I/O Acceleration from the Bottom Up

How will new SSD technologies shape future data serving infrastructures?

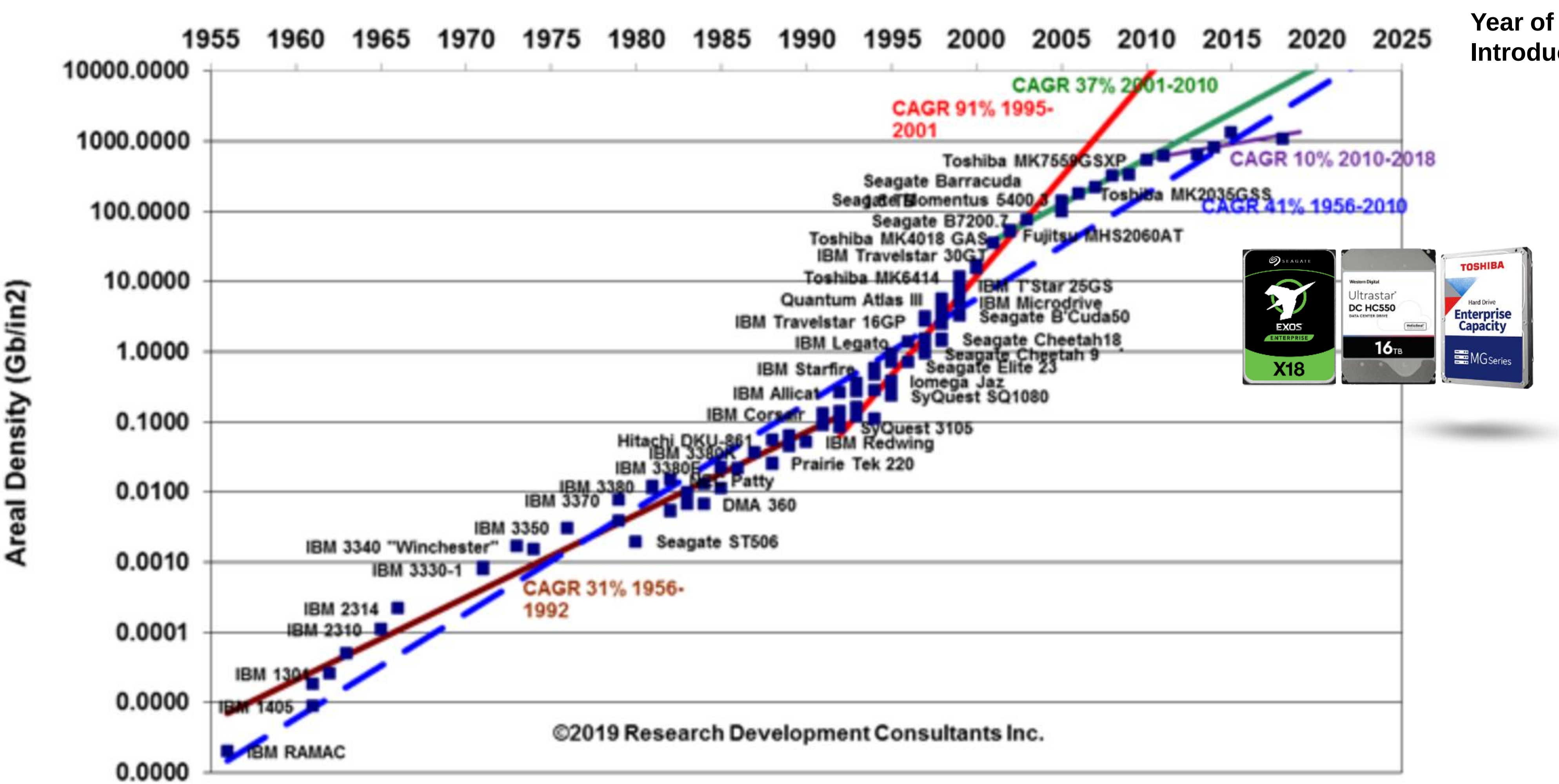
Sangyeun Cho

Memory Business
Samsung Electronics Co.

A little bit of history

Areal Density Trend, Rotating Media

Year of New Product Introduction

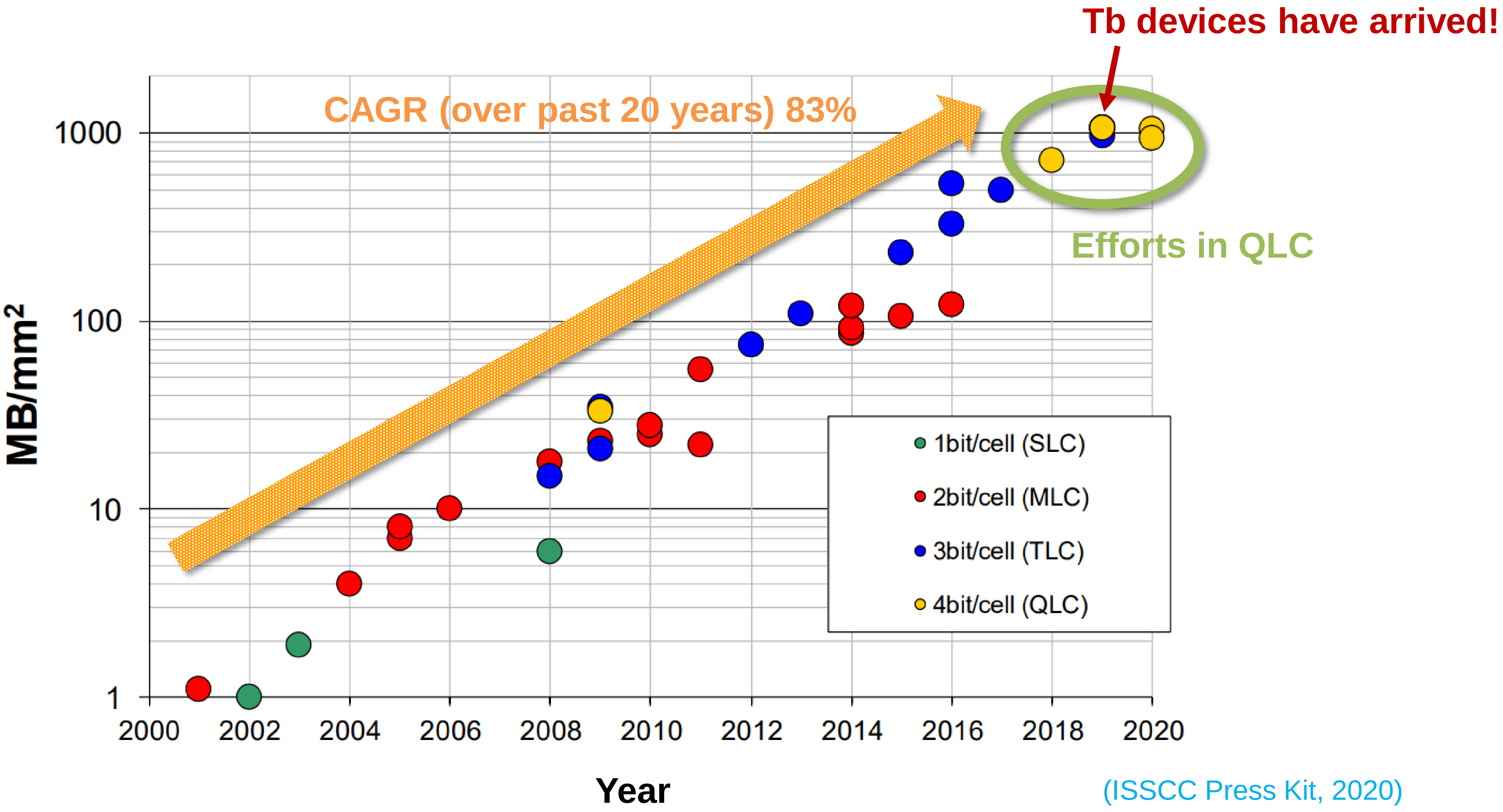


(16~18TB)

(By Ron Dennison, rondennison.com)

CHS (Cylinder, Head, Sector) Addressing
(3.75MB)

Areal Density Trend, NAND Flash Media



(ISSCC Press Kit, 2020)

Today's NAND Flash Memory (in Production)

	TLC	QLC	SLC
Die Capacity	**512Gb**	**1Tb**	**64Gb**
Areal Density	5Gbit/mm^2	7.53Gbit/mm^2	-
Page Read Latency	45μs	110μs	3μs
Program Throughput	82MB/s	18MB/s	160MB/s
Source	ISSCC 2019	ISSCC 2020	ISSCC 2018

Demise of Performance Hard Drives

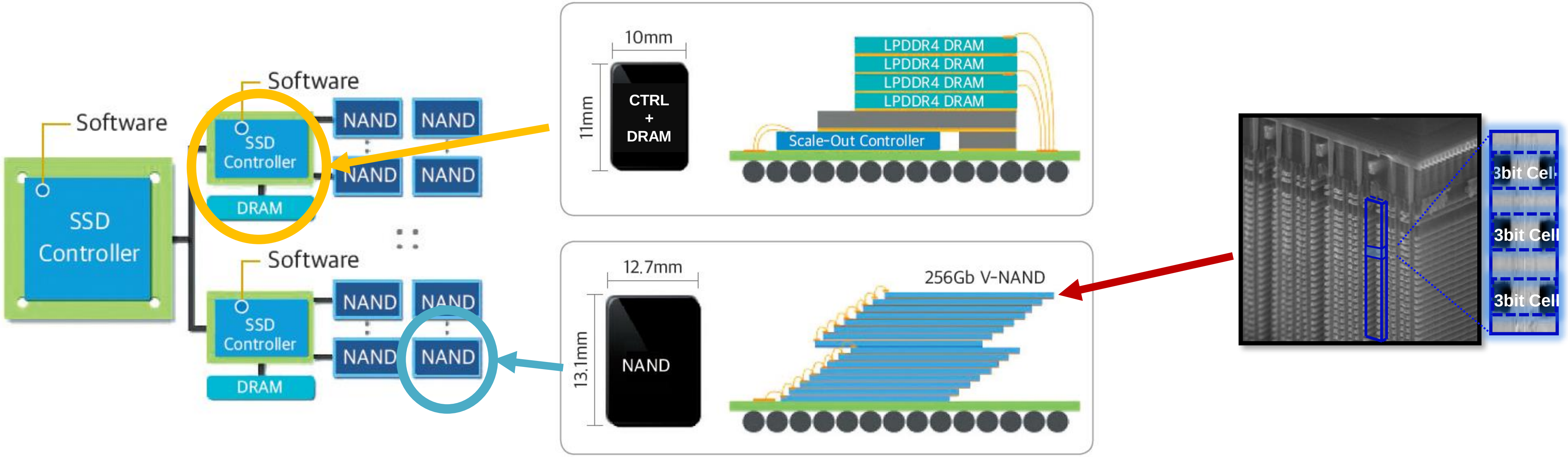
- **In 2016~2017, Samsung introduced industry's 1st enterprise SSDs built with 3D VNAND TLC**
  - **Status quo was to use planar SLC or eMLC**

- **Compelling MB/s, IOPS/$, IOPS/GB, and AFR advantages**
- **A 2.5" SSD offered capacity points from 0.5~16TB**

	Performance HDD	SSD (PM1633a, 2017)
Interface	**Dual-port SAS (6G~12G)**	**Dual-port SAS (12G)**
Density	250~600GB	0.5~16TB
Sequential Performance	<400MB/s	1,200MB/s (Read); 900MB/s (Write)
IOPS	<1K	200K (Read); 31K (Write)

