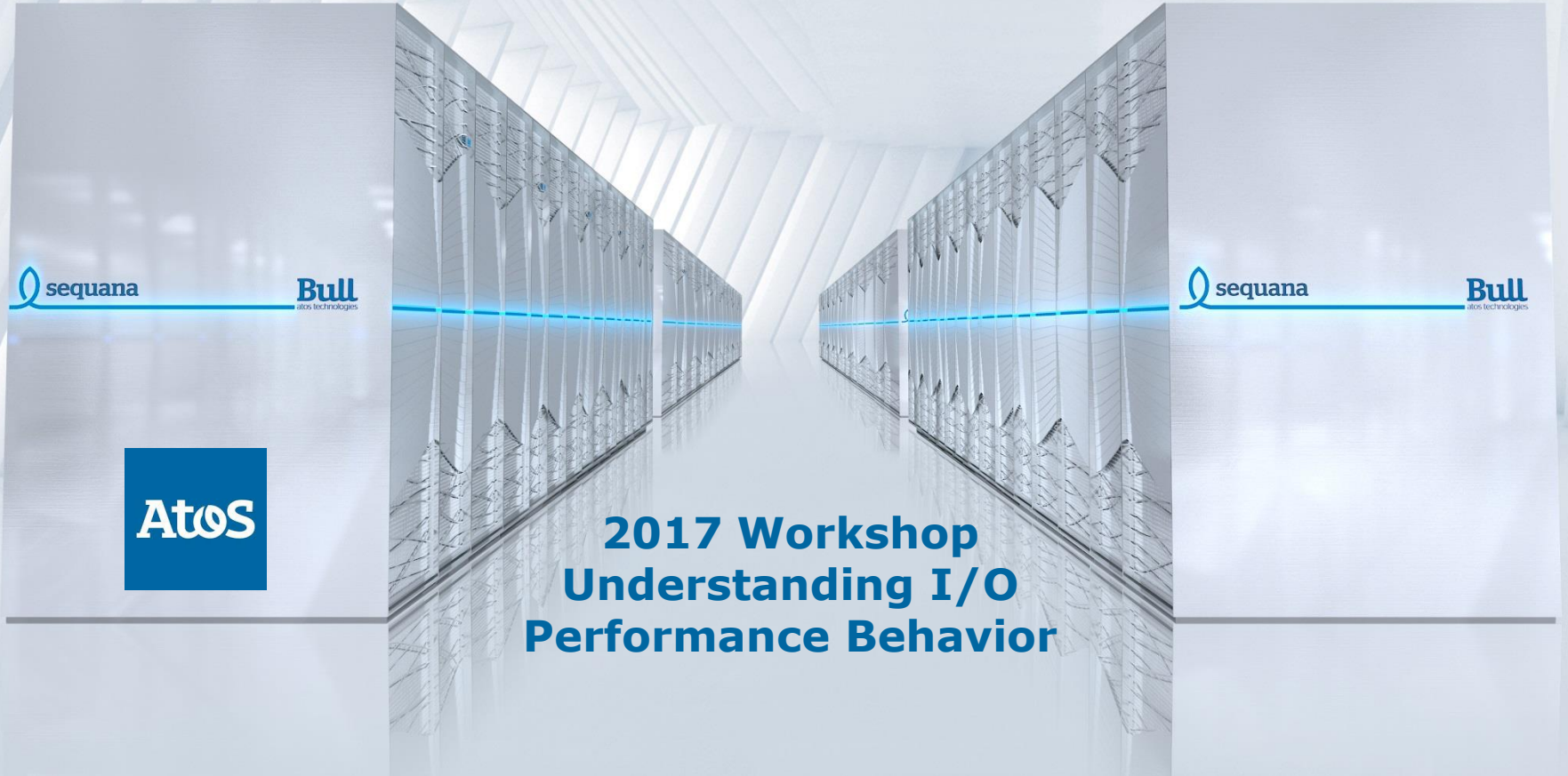
Atos Perspective & Studies on storage architecture and benchmarking



