




# How NVMe and 3D XPoint Will Create a New Datacenter Architecture

Emilio Billi  
CTO  
A3Cube Inc

## The Storage Paradigm Shift

To understand the future of the storage-memory devices we need to consider two things:

### First



**All Computation is the same (\*)**  
(Results = Math + Data)

↓

**To get results faster, two things need to happen:**

- Faster Data (Reducing data access latency)
- Faster Math (Reducing computation latency)

(\*) David A. Patterson, Daniel W. Hillis, Seymour Cray & Others...

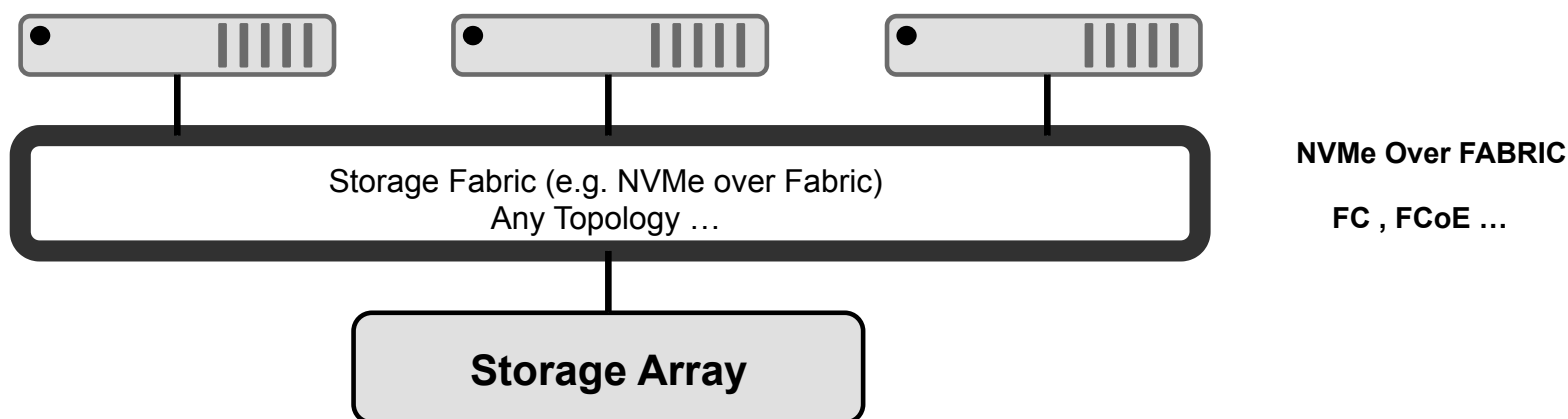
### Second

Modern applications like Data Analytics, Machine Learning Training, Data Mining ... are using storage actively in the same way that past applications used computation

**We are moving from HPC to HPD in all the everyday operations**

We need to start with a little bit of theory

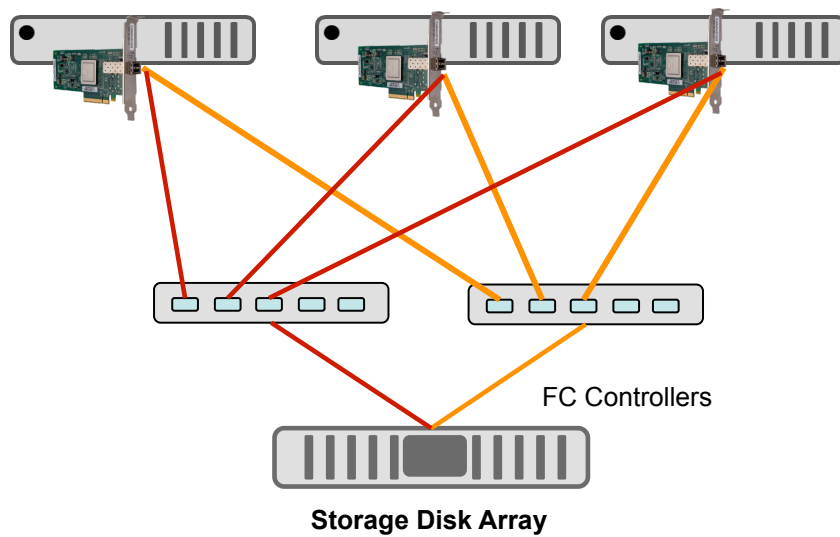
## Storage Architecture: Symmetric Storage Access



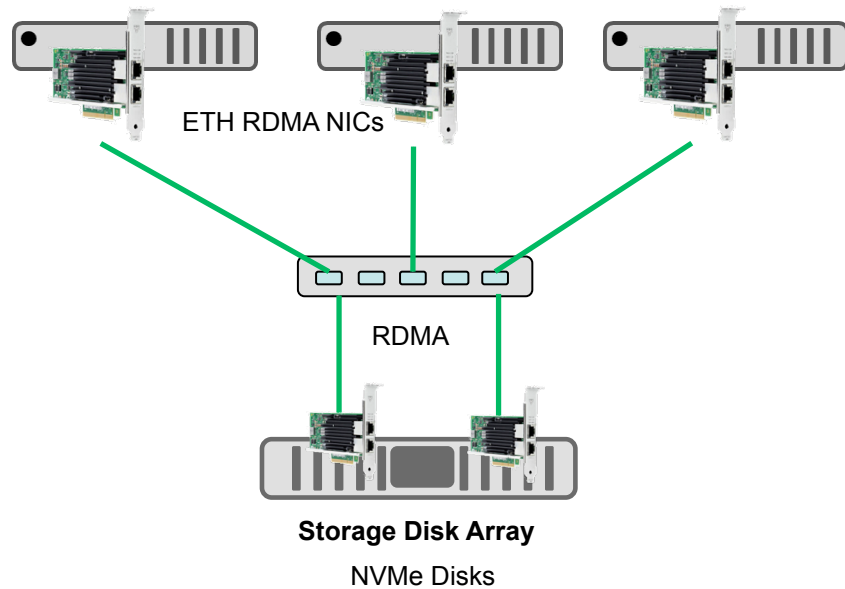
- Old Type of Storage Shared Architecture (The FC concept with a modern variants)
- Difficult to optimize at application level and application tuning
- No proximity algorithm
- Generic Type of Storage (No application specific optimization)

We need to start with a little bit of theory (cont.)