Achieving High Density

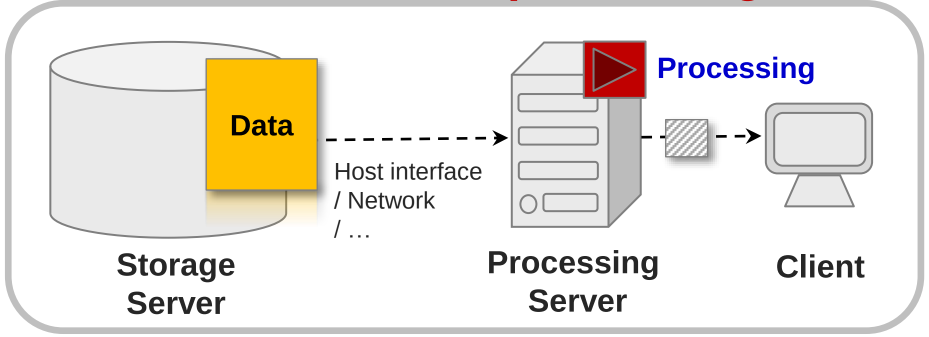
- **When mass-produced in 2017, 16TB PM1633a was the world's highest capacity drive (yes, including HDDs)**
- **A novel "scale-out" architecture**
  - **Main controller + many sub-controllers**
  - **Industry's 1st use of LPDDR4 DRAM in enterprise storage**