Atos

2017 Workshop
Understanding I/O Performance Behavior

Agenda

- Our perspective for Data Management & Storage Architecture
- I/O Accelerator solutions
- Benchmarking I/O solutions

Our Perspective for Data Management & Storage Architecture

Innovate to make the difference

Next gen. Smart Memory Hierarchy Management Platform

Fast IO Libraries

Aggregate small IO in large ones

Smart Burst Buffer

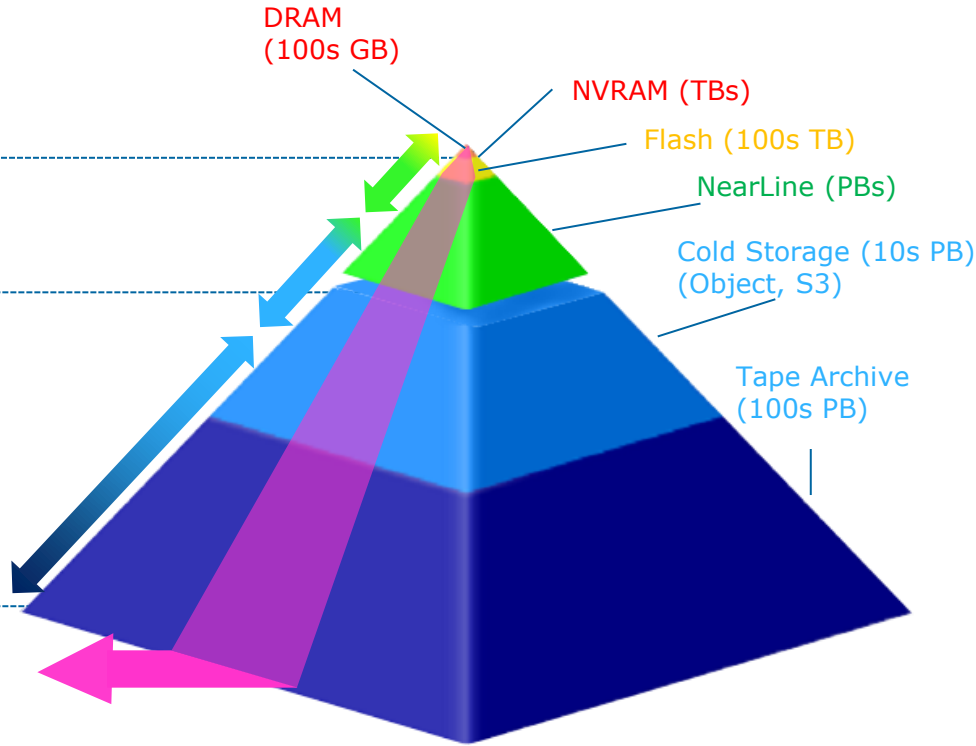
Let apps. compute faster by doing IO on fast disks, then forward IO on // FS

Next Gen Hierarchical Storage Manager (Bull Dino & Raptor)

Move data transparently or on demand between tiers. Learn from movement and compute patterns and move data preemptively

IO Instrumentation & IO Optimizer

Trap system wide IO patterns & feed a machine learning base




2
I/O accelerator Solutions

Storage and IO activities

- Understand application behaviors on large systems, determine IO patterns, predict optimal behaviors and take actions
- Rationalize data movements across the memory hierarchy layers (from DRAM to tape) but also across the system to optimize job placements as close to data as possible
- Involve machine learning mechanisms to learn from applications, bad or good behaviors and anticipate IO as much as possible and optimize parameters

- Understand I/O accelerator benefit for applications
  - NVMe/NVMf in-house solution
  - Seagate NytroXD: Seagate flash based accelerator feature for small block
  - DDN IME (Infinite Memory Engine) also known as Burst Buffer technology

Lead by CEI, the exploration Group

HPC Competency Center (HPC CC)

- Identify & study emerging technologies
- Understand benefits for your applications & production
- Drive you to adopt suitable solutions
- Support our customers to optimize new solutions implementation & usage