**Fibre Channel  
Servers**

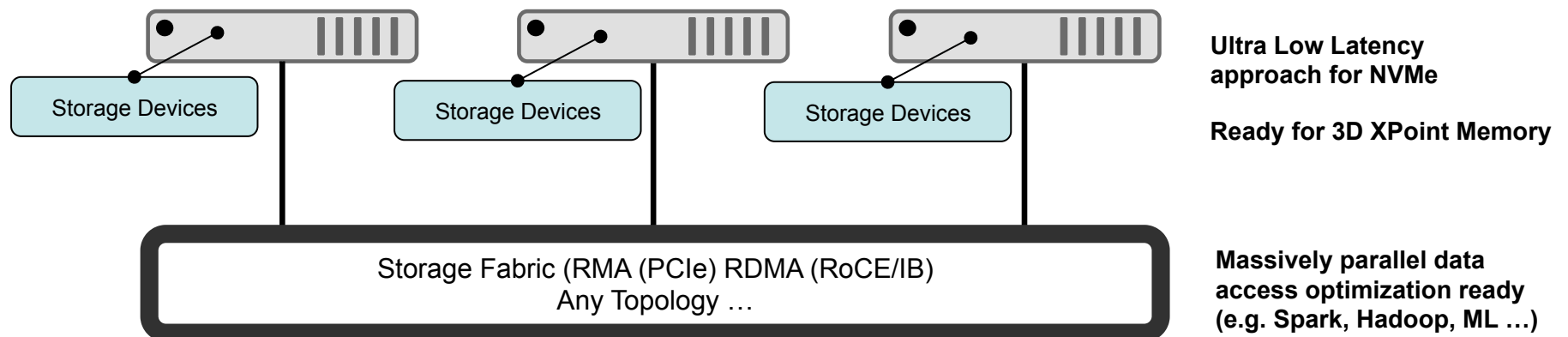


**NVMe Over Fabric  
Servers**



## We need to start with a little bit of theory (cont.)

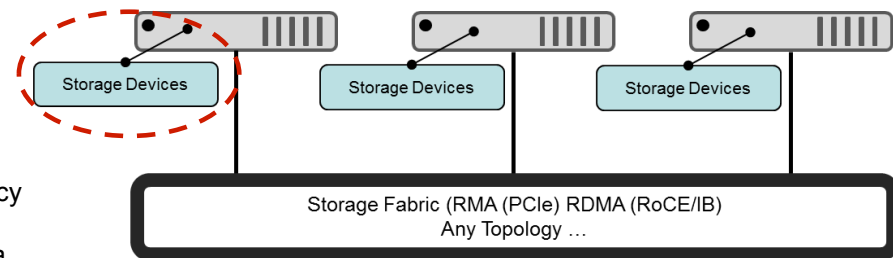
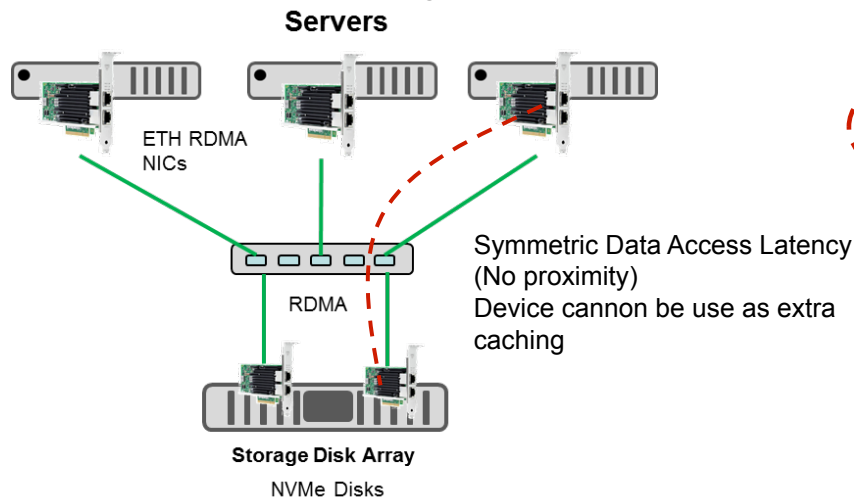
### NUFA Architecture: the new concept designed for modern applications and devices



- New radical approach
- Fit with modern application data pattern
- Extremely optimizable
- Cost effective (lower Capex and Opex)

## We need to start with a little bit of theory (cont.)

### Symmetric Data Access vs NUFA Architecture



- Non uniform Data Access Latency
- Lowest Latency optimization
- Proximity and multiple collaborative caching possible
- NVMe and 3D XPoint can be used as memory extension
- (e.g. SSDAlloc locally with no latency penalty )

**Hybrid implementations are possible for easy capacity scaling**

## The Importance of the Latency vs Bandwidth

### What Are bandwidth and Latency and where is the Problem

- Bandwidth is easier to understand than Latency
- Cold Storage relies on Bandwidth
- Hot storage relies on latency
- 100 Gbit/s seem faster than 1 us network

#### Example of High Bandwidth

Move 100s of people in a single shot

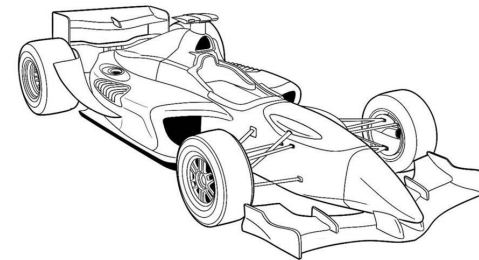
Efficient on long distance



#### Example of Low Latency

Move 1 person extremely fast

Design to win in short distance



Modern application are latency driven (e.g. Analytics, Fast Databases, Machine Learning ...) not bandwidth driven



## The Importance of the Latency vs Bandwidth (cont.)

### What we can said about latency:

- Latency is the most critical performance factor because it directly affects system data exchange time.
- Latency means **lost time**; time that could have been spent more productively producing computational results, but it is instead spent waiting for I/O resources to become available.
- Latency is the "**application stealth tax**", silently extending the elapsed times of individual computational tasks and processes, which then take longer to execute.
- Once all such delays are accounted for, the overall system performance can drop significantly.
- System latency performance matters – and not just at the storage device level, but across the system, through system, inter nodes fabric, and software applications.