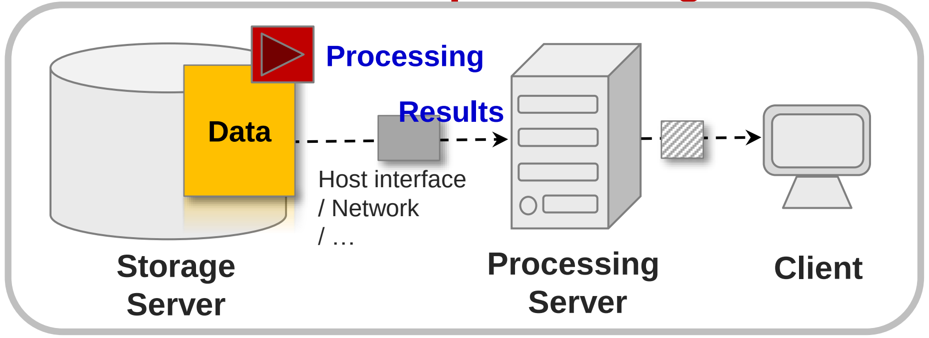
Short-circuiting data to compute

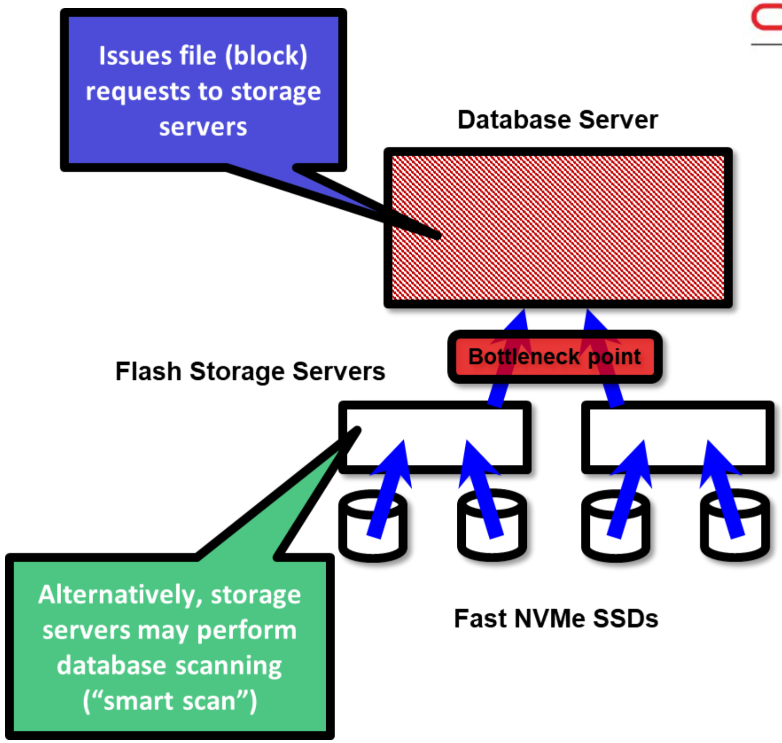
Moving Data vs. Moving Compute

Traditional data processing



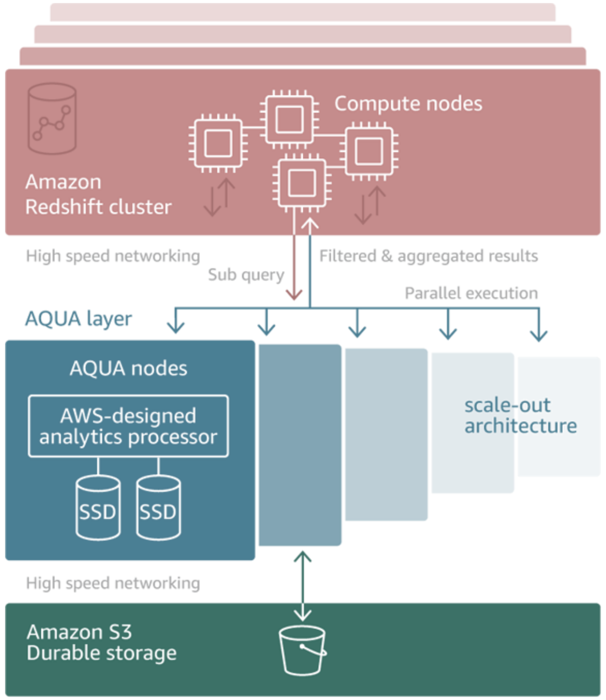
Near-data processing



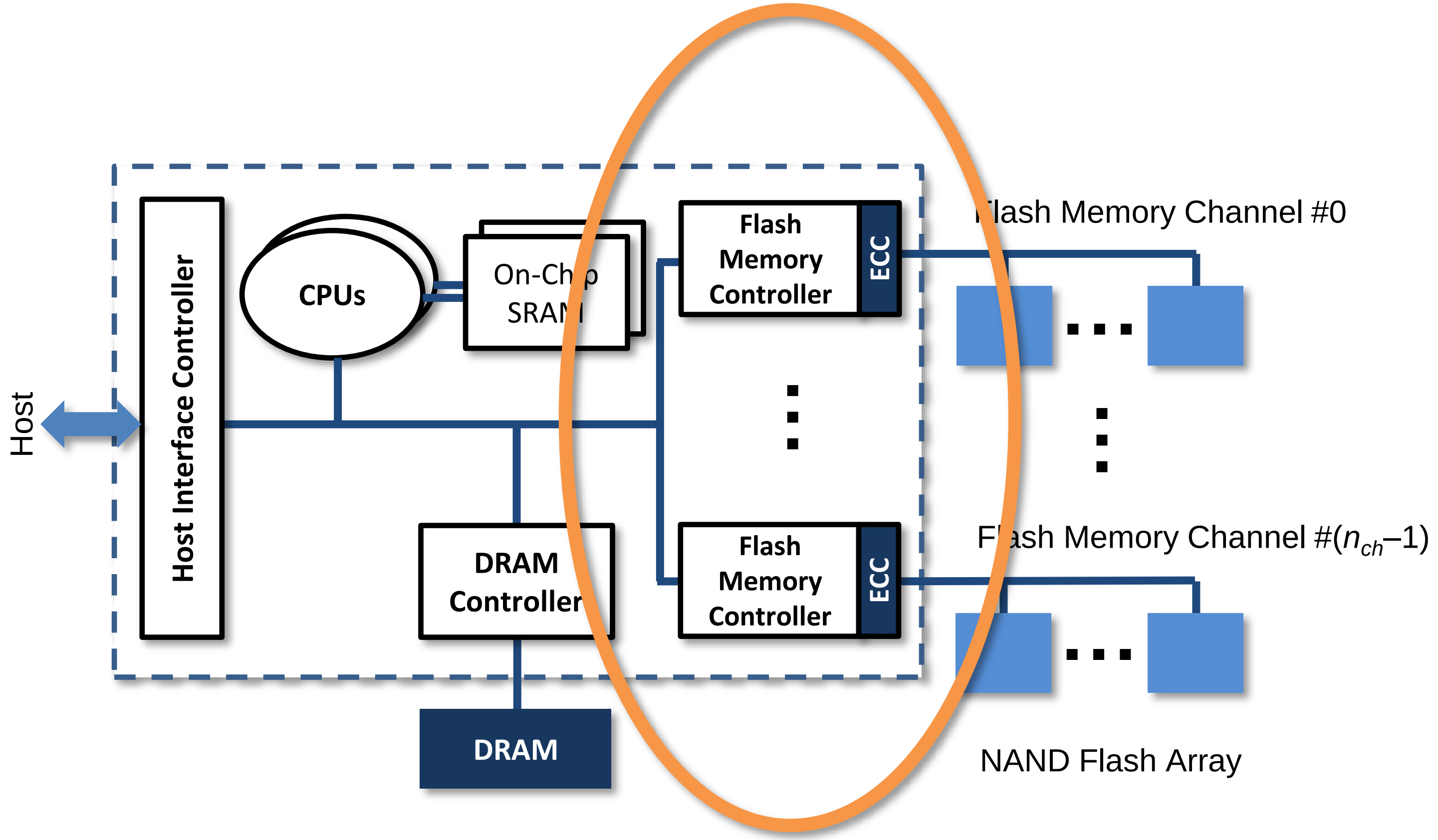


ORACLE
EXADATA

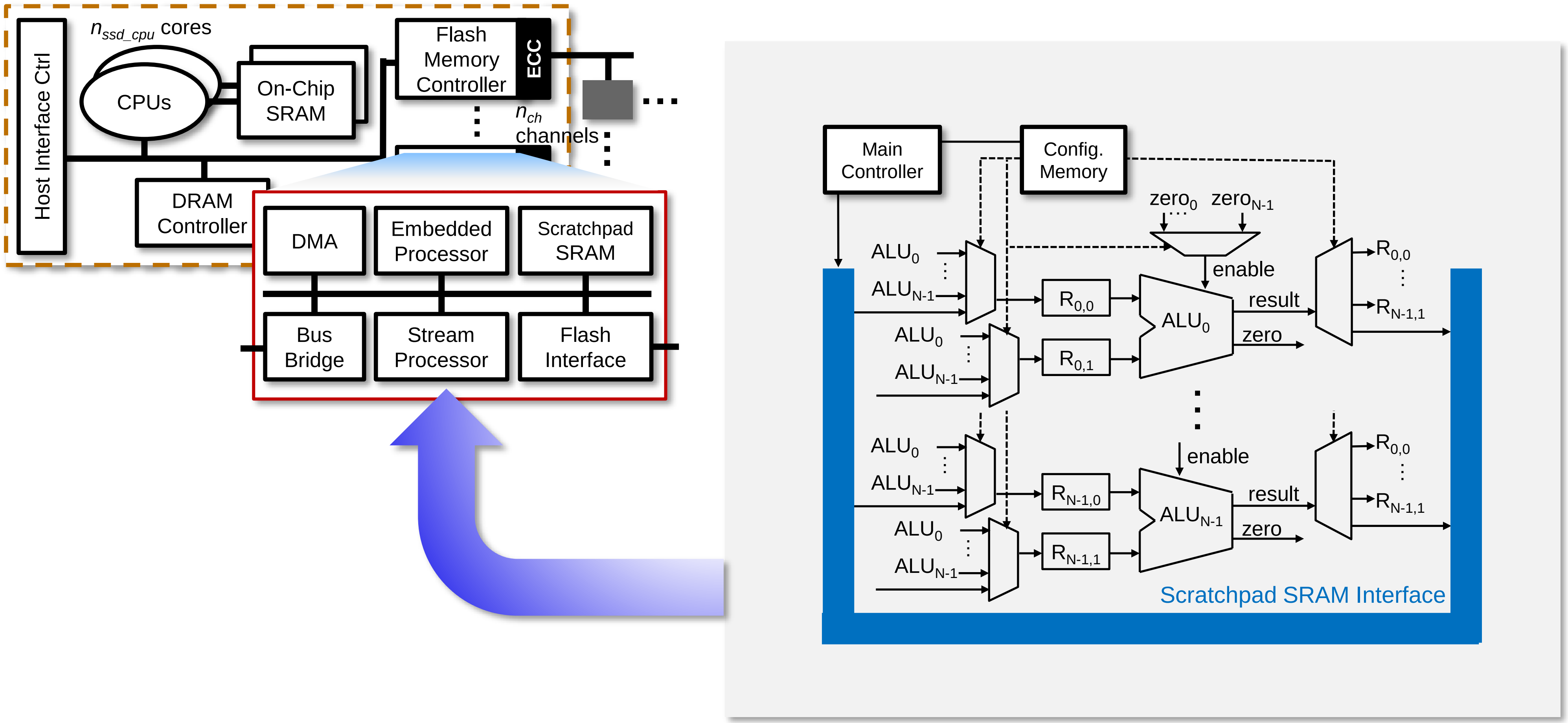
AWS AQUA Architecture



Pushing Compute to the Far End

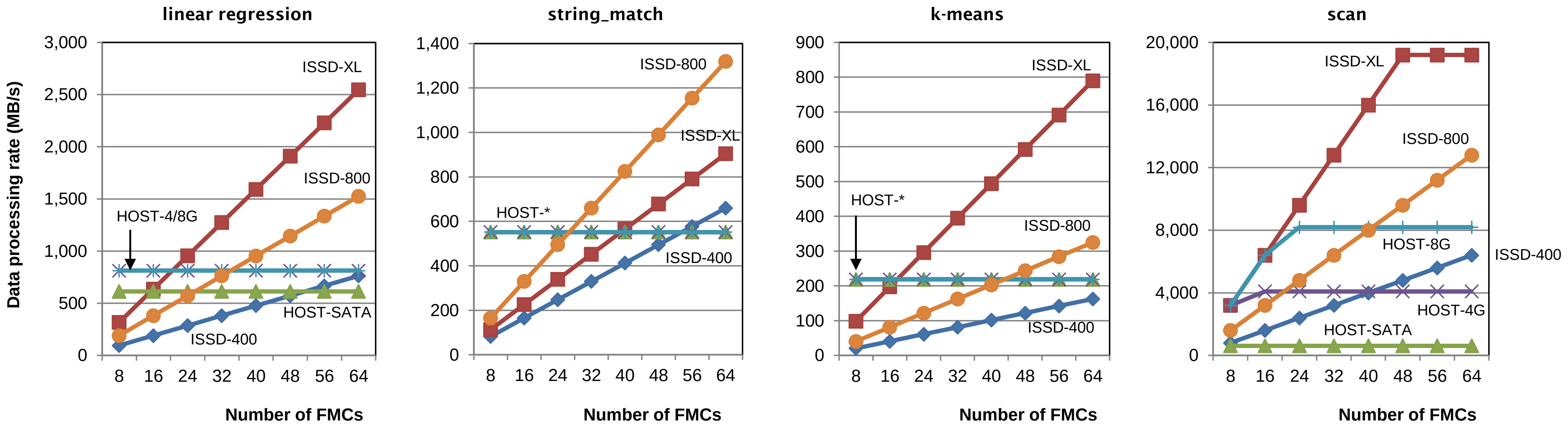


Pushing Compute to the Far End



[Cho, Park, Oh, Kim, Yi, and Ganger. “Active disk meets flash: a case for intelligent SSDs.” ICS 2013]

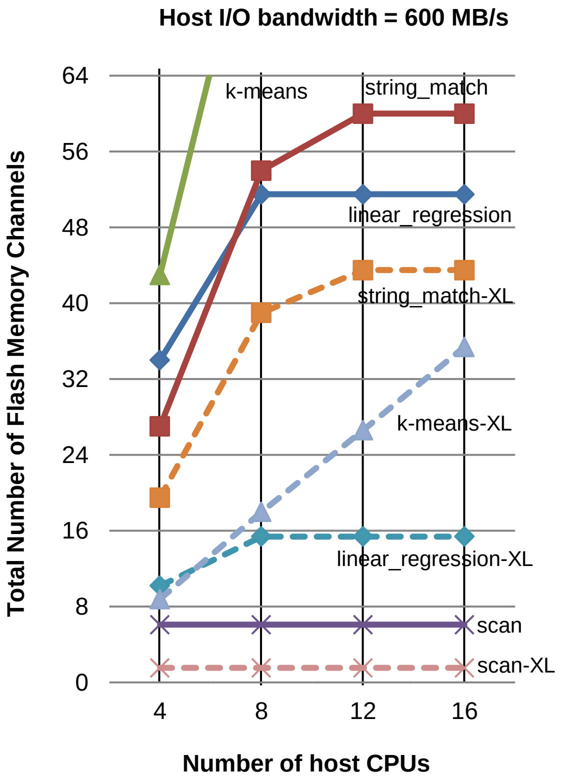
Data Processing Throughput

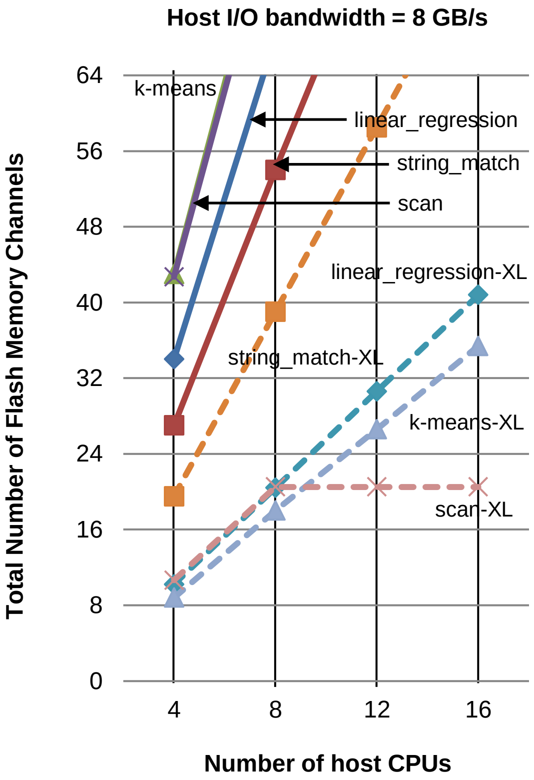


ISSD-XL: intelligent SSD with an accelerator (stream processor) per flash memory channel
ISSD-800: intelligent SSD with an embedded processor per flash memory channel running @800MHz
ISSD-400: intelligent SSD with an embedded processor per flash memory channel running @400MHz
Host-*: host server processing with I/O bandwidth of *

[Cho, Park, Oh, Kim, Yi, and Ganger. “Active disk meets flash: a case for intelligent SSDs.” ICS 2013]

Throughput Efficiency

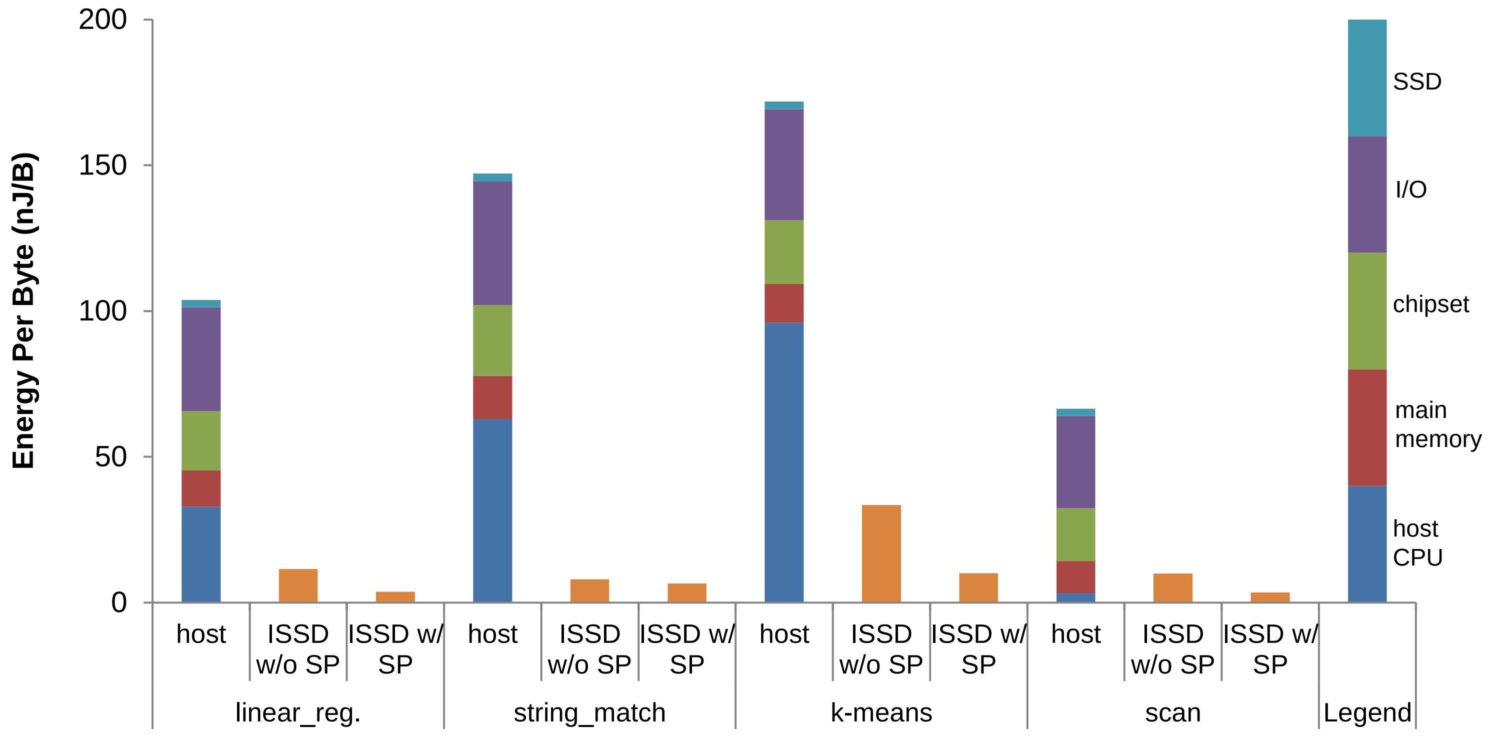




Solid lines capture “iso-performance” points with intelligent SSD processing (# channels) vs. host CPUs (# cores)
Dotted lines capture “iso-performance” points with intelligent SSD processing + acceleration vs. host CPUs

[Cho, Park, Oh, Kim, Yi, and Ganger. “Active disk meets flash: a case for intelligent SSDs.” ICS 2013]