30 experts

Current Activities

- IO accelerator Technologies
  - NVMe - NVMf up to tight integration with Slurm Batch Scheduler
  - Seagate NytroXD, Small Block Accelerator
  - DDN IME, Burst Buffer

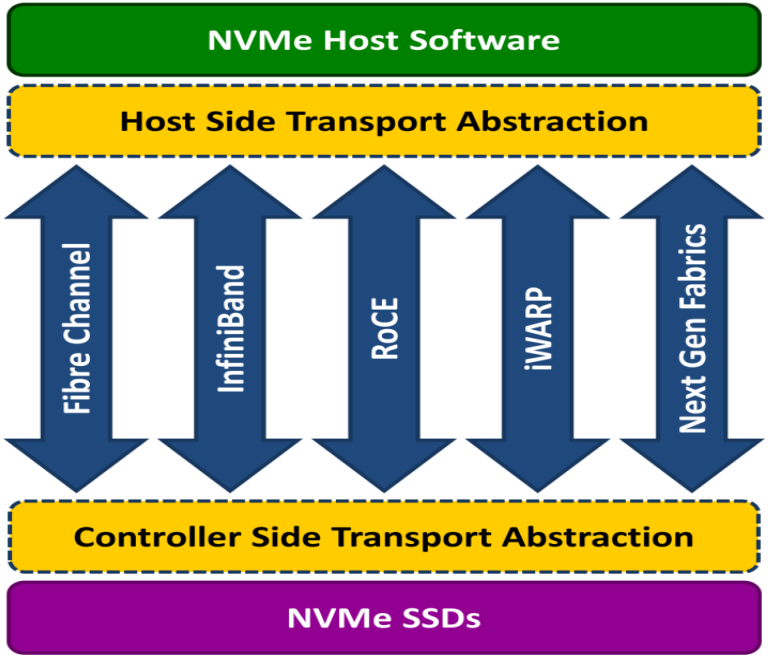
- Applicative Checkpoint/Restart, FTI

- HDEEVIZ “High Definition Energy Efficiency Visualization”
  - Job energetic profiling Visualisation
  - Power consumption profiler

NVMe over Fabrics (NVMf)

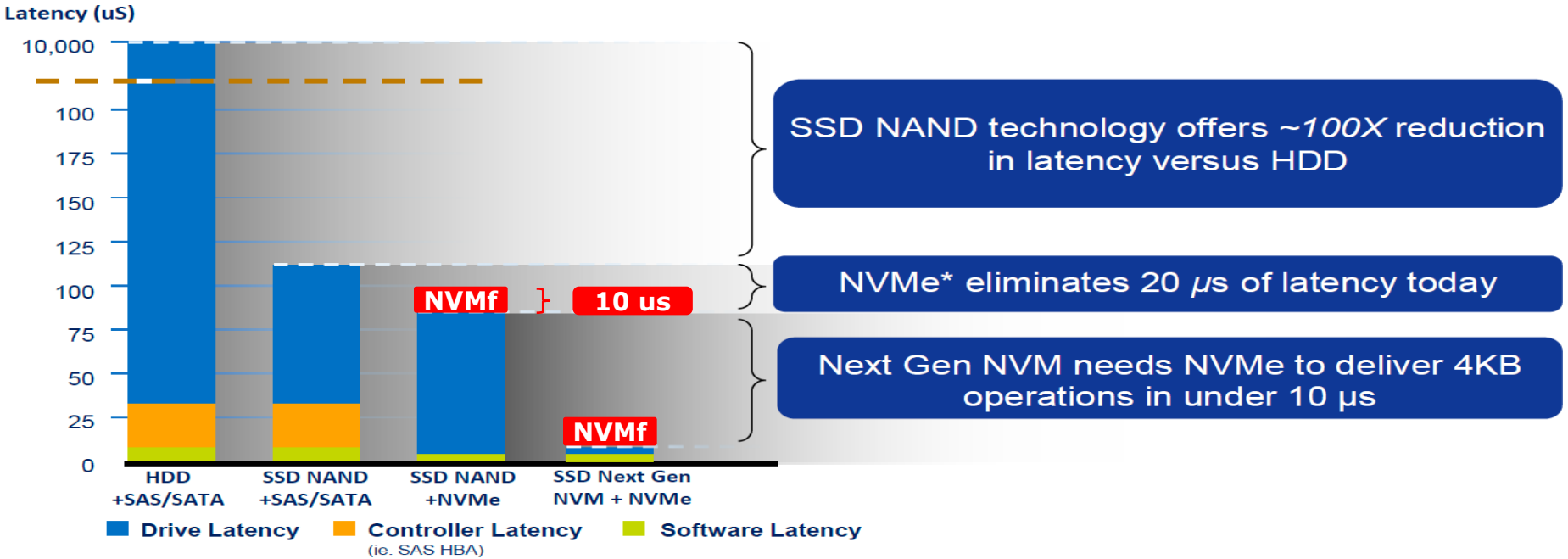
What is NVMf?