## The Importance of the Latency vs Bandwidth (cont.)

Hardware Latency vs Real-Latency

To evaluate the application performance we need to pay attention to the **REAL-LATENCY**  
(Hardware+ Drivers, Kernel, APIs ...)

This is the latency that really impacts on the application

**Example with Ethernet:**

Eth **Hardware Latency**: 100-150 ns, MAC average modern NICs

Eth **Real-Latency**: 10 us Standard TCP/IP (Zero Byte)

## The Importance of the Latency vs Bandwidth (cont.)

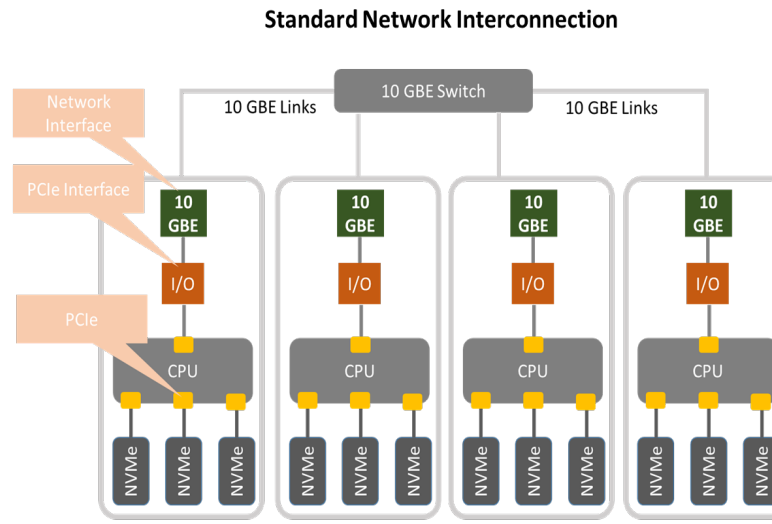
Why with NVMe latency is important and with 3D XPoint will be critical

### Consideration about Real-Latency

#### Inter Storage Latency!

Intel 82598EB 10 Gigabit AT CX4

Message Size	One Way Latency 10GigE
1	20.0
4	20.0
8	19.9
12	20.0
16	20.0
24	20.0
32	20.0
48	20.0
64	20.1
96	51.3
128	51.3
256	51.2
512	51.2
1024	49.9
2048	62.6
4096	125.0
8192	124.9
16384	125.0



@ Each hop (e.g. network switches) the situation become worst and worst

#### Typical Scale out modern Approach

Software Defined Storage most common mistake

10 us < NVMe < 100us

## The Importance of the Latency vs Bandwidth (cont.)

Why with NVMe latency is important and with 3D XPoint will be critical

Consideration about Real-Latency

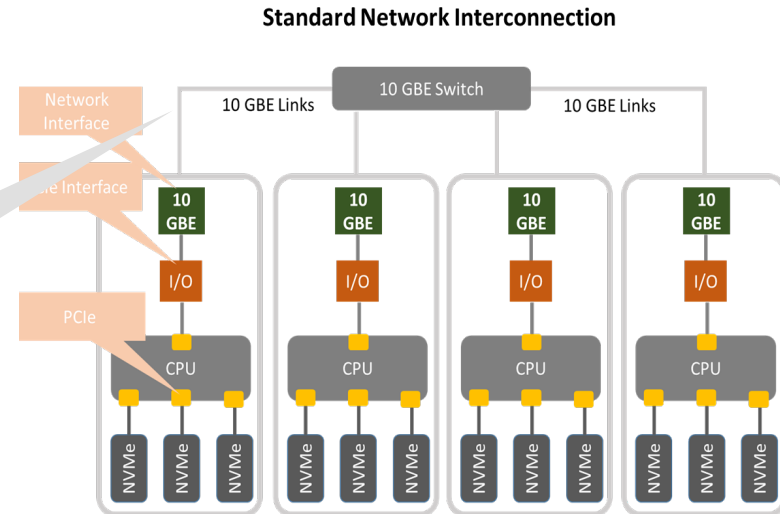
Using RDMA the problem can be solved  
(e.g. NVMe over fabric)

RDMA Mellanox 3.44 us

But ... What About 3D XPoint?