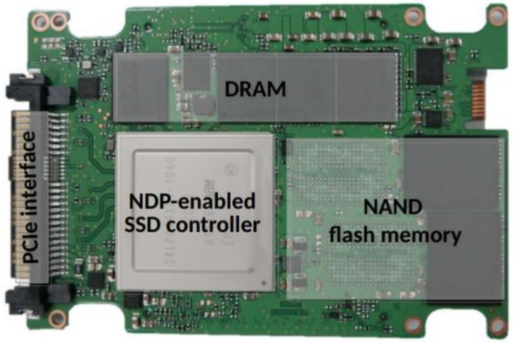
Energy Efficiency



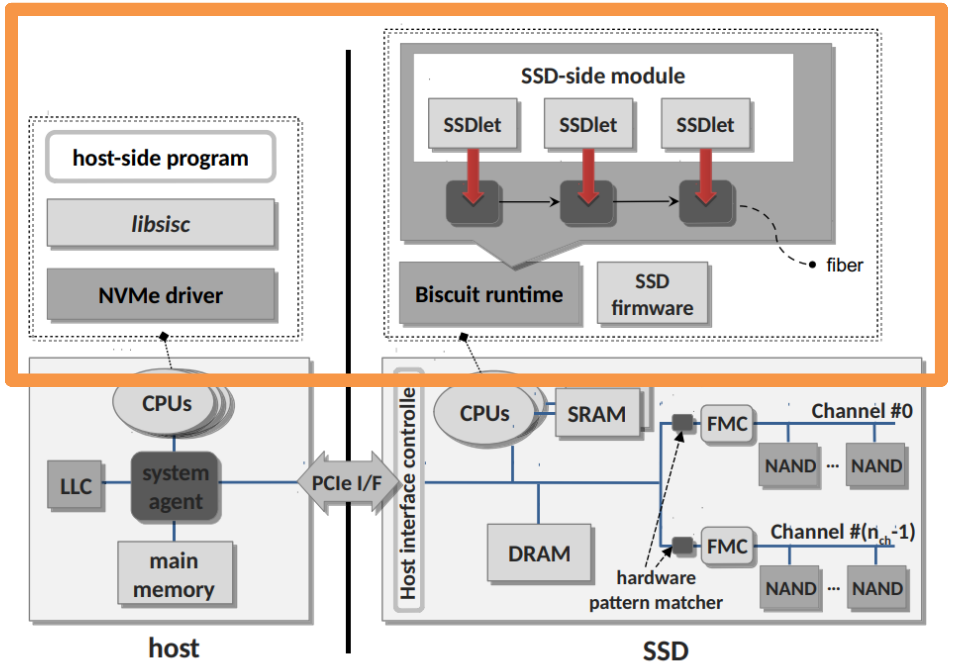
host: host server processing
ISSD w/o SP: intelligent SSD with an embedded processor per flash memory channel
ISSD w/ SP: intelligent SSD with a stream processing acceleration per flash memory channel

[Cho, Park, Oh, Kim, Yi, and Ganger. “Active disk meets flash: a case for intelligent SSDs.” ICS 2013]

Near Data Processing with Biscuit



- **An intelligent SSD for In-Storage Compute (ISC)**
- **Strong emphasis on programmability**
  - **User-friendly C++11 based programming model**
  - **Dynamic loading of user binary onto SSD**
  - **Seamless support for hardware acceleration**

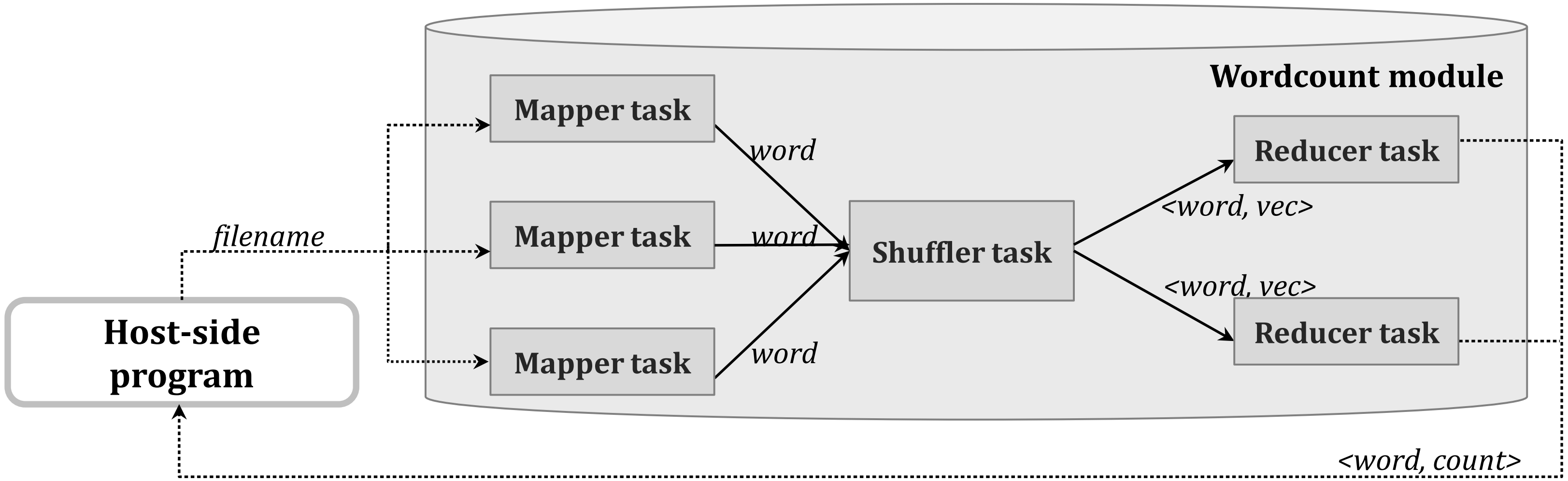


[Gu et al. "Biscuit: A Framework for Near-Data Processing of Big Data Workloads." ISCA 2016]

Biscuit Programming Model

- Biscuit follows a *data-flow model*
  – Data movement through ISC tasks determines their order of execution
  – On receiving all required inputs, an ISC task produces output and passes it to the next ISC tasks in the dataflow path

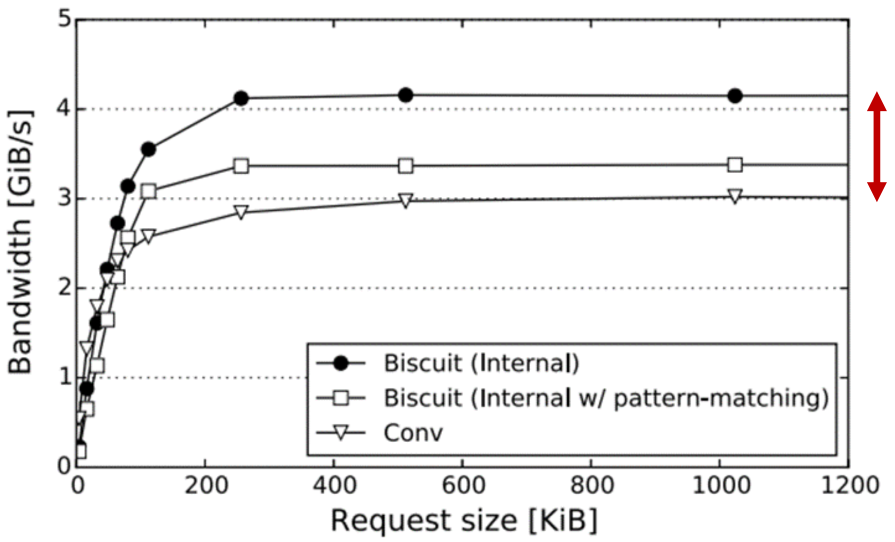
- A Biscuit program is composed of *ISC tasks* and *a host-side program*
  – An ISC task is a unit of work that runs on an ISC-enabled SSD
  – Both run concurrently in the SSD and the host, respectively



Biscuit: Basic Performance

Due to the interface speed limit (PCIe x4 in this case), SSD's internal bandwidth is ~30% higher

(Asynchronous I/O Bandwidth)

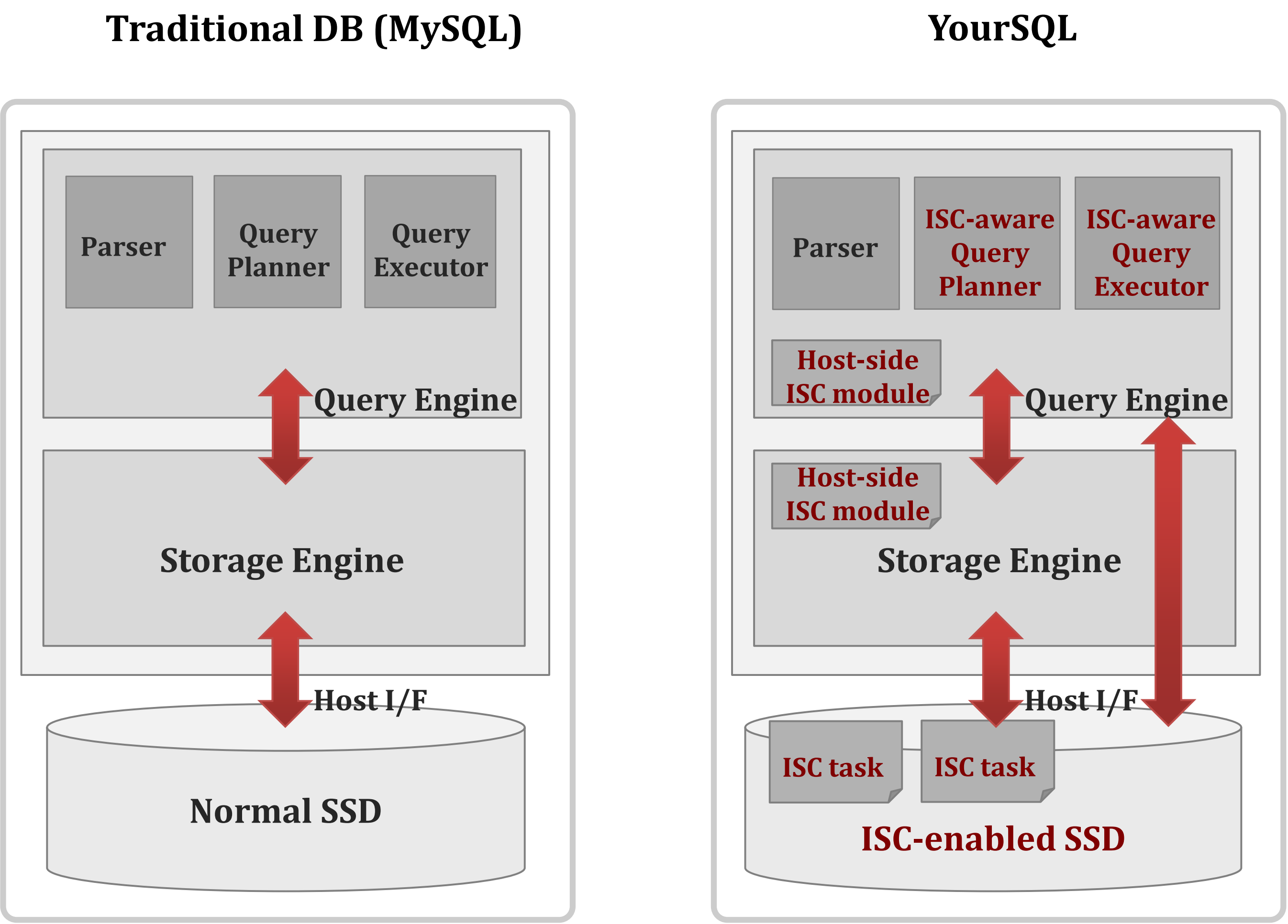


(Pointer Chasing Microbenchmark (sec))

	#threads	0	6	12	18	24
Exec. time (s)	**Conv**	138.6	143.5	152.5	154.9	155.0
	Biscuit	124.4	124.0	123.3	123.9	123.5