- NVMe block protocol tunneled through an RDMA fabric
  - Flash-optimized fabric-attached storage
- Extension of NVMe
  - NVMe was introduced in 2011
  - NVMf has been published in June 2016
- NVMf targets
  - distance connectivity to NVMe devices with no more than 10 µs of additional latency (vs ~100us for iSCSI)
  - increase of the number of devices being accessed by the host (typically 1000+)
  - reduce CPU usage from TCP/IP processing



NVMe over Fabrics (NVMf)

What to expect?



Source: Intel

NVMe over Fabrics (NVMf)

Use Cases

- IO acceleration for IO-bound applications traditional HPC code, databases
  - Proxy Burst buffeR
  - Dynamically-allocated fast temporary storage
  - traditional HPC code, databases
  - Fast deployments of Big Data workloads on HPC systems
- Real-time analytics
- Checkpoint/Restart

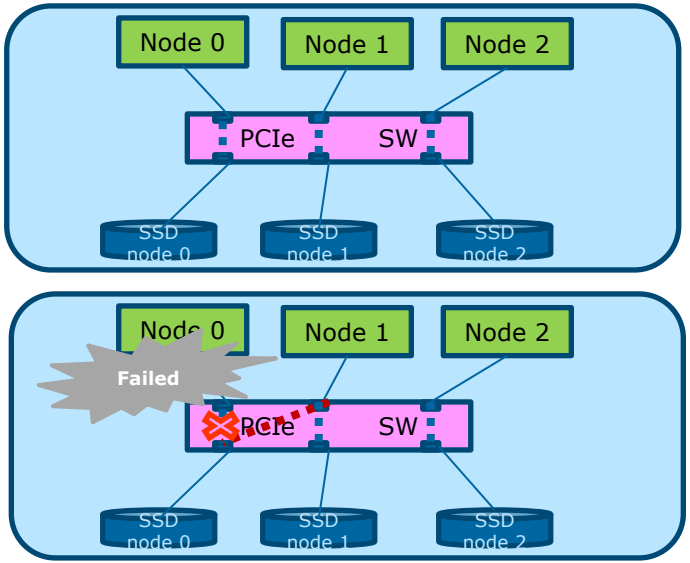
Bull sequana X1000

SKL & KNL blade w/PLX and NVMe SSD

- 1 bi-sockets nodes blade
- 3 compute node + 3 NVMe disks + PLX PEX8733 PCIe Switch

- Bull PCIe Switch Management Library – libbpsm
  - Library based on libpciaccess
  - API

- Fault Tolerance Interface
  - Multi-level checkpoint/restart library
  - http://leobago.github.io/fti/
  - Work in progress