10 us < NVMe < 100us

Typical Scale out modern Approach  
Software Defined Storage most common mistake



## 3D XPoint in a nut shell



**1000X**

faster THAN NAND



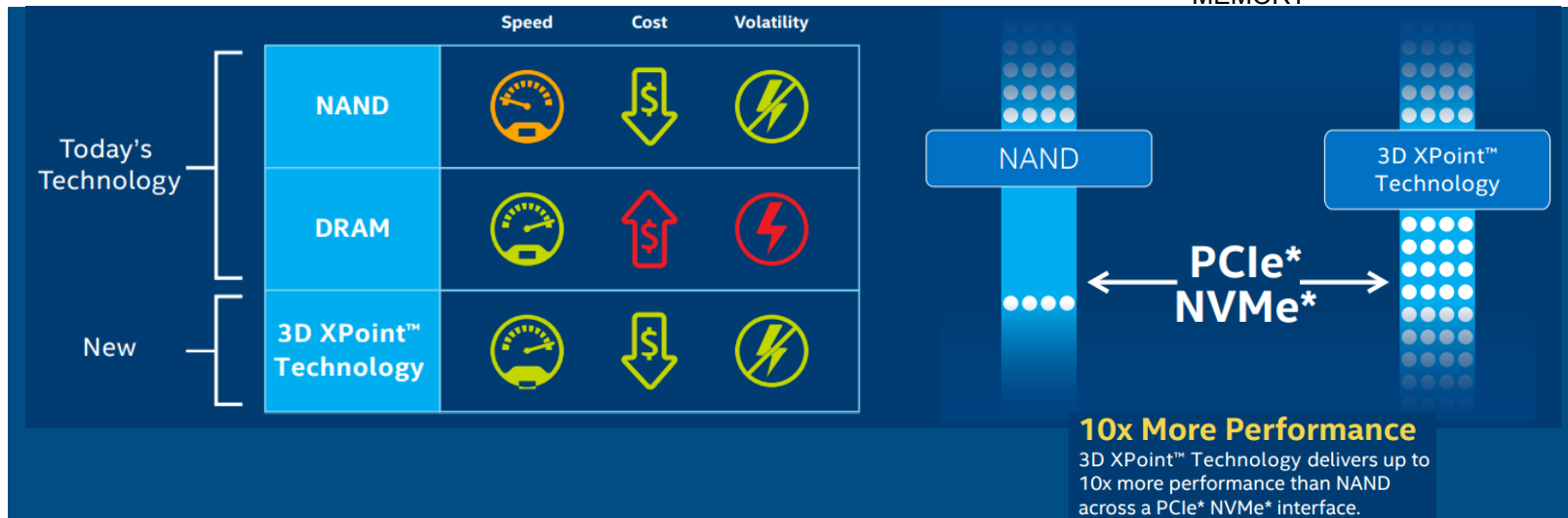
**1000X**

faster THAN NAND



**1000X**

denser THAN CONVENTIONAL  
MEMORY





## The Importance of the Latency vs Bandwidth (cont.)

Why with NVMe latency is important and with 3D XPoint will be critical

**Some key points that we need to consider:**

- **3D XPoint average latency R/W 100-500 ns**
- **3D XPoint byte addressable**
- **3D XPoint can act as memory device**
- **Can be used by application as memory devices**
- **It is not just a faster storage device**

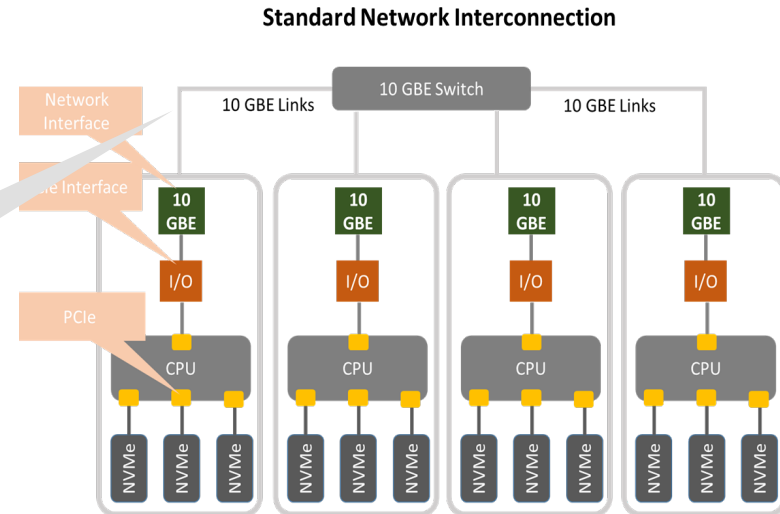
## The Importance of the Latency vs Bandwidth (cont.)

Why with NVMe latency is important and with 3D XPoint will be critical

With 3D XPoint this latency is not any more enough ...

RDMA Mellanox 3.44 us

Remember we are talking about Real-Latency not Hardware Latency



Typical Scale out modern Approach

Software Defined Storage most common mistake

100 ns < 3D XPoint < 500ns



## 3D XPoint Scale-Out Challenges

- **Maintaining Ultra Low Latency between data across local and remote devices**
- **Use proximity to take maximum advantages form Byte addressability (System Memory Extension)**
- **Combining sophisticated technology in a new way:**
  - Direct Addressability (extension of mmap ...)
  - Open Channel architecture managed using distributed kernel approaches or “replicated multi-kernels approach” to achieve cluster wide optimization



## 3D XPoint Scale-Out Solutions

Efficient Memory Clustering  
3D XPoint Scale-Out  
Global memory pools

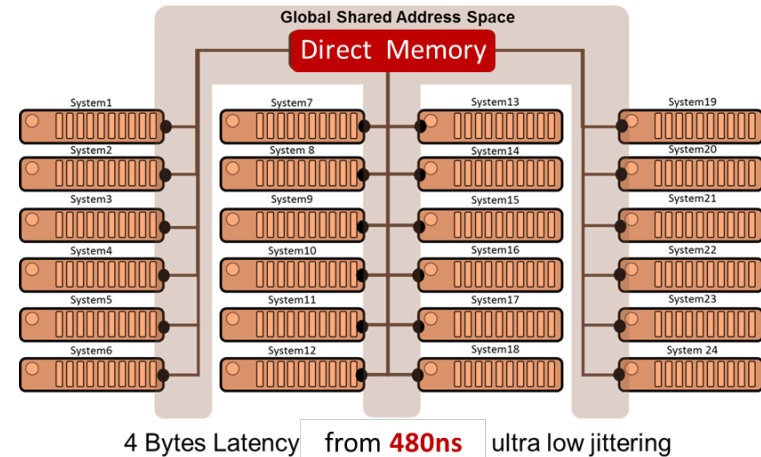
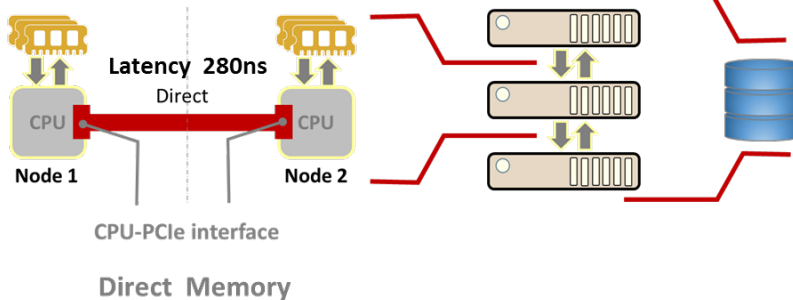
Advanced use of PCIe direct memory injection @ Cluster level