Data inspection and I/O inside the SSD results in 10~20% reduction in latency + resilience against host CPU loads

YourSQL on Biscuit

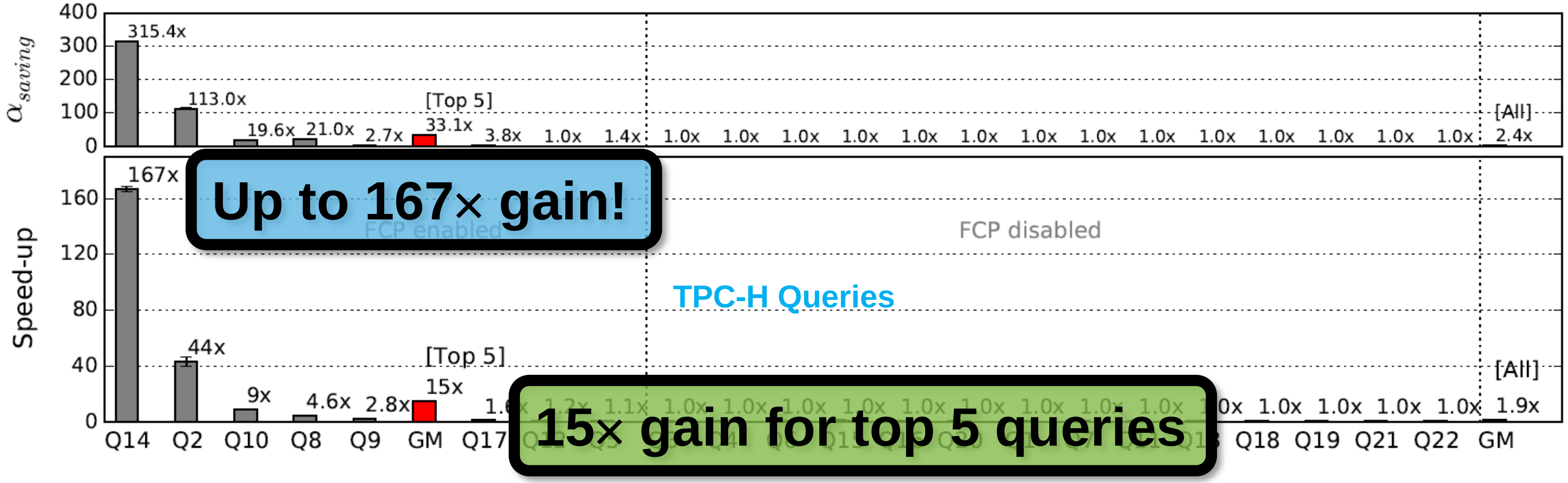


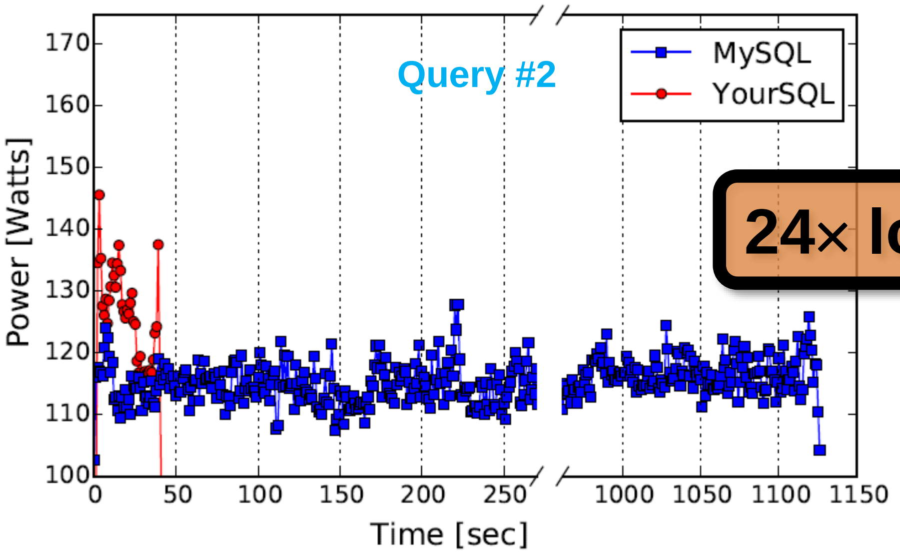
Key design considerations

- Partitioning of host/ISC tasks
- Defining interfaces between the host and ISC tasks
- Optimized query planner for ISC
- Reorganized datapath for ISC

[Jo et al. "YourSQL: A High-Performance Database System Leveraging In-Storage Computing." VLDB 2016]

Evaluation Results

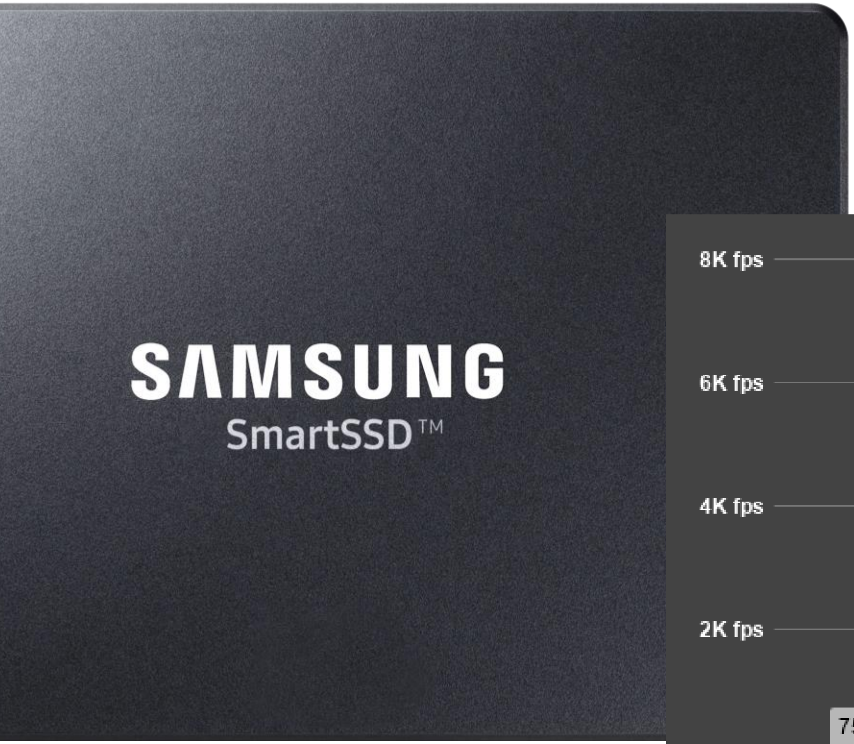




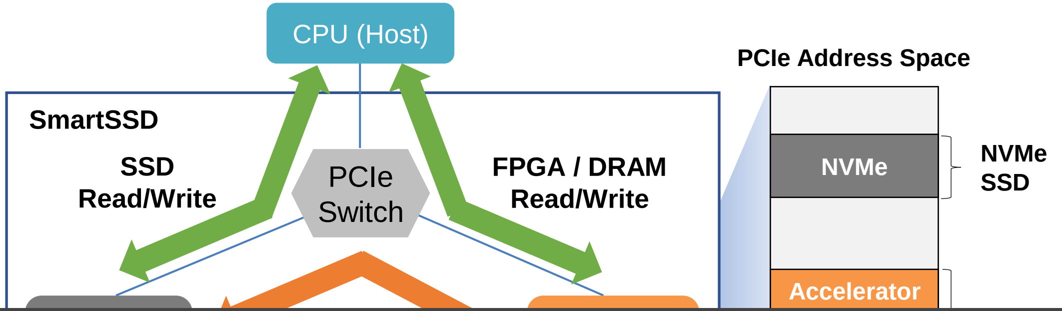
On dual Intel Xeon E5-2640 w/ 64 GB DRAM
Samsung PM1725 1TB SSD
MariaDB 5.5.42 w/ modifications using Biscuit
Workload is TPC-H w/ a scale factor of 100

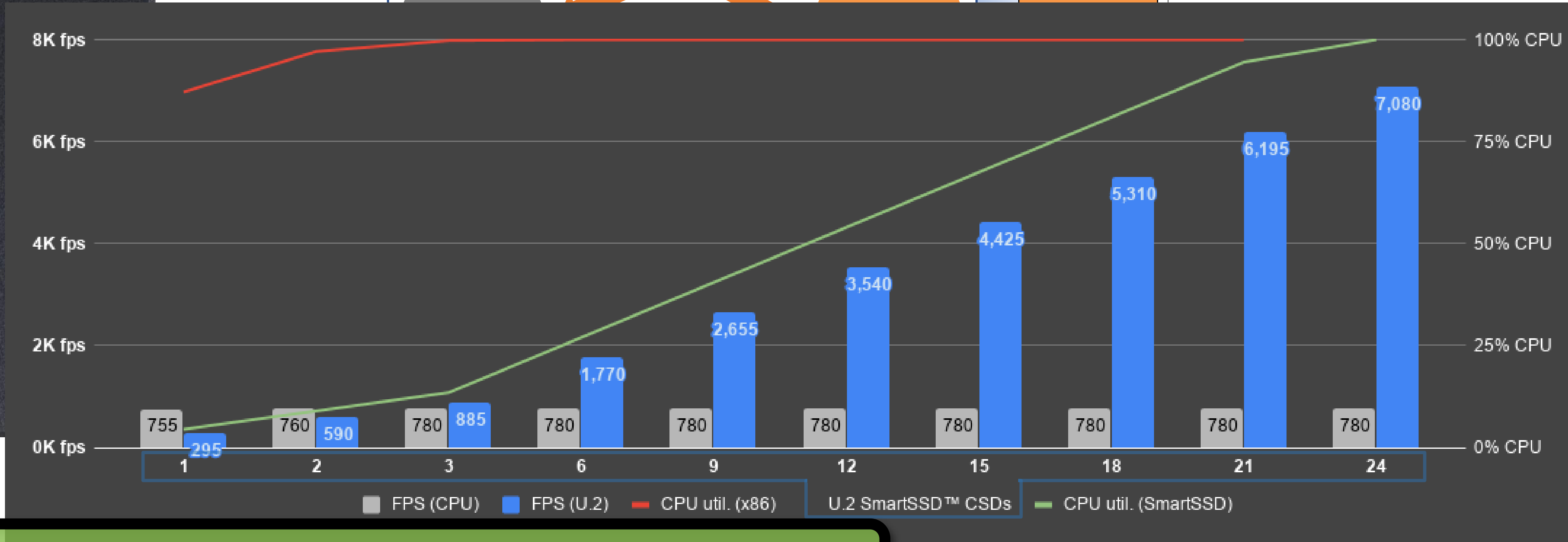
[Jo et al. “YourSQL: A High-Performance Database System Leveraging In-Storage Computing.” VLDB 2016]