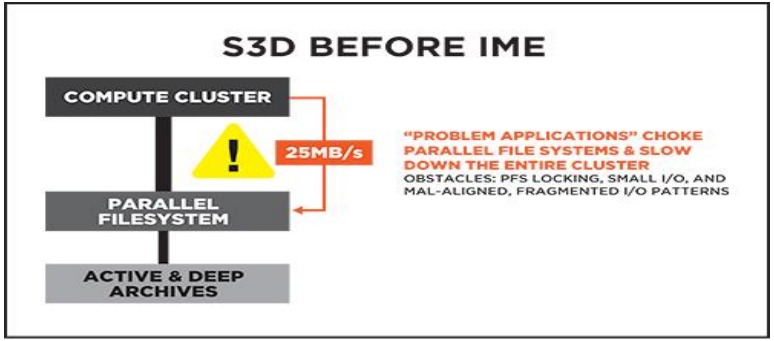


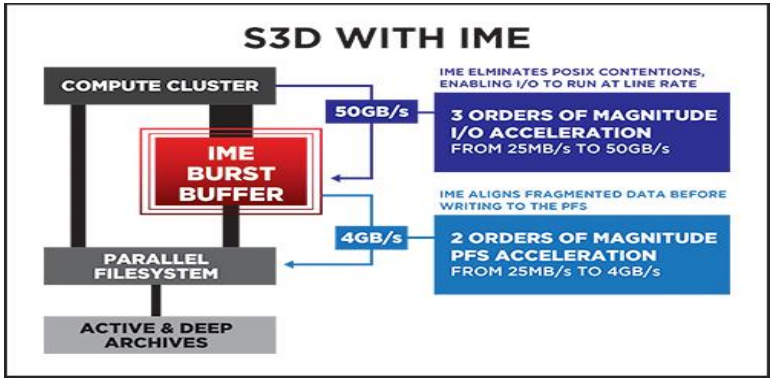
DDN Infinite Memory Engine

Work in progress

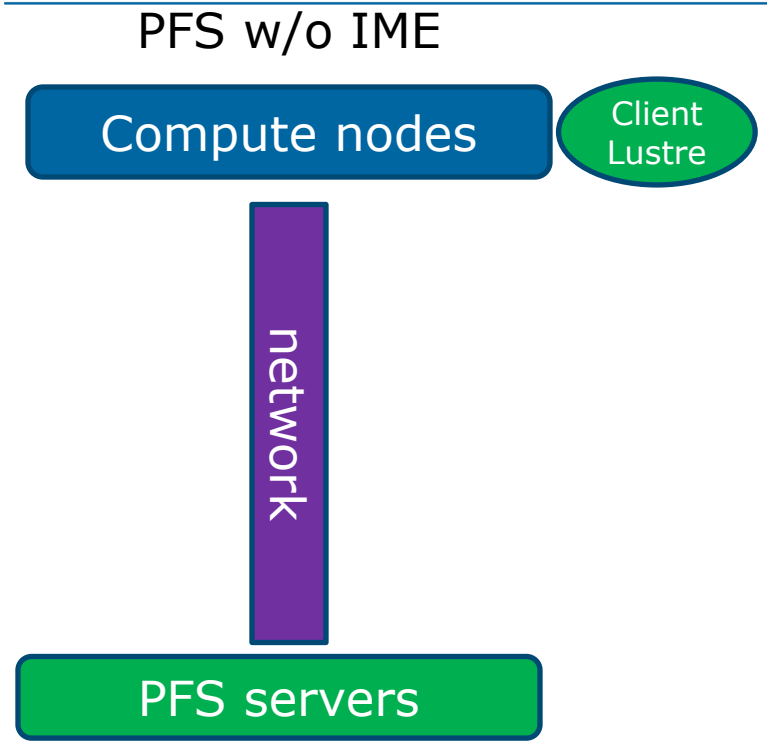
Atos

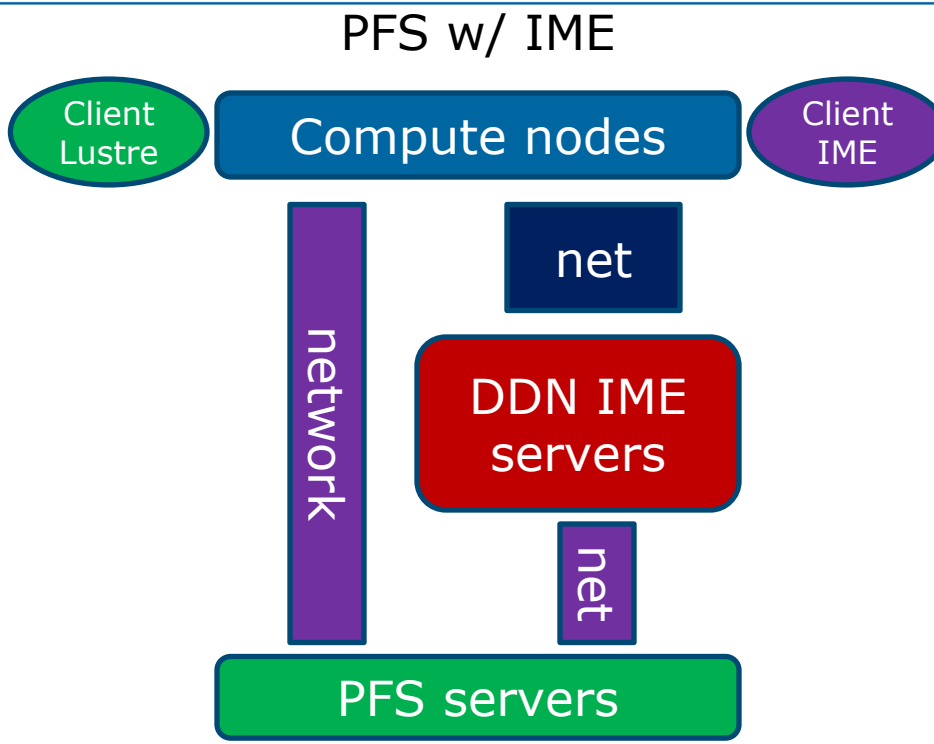
- All flash arrays
- Fits seamlessly between a cluster and the parallel file system (Lustre or IBM Spectrum Scale)
- Several benefits
  - Write cache to absorb write bursts
  - Filesystem acceleration by realigning writes
  - An API to optimize the reads (stage-in, stage-out)