Fully working product on the market (e.g. RONNIEE Express™, Dolphin IXH)

CPU as NIC for direct remote memory injection

Example and numbers:

### The RONNIEE Express™ Direct Interconnect System



### Software Included

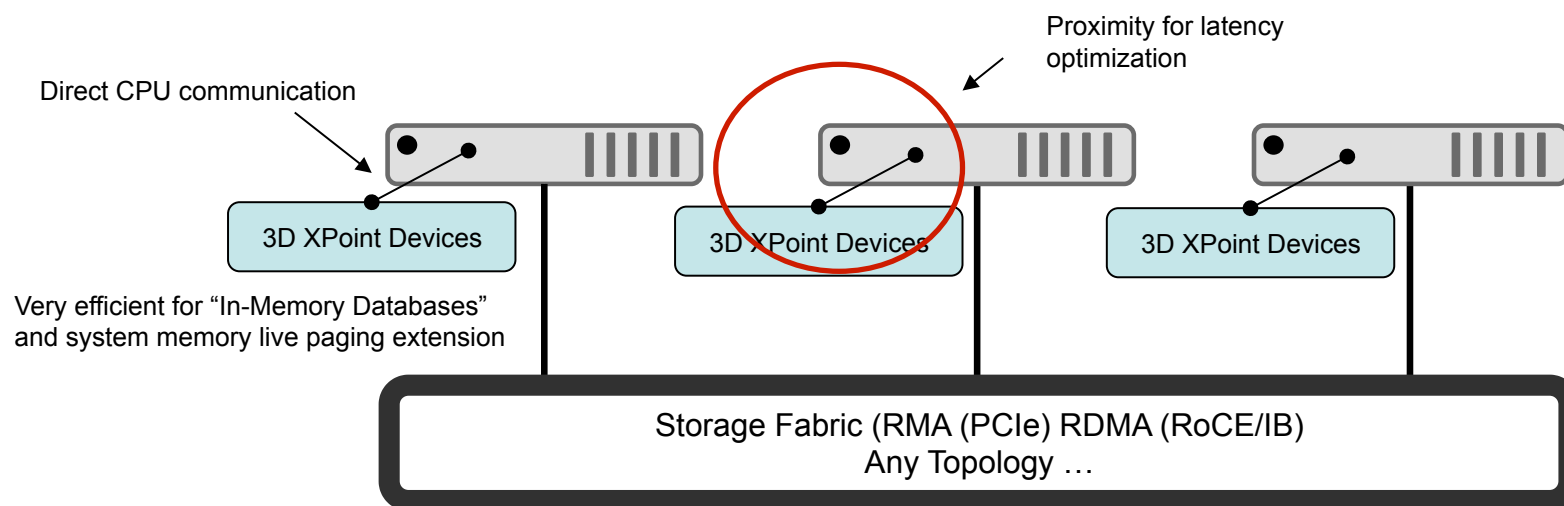
Real-Latency Application to Application

Emilio Billi - A3Cube Inc



## 3D XPoint Scale-Out Solutions

@ Architectural Level



From the concept of NUMA (Non Uniform Memory Access) to the concept of NUFA (Non Uniform File system Access)



## NVMe and 3D XPoint Advanced Features

Other advanced features built in the NVMe interface that enable new kind of efficient architectures @ datacenter level

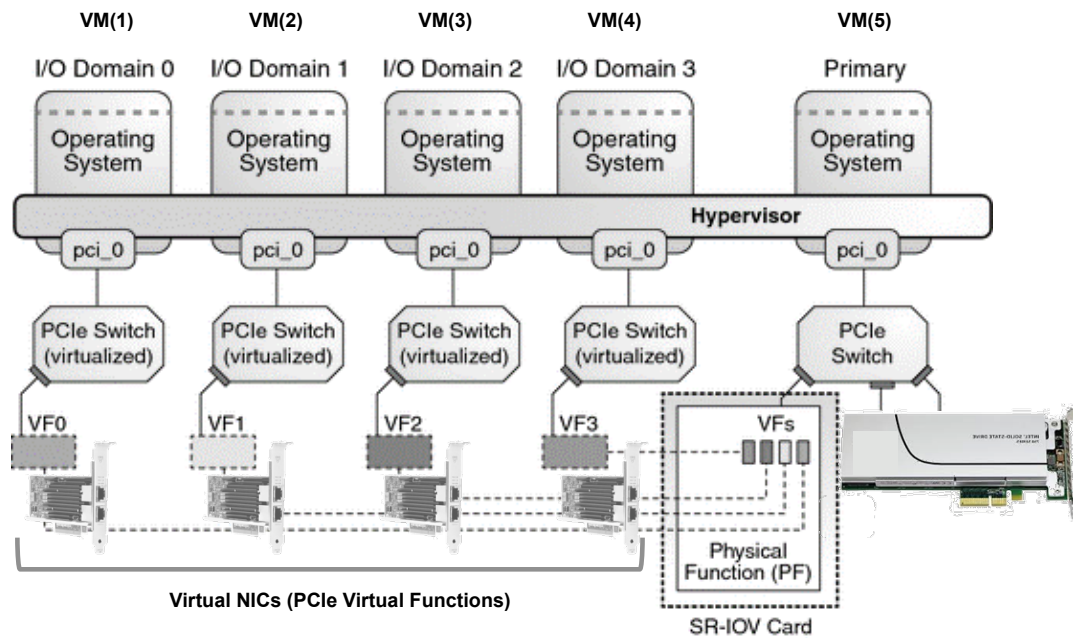
NVMe and 3D XPoint in combination with PCIe cluster wide global memory mapping or over RDMA features permit to open new datanceter scenarios.