Samsung SmartSSD™



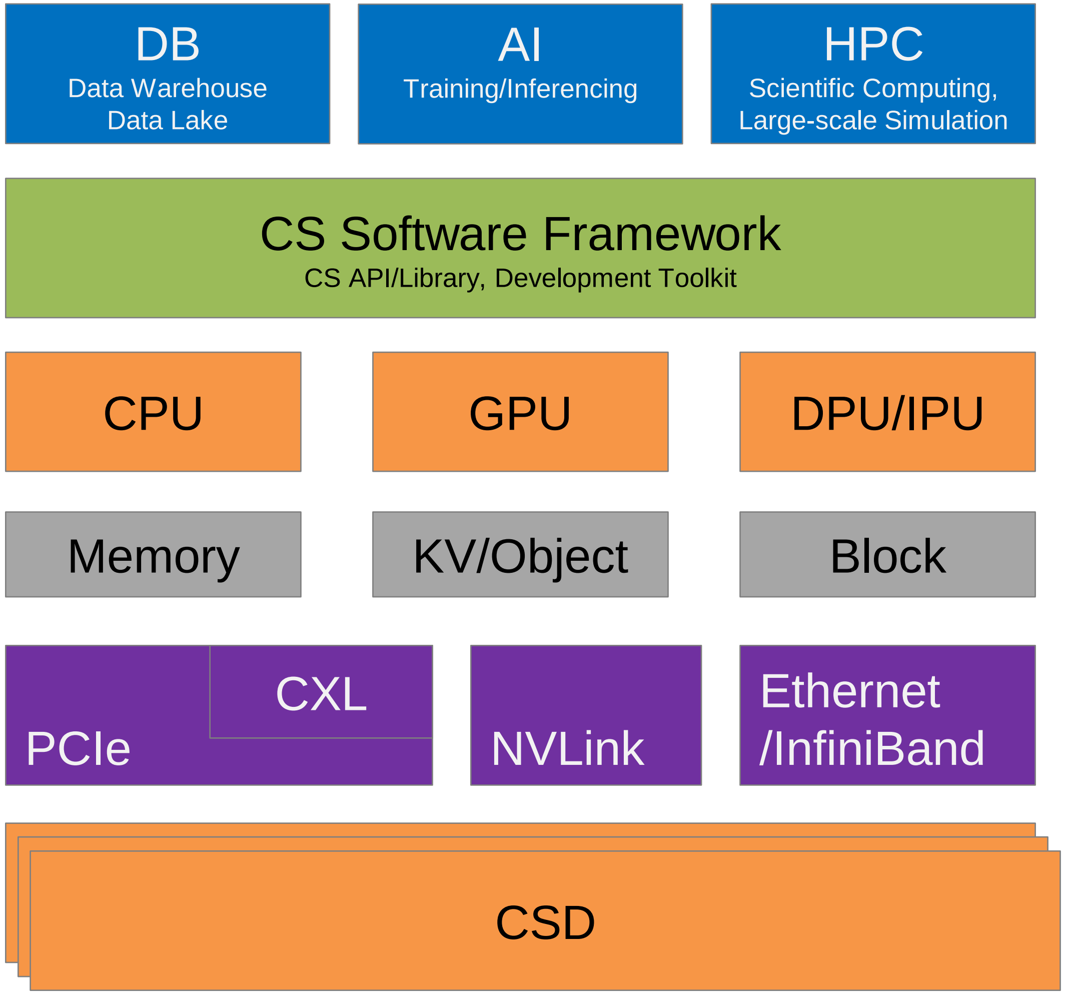
(samsungsemiconductor-us.com/smartssd)





Performance scales as we add SSDs

Moving Forward



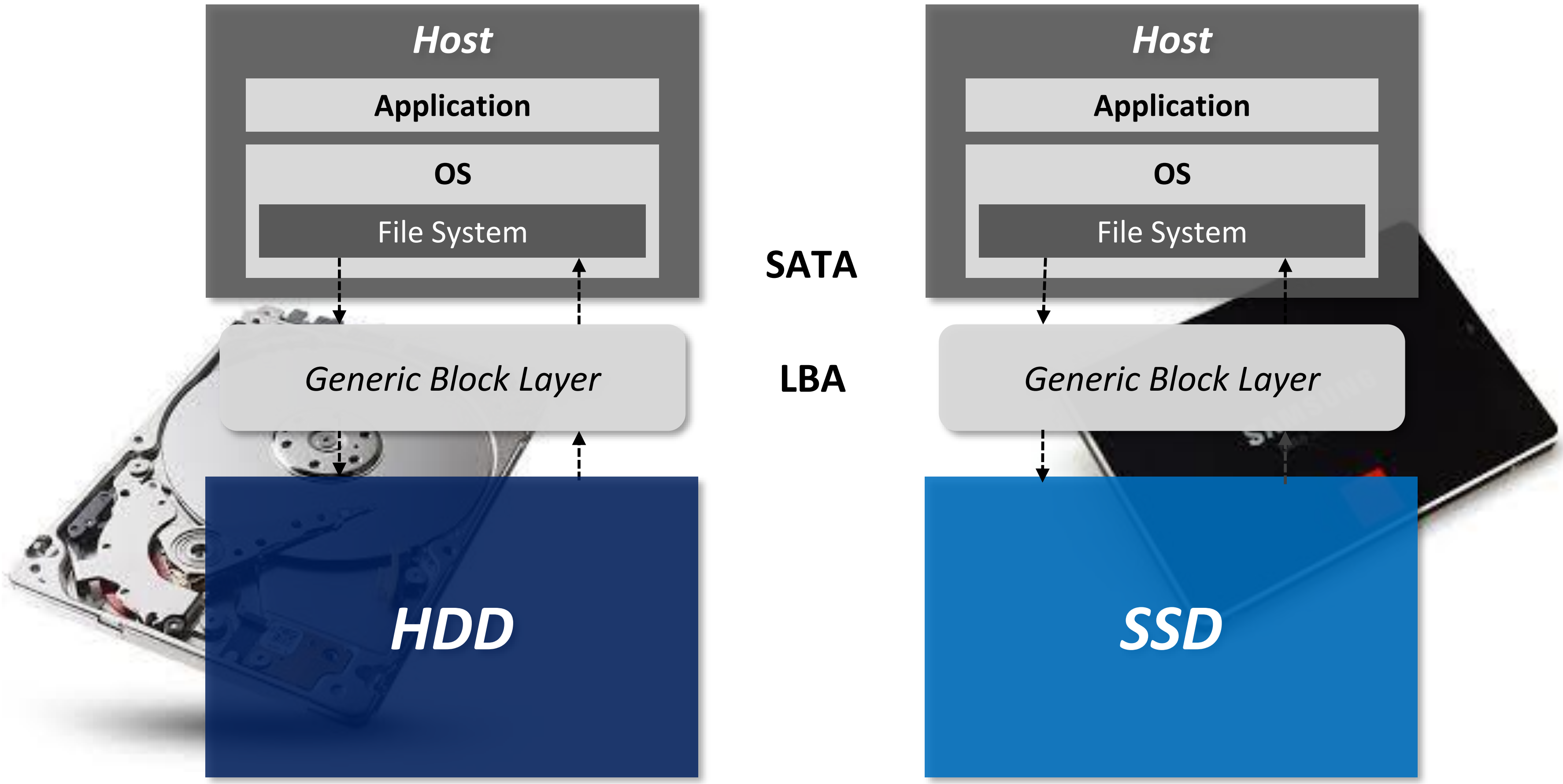
- **“Computational Storage” is being standardized at SNIA/NVMe**

- **What target applications?**
- **What programming models?**
- **How to coordinate and maximize the use of all platform resources?**
- **Which data access mode?**
- **Which interconnect technologies?**
- **How to best utilize many computational storage devices?**

