



OPENFABRICS
ALLIANCE

OpenFabrics
Software
User Group
Workshop

Lustre* Filesystem for Cloud and Hadoop*

Robert Read, Intel

Lustre* for Cloud and Hadoop*



- Brief Lustre History and Overview
- Using Lustre with Hadoop
- Intel® Cloud Edition for Lustre

What is Lustre*

- High performance parallel filesystem for Linux environments
 - Designed to use RDMA on high performance fabrics
- Allows large number of users to share a file system
 - High speed and low latencies
 - Across local or wide area networks
- Designed for reliable storage
- Ideal for streaming IO and large, shared datasets
- Evolving to bring parallel filesystem to new workloads

Quick History



- Originally built to solve next gen IO problems for HPC
- An open source (GPL) project from the beginning
- Core team has survived numerous transitions and is now safely established in Intel's HPDD.
- Used in production systems since 2002
- Intel Enterprise Edition for Lustre* since 2013
- Intel Cloud Edition now available on AWS Marketplace

Lustre* Storage Components

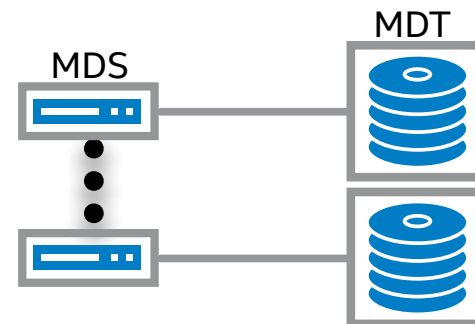
Management Target

