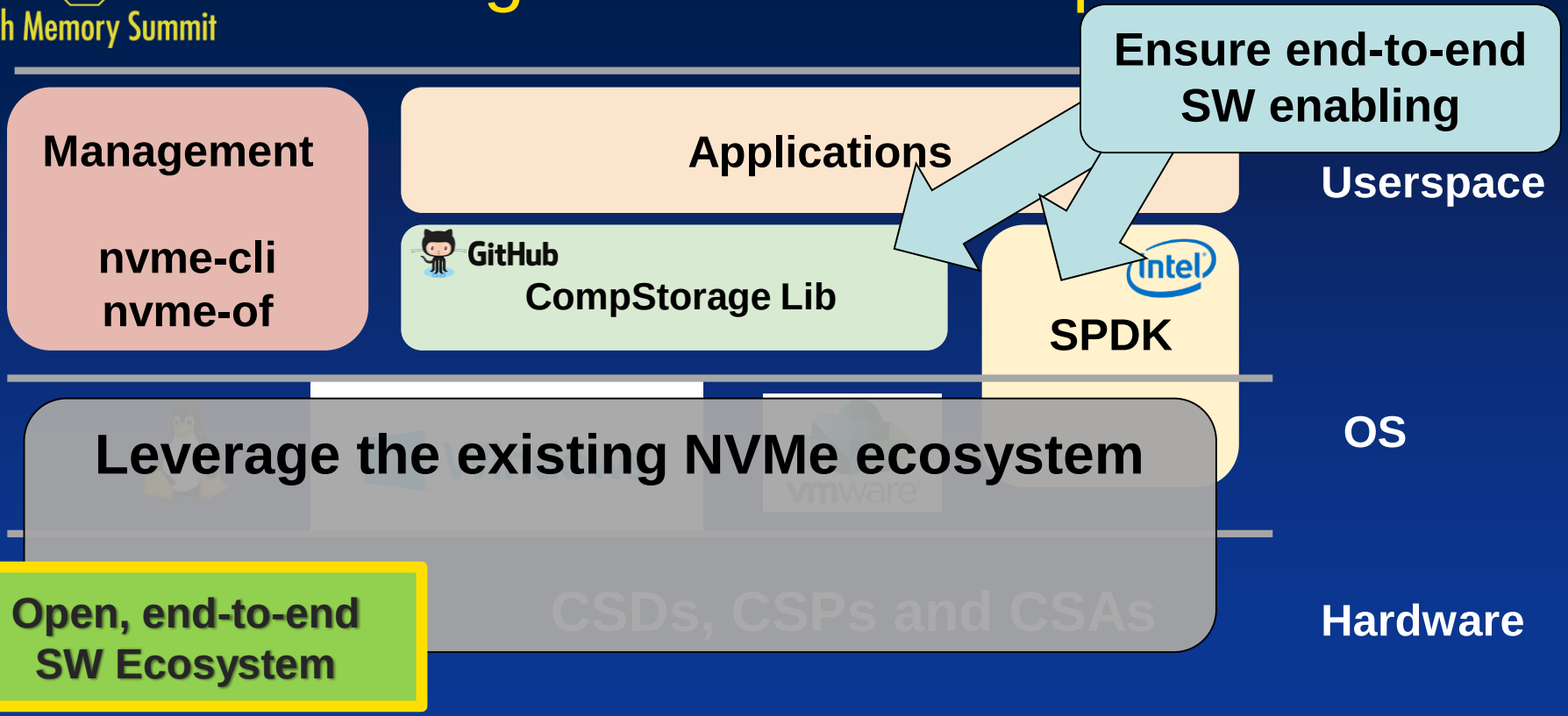
- Ensure storage interfaces can abstract the HW architecture
  - CSD, CSP, & CSA
- ***Remember***: Computational Storage is more than just an SSD
- Structure must allow for the HW Architecture to change without a need for the SW infrastructure to change



Scaling from a SW Perspective



Scaling from a SW Perspective



Drive toward Standardization

- Support near term products getting to market
  - Need broad support to establish an ecosystem
  - Keep it *simple!*
- Prioritize well-understood usage models
- Expand where Computational Storage adds value

Flash Memory Summit

Conclusion

Computational Storage is a promising technology that alters the storage architecture paradigm.

- Addresses inconsistency in Compute, Network & Storage performance advances
- Abstracts the HW Architecture from the SW Interface
- Brings consistency in the ecosystem *infrastructure* while enabling *innovation* for products in the space

Flash Memory Summit

Call to Action

- Work is underway to standardize the infrastructure
- Need participation from a variety of Software, Firmware and Hardware experts
- Plenty of opportunity to participate, influence and innovate in this space.

Engage.

https://www.snia.org/computational

Notices and Disclaimers

- Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration.
- No product or component can be absolutely secure.
- Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. For more complete information about performance and benchmark results, visit http://www.intel.com/benchmarks .
- Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit http://www.intel.com/benchmarks .
- Intel® Advanced Vector Extensions (Intel® AVX)* provides higher throughput to certain processor operations. Due to varying processor power characteristics, utilizing AVX instructions may cause a) some parts to operate at less than the rated frequency and b) some parts with Intel® Turbo Boost Technology 2.0 to not achieve any or maximum turbo frequencies. Performance varies depending on hardware, software, and system configuration and you can learn more at http://www.intel.com/go/turbo.
- Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.
- Cost reduction scenarios described are intended as examples of how a given Intel-based product, in the specified circumstances and configurations, may affect future costs and provide cost savings. Circumstances will vary. Intel does not guarantee any costs or cost reduction.
- Intel does not control or audit third-party benchmark data or the web sites referenced in this document. You should visit the referenced web site and confirm whether referenced data are accurate.
- Intel, the Intel logo, and Intel Xeon are trademarks of Intel Corporation in the U.S. and/or other countries.
- *Other names and brands may be claimed as property of others.
-