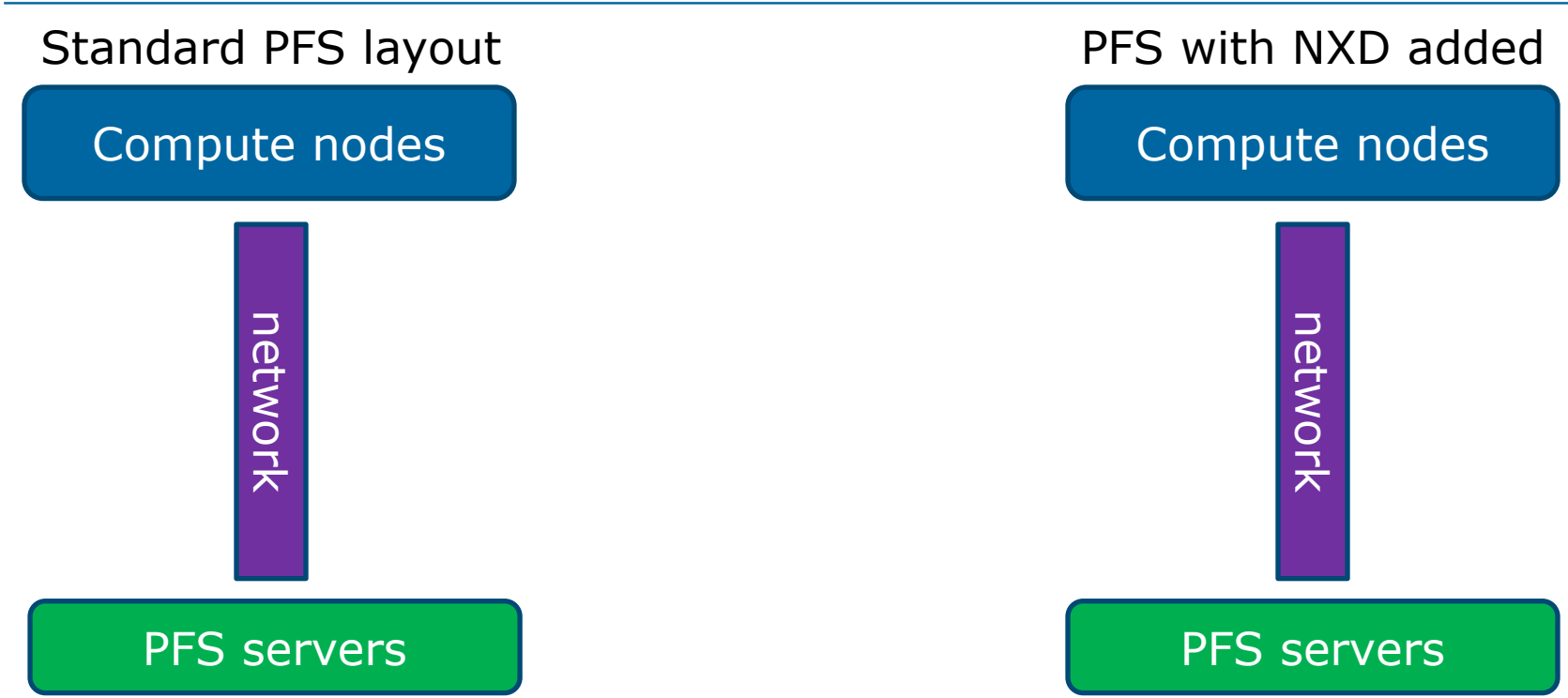


Architecture w/o & w/ IME

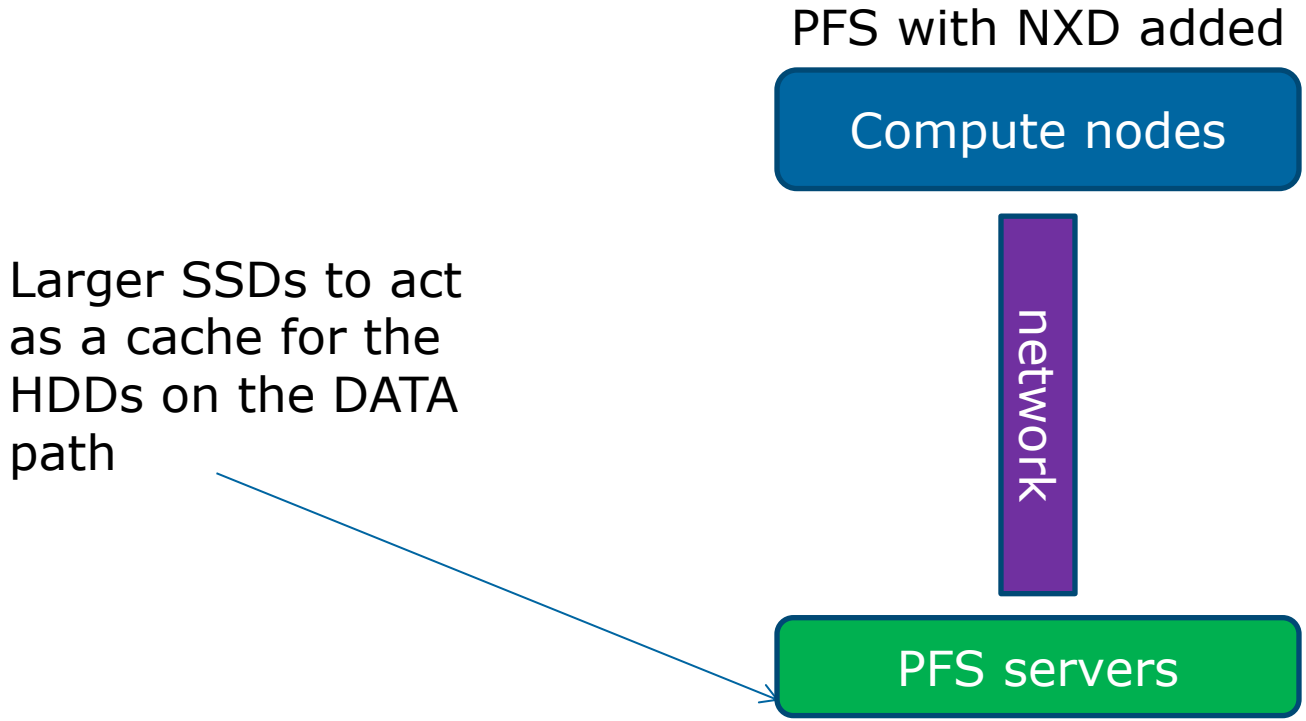




Seagate NytroXD



Seagate NytroXD




3
Benchmarking I/O solutions

CEI Exploration Group : our Drivers

Your production

Performance

- System level
- w/& w/o I/O accelerator solution
  - User level Synthetic benchmarks (IOR, fio,…)
  - Applications IO kernels
  - Real applications

Resiliency

- Integration
- Administration
- RAS
- Support

Energy efficiency

- Performance/Watt

A&P (benchmarks team) drivers

- A&P : "*Applications and Performance*"
- Business oriented : RFPs, Tenders,… with hard deadlines
- Deliverable : benchmark (BM) report, *and output files*
- BM Execution rules:
  - sometimes open/fuzzy (eg. "10 GB/s IO bandwidth")
  - often (almost always today) strict and accurate (cmd line +flags requested)
- Synthetic IO BM, metrics requested by most RFPs:
  - **sequential** large block bandwidth : ~always since more than 15 years
  - **metadata** : more and more, but 10 years ago was really a small activity
  - **random** small block :
    - iops : sometimes ;
    - latency : never seen yet

A&P : benchmarking at several levels

- Measurement vs Extrapolation
  → Commitment

- Win the deal
- CEPP team / Fast start

- Install the system
- Acceptance : **run the BM to demonstrate commitments**

- Production, support, …

A&P : extrapolation

- Single value *sometimes easy*:
  - **sequential** bandwidth : find the bottleneck in architecture
  - **metadata** :
    many metrics;
    what relations ship between MDT iops and empty file creation rate ??