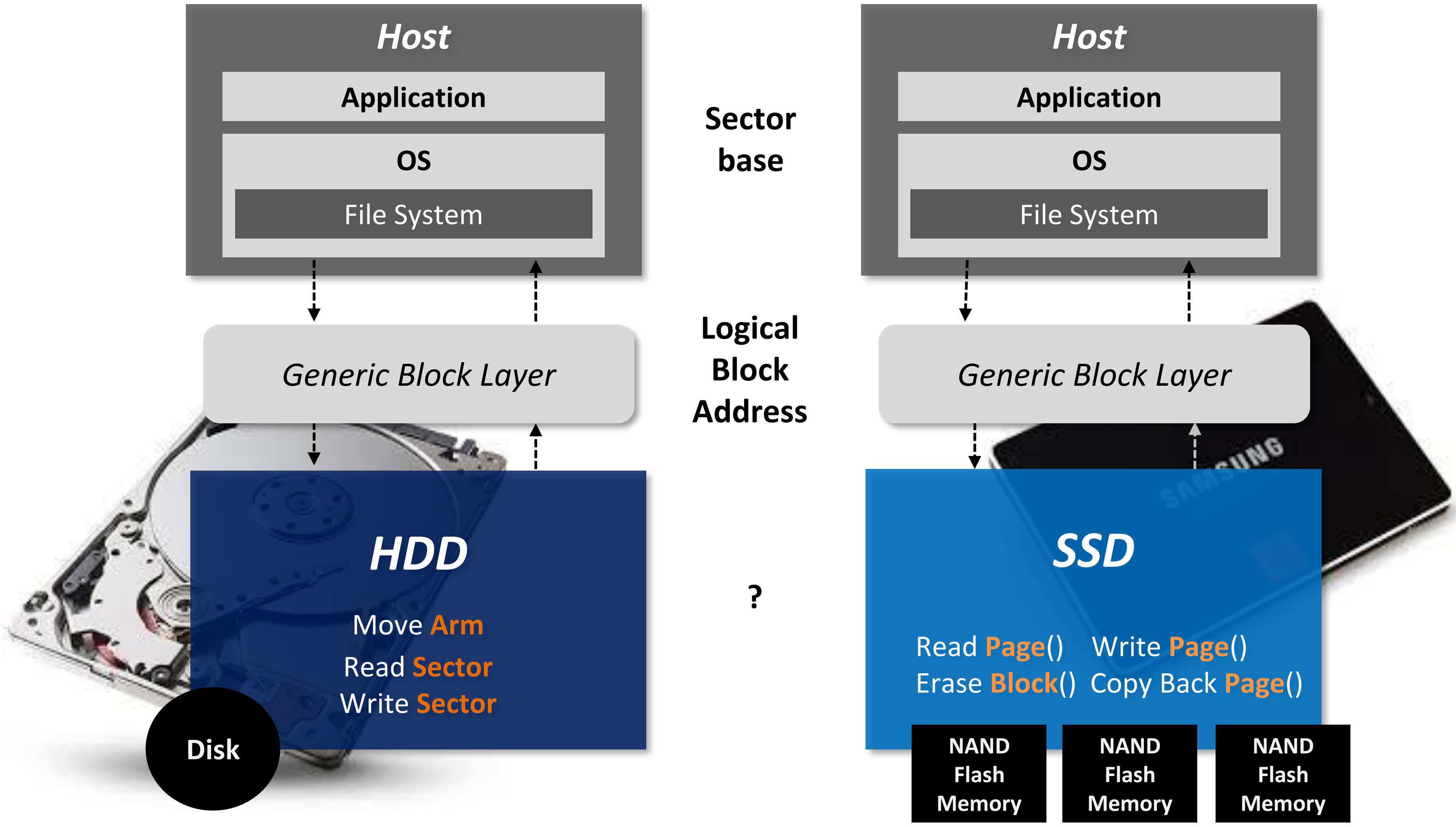
Getting the most from the media

Logical View of Physical Media

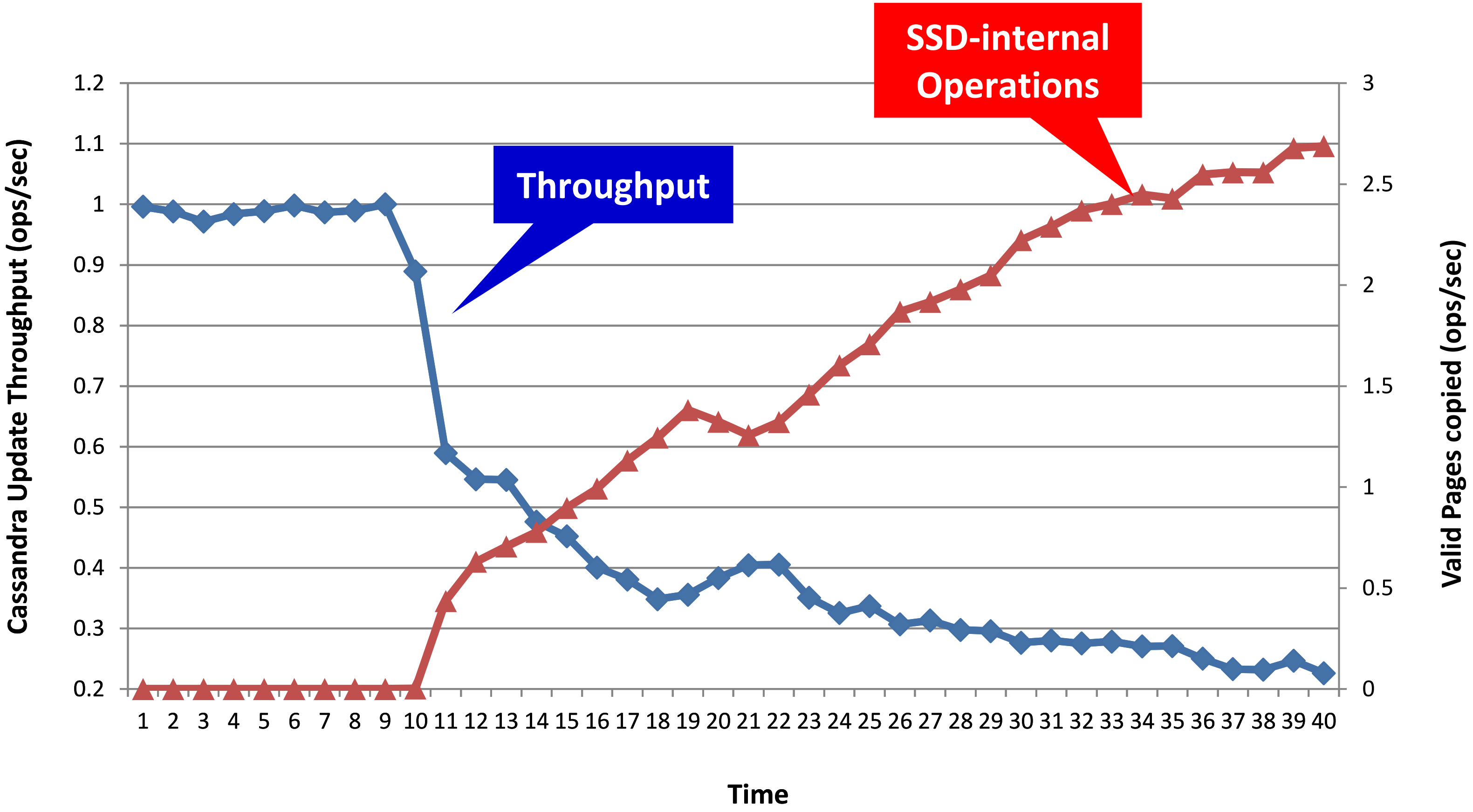
- **The LBA interface (introduced circa 1986) has helped straightforward switching to SSD**



Logical View of Physical Media

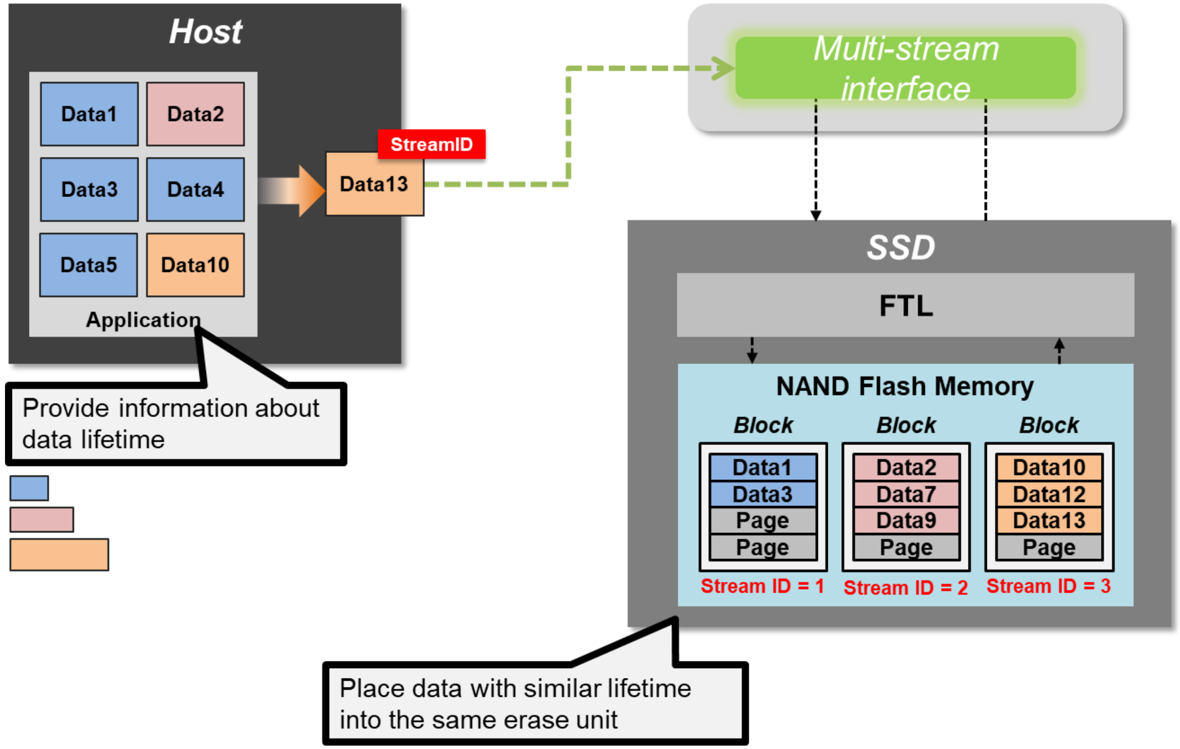


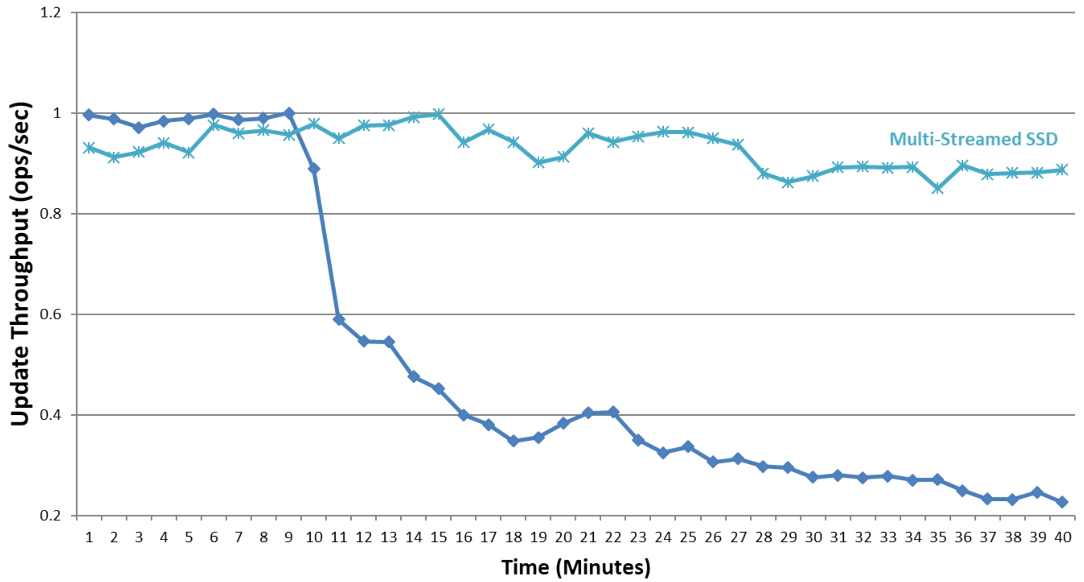
Fresh vs. Sustained Performance



[Kang, Hyun, Maeng, and Cho. “The Multi-streamed Solid-State Drive.” USENIX HotStorage, 2014]

Multi-Streamed SSD





[Kang, Hyun, Maeng, and Cho. "The Multi-streamed Solid-State Drive." USENIX HotStorage, 2014]

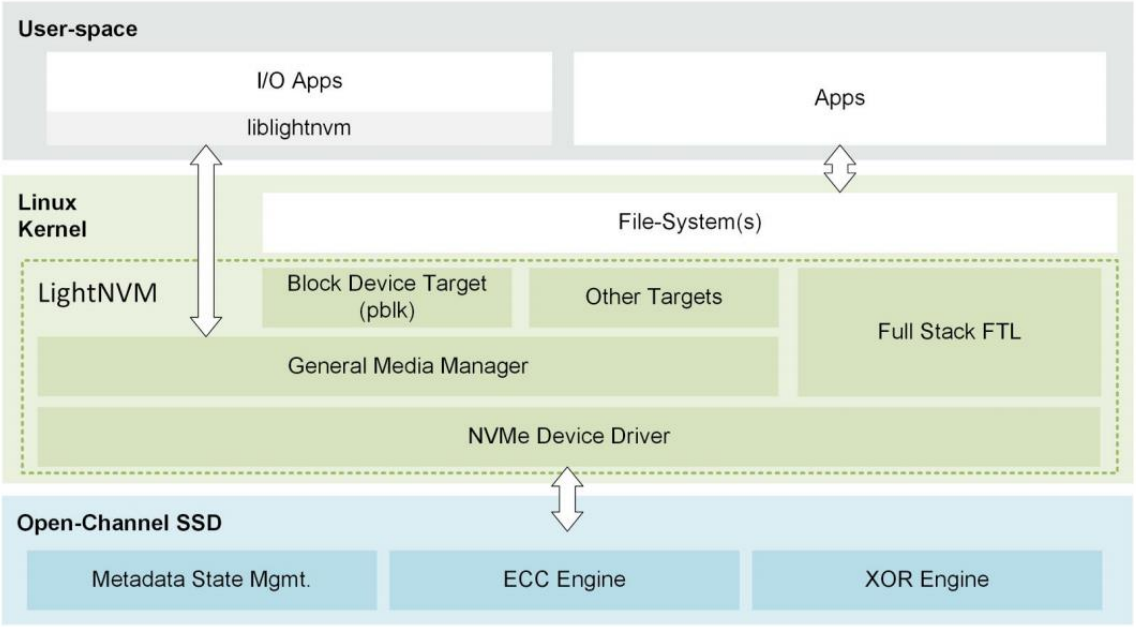
- **Published at HotStorage 2014**
- **Standardized in 2017 (SAS/NVMe)**
  - **Linux support since 2017**
- **Product debut in 2016~2017**

Simple, intuitive, additive model;
Model concrete enough to predict effects

Model is still abstract; host can't control data placement on specific physical units

Open-Channel SSD

- **Philosophy-wise, OC-SSD aims to expose the media to the host software for direct management**
  - **Eliminate (parts of) FTL and give full control of data placement and access schedule to media units**



["Open-Channel Solid State Drives NVMe Specification." Revision 1.2, April 2016]

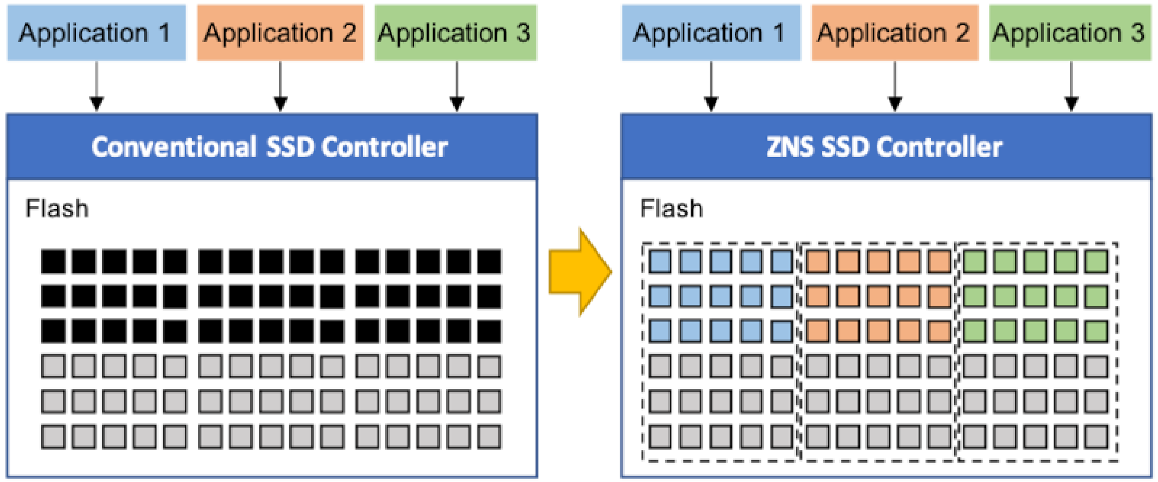
Host has complete control over data placement on NAND flash media (no LBA); Opportunities exposed for "cross-layer" optimizations between applications, file system, and FTL

Media idiosyncrasies underestimated;

Would you go back to CHS addressing from LBA?