### **The Cluster SR-IOV** (Single Route IO Virtualization)

# Introducing Cluster wide SR-IOV

## How today PCIe SR-IOV is used

Well-Know SR-IOV Implementation



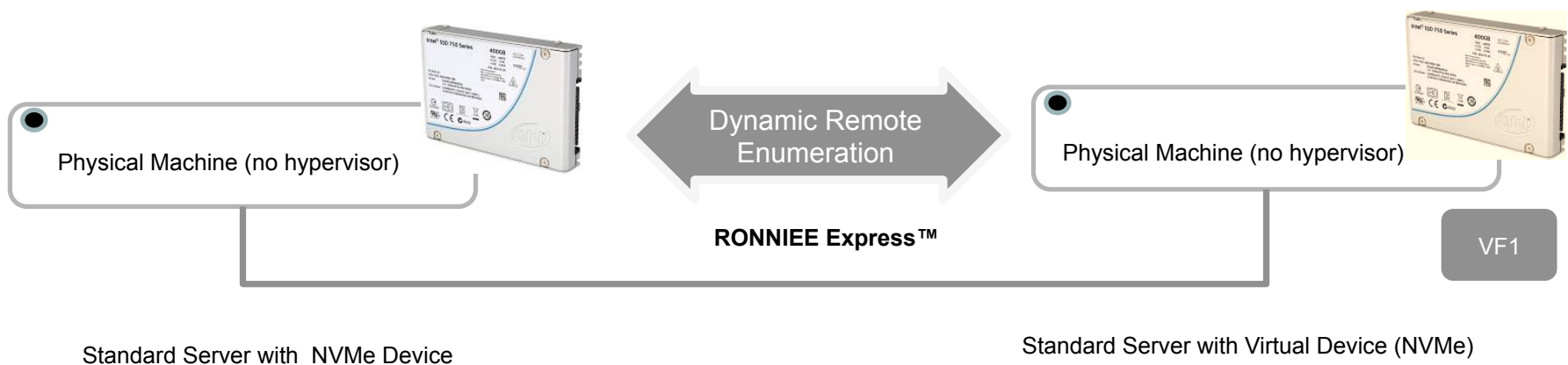
**A) Single Server → Multiple Virtual Machines**

**B) Hypervisor**

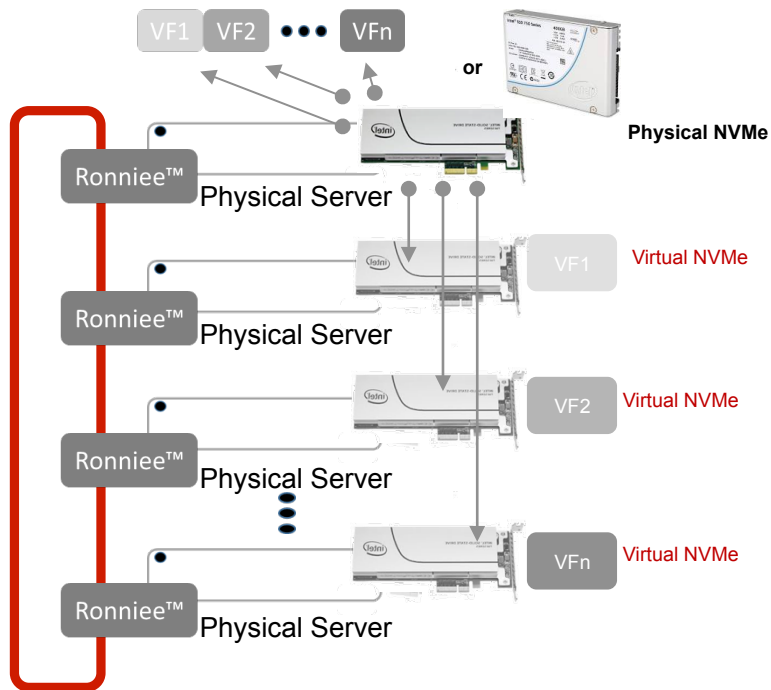
**C) SR-IOV Card in the main server (Host)**

## Introducing Cluster wide SR-IOV (cont.)

Distributed PCIe discovery: operating system and virtual machines enable remote discovery, addressing, access and use of standard PCIe devices.



## Introducing cluster wide SR-IOV (cont.)



- 1) Operating System Transparent
- 2) Virtual Device uses original OS driver (no modification)
- 3) NVMe(s) are seen as local by all the servers  
**( Un-Supervised Sharing: No software control involved, native direct disks access)**

Virtual NICs (PCIe Virtual Functions)

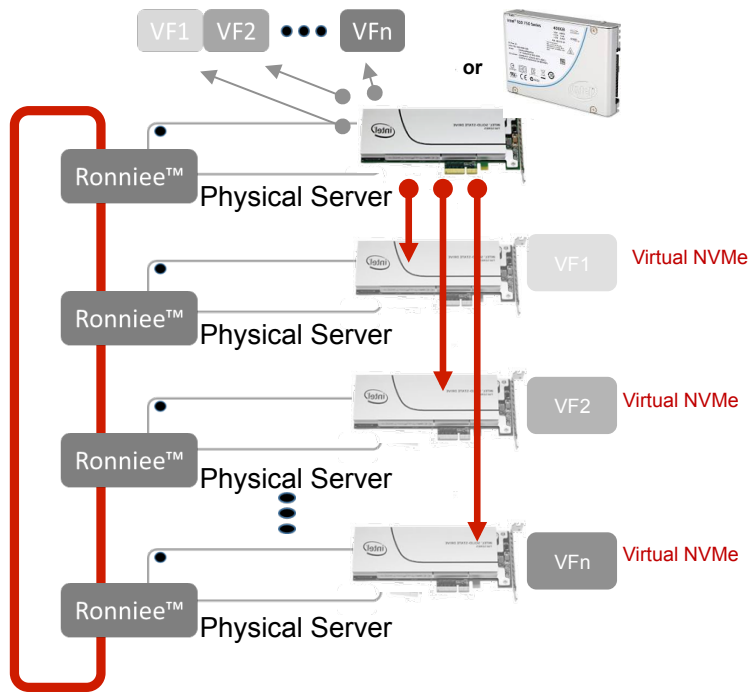
Utilization made easy using unmodified shared disk file system