  - **random** small block
    - iops
    - latency

- Extrapolate a whole graph …

Real life application

Understand how your applications behave

- Communication, memory bound, CPU, IO
- Profiling tools
  - Darshan, MAP, …
  - Inspectio (home made tool)
    - library : *inspectio.so (LD_PRELOAD)*
    - wrapper for meaningfull IO libc calls : *open,close,read,write*
    - « summary file » by process at job end
    - amount of bytes read/written from/to each file
    - amount of time spent in each file
    - Try to answer the most common questions
      - time spent on IO ?
      - How many process access each file ?

COSMO@DKRZ

Fast IO Libraries
Aggregate small IO in large ones

- **Challenge**
  - run Cosmo in 329s (contractual)
  - Impossible even after compute code optimization
  - Only trigger is to use more cores to decrease run time
  - Code run in 350s
  - Core count 890 cores to get to 329s

- **Solution**
  - Run fast IO Library (beta)
  - Code run in 329s as contractually expected
  - Core count = 725 cores

Boost application & system efficiency

Net gain

165 cores = 18%

Our ressources
Lab & Production clusters

- **Atos HPC clusters location**
  - Lab @Echirolles
    - OPA, FDR, EDR,
    - ~45 compute nodes
  - Production @Angers
    - Large compute cluster (up to 540 nodes, EDR, 3:1)
    - GPFS and Lustre
  - NVMe-NVMf
  - Seagate CS L300N - Nytro XD
  - DDN IME light – SFA7K

- **Both have external remote access for customers & prospects**

Thanks

For more information please contact:
T+ 33 1 476 29 7148
denis.gutfreund@atos.net

Bull
atos technologies