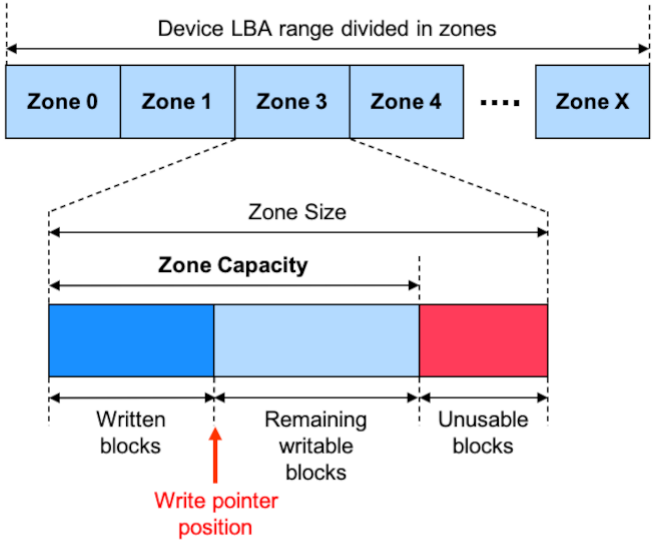
Zoned Namespace (ZNS) SSD

- **SSD capacity is split into "zones" that are sequentially written**
  - **An SSD zone is analogous to that of shingled magnetic recording HDDs**



Regular SSD: Device controls data placement

ZNS SSD: Applications control data placement in zones



(zonedstorage.io)

Host has control over data placement on NAND flash media;
Complicated media management resides within the SSD

Host software must be aware of zones (SMR support is leveraged);

Design trade-offs still being explored

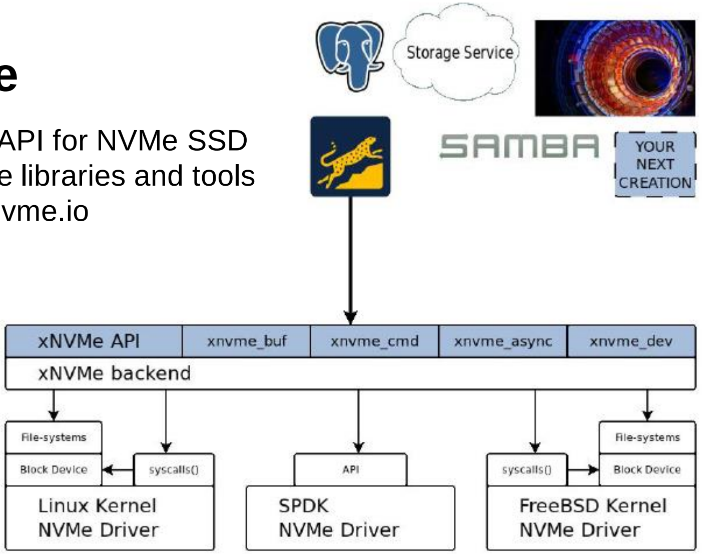
Moving Forward

Samsung Introduces Its First ZNS SSD With Maximized User Capacity and Enhanced Lifespan

Maximum available storage capacity and 3-4x longer lifespan enable server systems to run big data and AI applications more reliably and efficiently

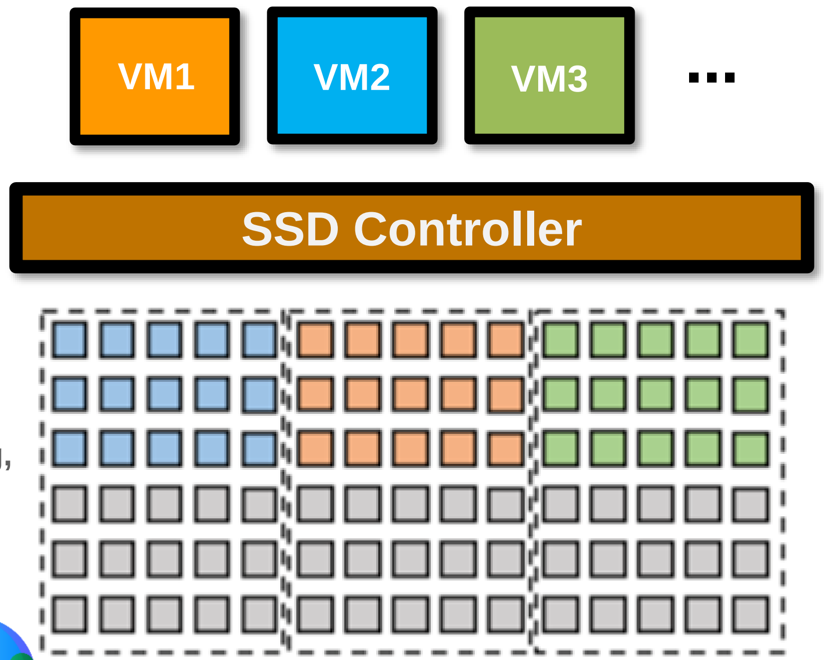
xNVMe

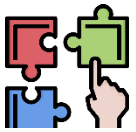
- Unified API for NVMe SSD
- Software libraries and tools
- http://xnvme.io



- **ZNS SSDs are available and are poised to offer strong use cases for large storage systems**
  - Very concrete interface
  - Good fit for many-bit cell technologies
- **Software availability and readiness remains a challenge for users**
- **More end-to-end software building and design trade-off studies are needed**

Physical Isolation of Storage Resources

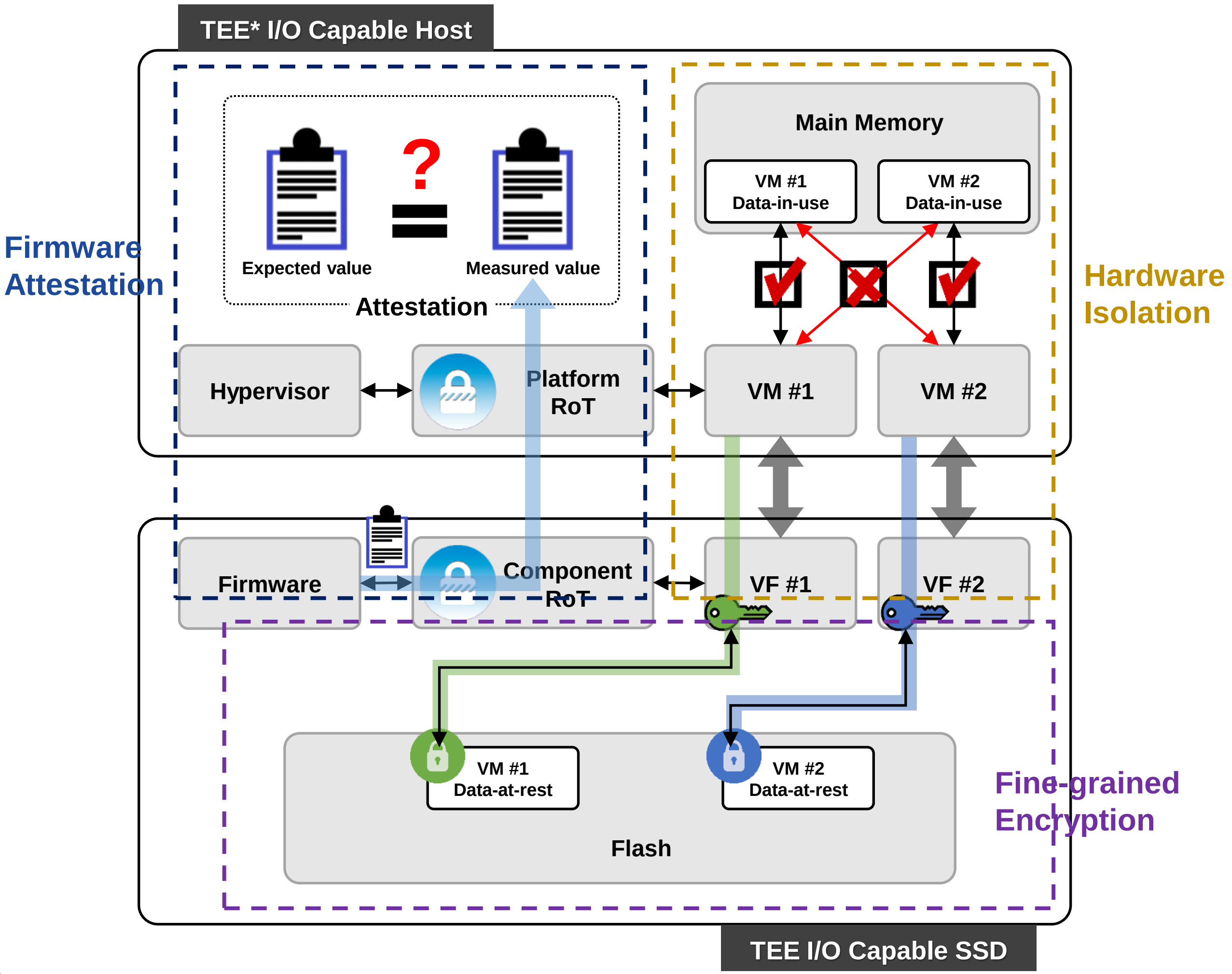




Capacity partitioning, data placement

Hardware acceleration of I/O and key functions

Per partition (VM) bandwidth allocation,
per partition performance QoS/schedule guarantee,
per partition house keeping, ...



Outro

- SSDs offer the density and performance required by modern workloads and infrastructures
  - In turn, SSD idiosyncrasies affect how systems are designed
- System changes are expected to realize ideas around SSDs
  - Short-circuiting of data and compute
  - NAND flash media aware storing/retrieving of data
  - Hardware-level isolation support for multi-tenancy
- Future SSDs offer system level optimization opportunities
  - Further end-to-end software building efforts are needed
  - Novel data-compute mapping/coordination ideas are wanted

I/O Acceleration from the Bottom Up

How will new SSD technologies shape future data serving infrastructures?

Sangyeun Cho

Memory Business
Samsung Electronics Co.