**OCFS2 Oracle**  
**GFS Redhat**  
**GPFS IBM**

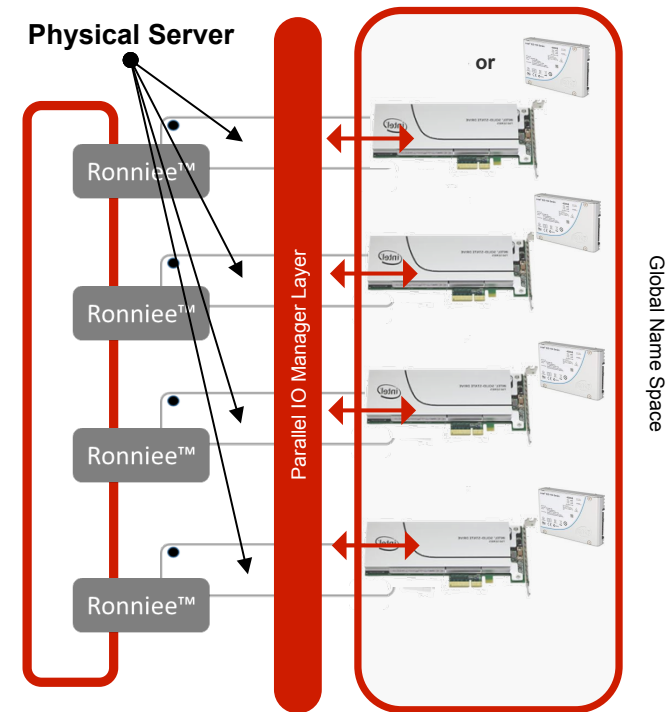
....  
 Or using multiple partitions inside the disks

## Introducing Cluster wide SR-IOV (cont.)

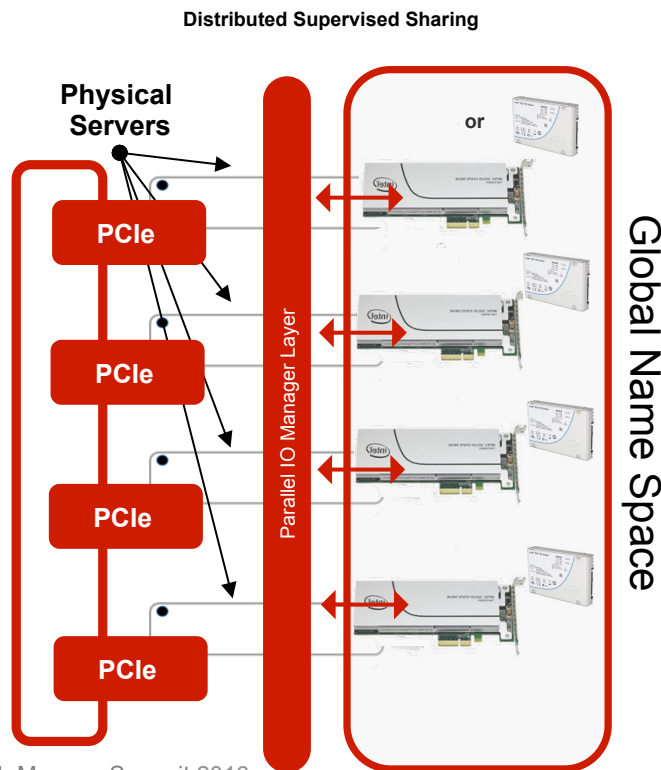
### From Unsupervised Sharing



### To Distributed Supervised Sharing



## The Key Points of the Parallel IO Manager Layer



A new way to think about NVMe sharing:

- 1) Symmetrically sharing devices across different nodes
- 2) Managing cooperative caching and memory proximity
- 3) Access distributed data in fully parallel way from any nodes and from any application



# How architecture impacts on application performance





## A Benchmark

**Application:** Hadoop

**Comparison between:**

**A)** Standard implementation with NVMe as storage and Infiniband FDR (IPoIB) 56 Gbit/s

**B)** NUFA Architecture (\*)

(Same hardware of point A with : Distributed Data Management, Pervasive SR-IOV, direct NVMe data access (with proximity) )

Simple test : 40 GB Teragen

(\*) Fortissimo Foundation is a commercial product the first introduce all the technology described before (including direct memory injection) on standard server.



## A Benchmark

Symmetric Storage  
architecture on NVMe

NUFA Storage  
architecture on NVMe

**A**

**B**

**Execution Time**

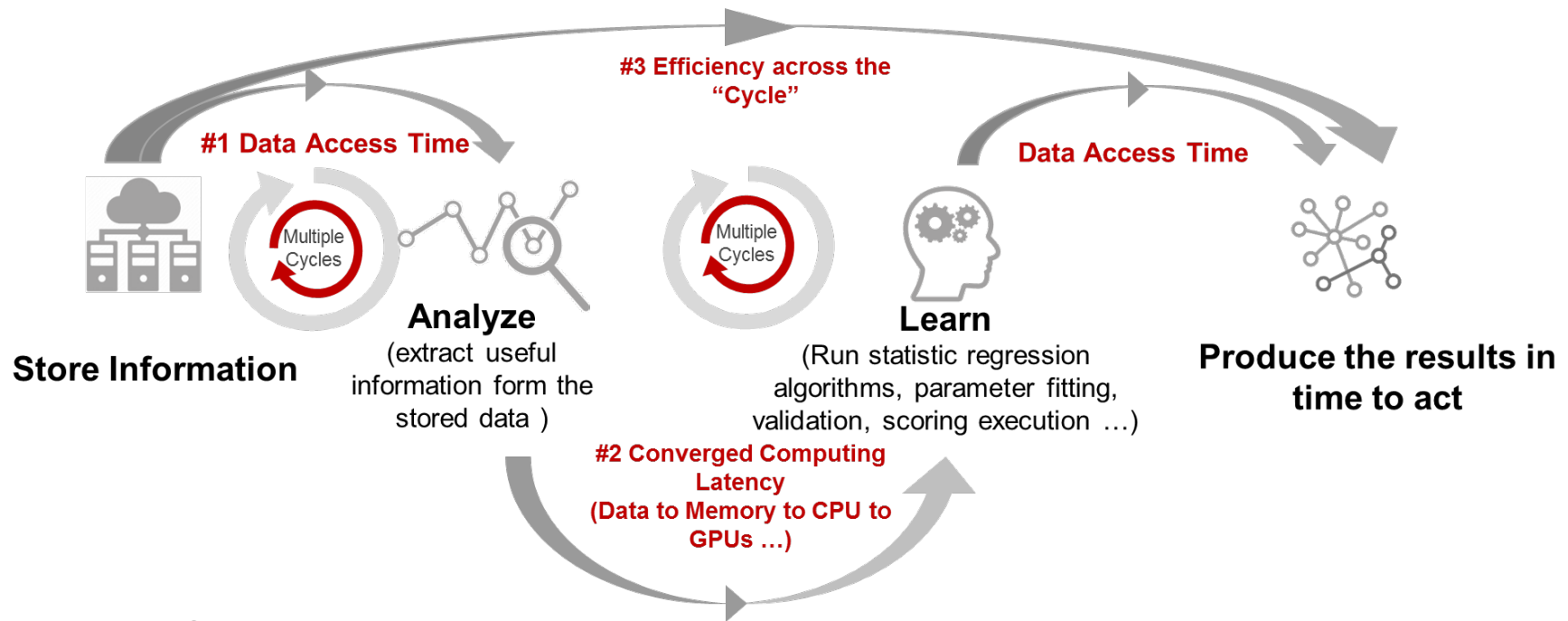
**11.53 min**

**51 sec**

Same hardware, same storage, same NVMe  
different architecture organization

(Hardware configuration available on request)

# The “Data/Computing Cycle” for the foreseeable future





## Machine Learning Training Data Challenges

- **Massive Datasets**
- **Massively Parallel Computation and Data Operation**
- **1000s of Iterations**
- **Intense use of Map Reduce**
- **Hybrid computing (possibility for direct accelerators storage access)**

## Machine Learning Training Data Challenges

In a Nutshell:



**All Computation is the same (\*)**  
**(Results = Math + Data)**



**To get results faster, two things need to happen:**

- **Faster Data (Reducing data access latency)**
- **Faster Math (Reducing computation latency)**

(\*) David A. Patterson, Daniel W. Hillis, Seymour Cray & Others...

## Coming back to the architecture

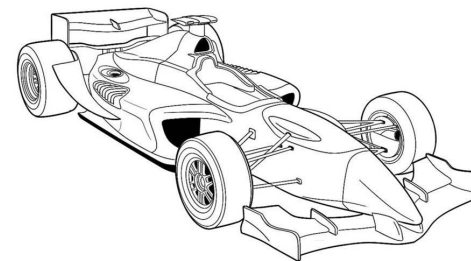
### High Bandwidth



It is useful if you need to carry large amount of data from A to B

**Completely unusual in massive parallel scenarios with multiple time critical data path if there is not also low latency combined.**

### Low Latency



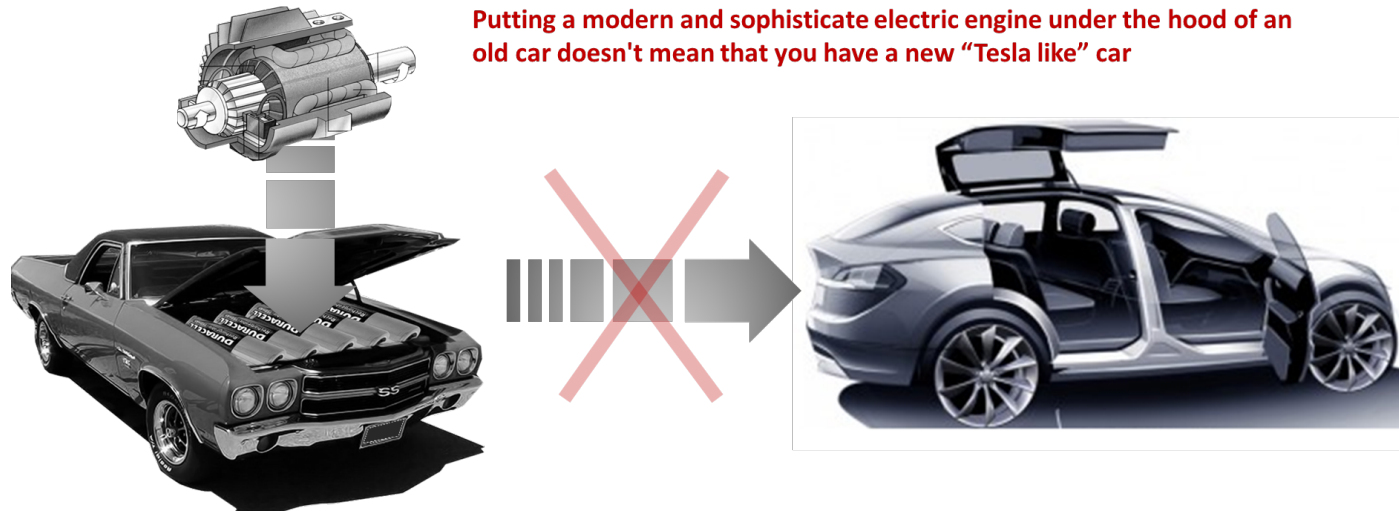
Analytic and ML require to exchange data between 1000s of computational and storage devices cores distributed between different nodes, using different path (not just A to B)

**NVMe , 3D XPoint are able perfectly to support that but only considering the total architecture**

## Coming back to the architecture (cont.)

### The Storage Paradox

Why working at architectural level instead at software level only



By putting new technologies (SSDs, PCIe SSDs, ... adding complex software and ...) under the hood of an old storage scale OUT architecture you doesn't create a new system and you doesn't achieve a real high level of performance.

**Faster are the devices better design is require in the overall system architecture**



Thank you  
Questions?