Lustre mount service
Initial point of contact for Clients



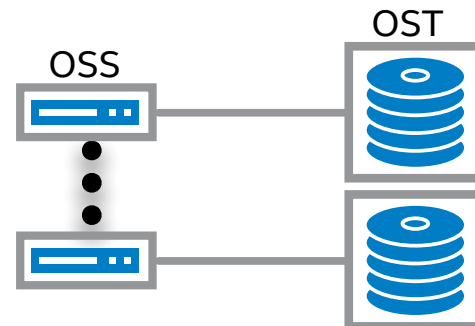
Metadata Targets

Namespace of file system
File layouts, no data
Scalable



Object Storage Targets

File content stored as objects
Striped across multiple targets
Scales to 100s

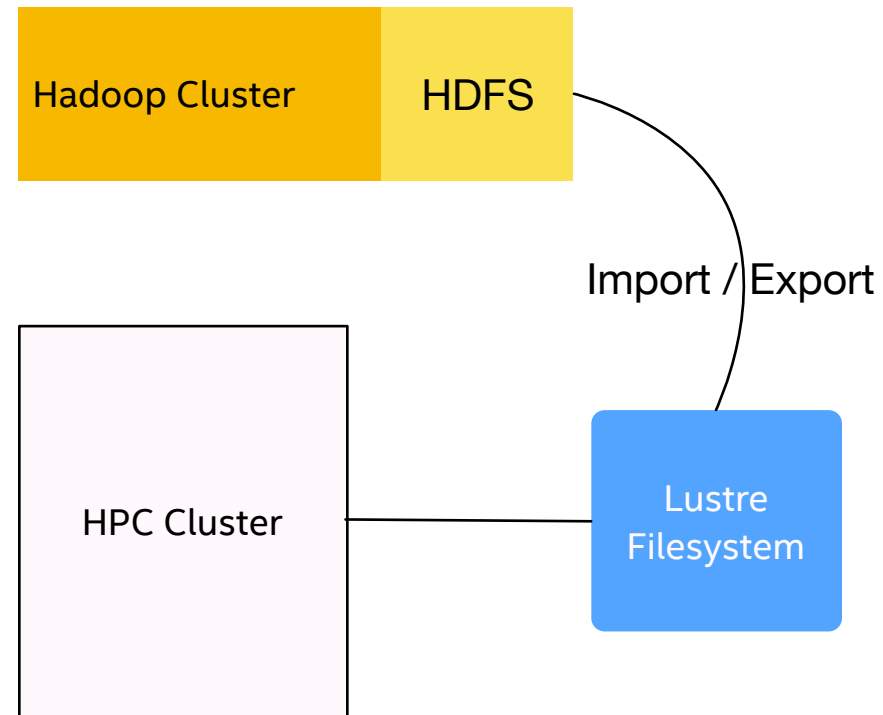


Hadoop* Introduction

- Open source framework for data-intensive computing
- Parallelism hidden by framework
 - Highly scalable: can be applied to large datasets (Big Data) and run on commodity clusters
- Comes with its own user-space distributed file system (HDFS) based on the local storage of cluster nodes

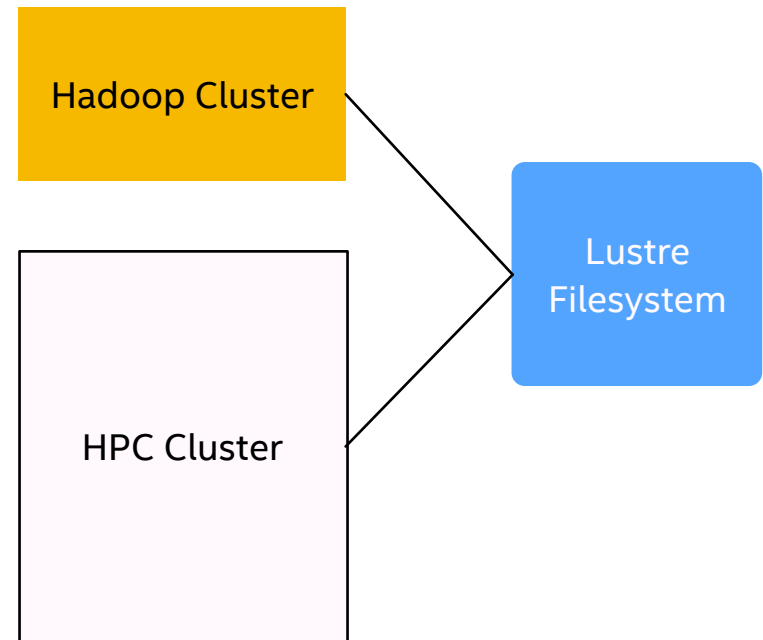
Hadoop* with HDFS

- HDFS locality has advantages, but...
 - HDFS requires import/export to share data
 - Compute nodes require local storage
 - Hadoop nodes are both compute and IO nodes
 - Hadoop nodes are single-purpose



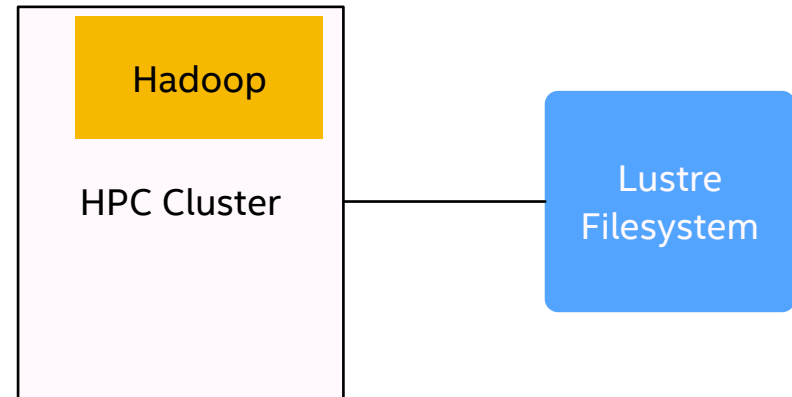
Hadoop* Adapter for Lustre*

- Shared data repository for all compute resources
- Use data in place (no import/export)
- Dedicated Compute and IO nodes



HPC Adapter for MapReduce

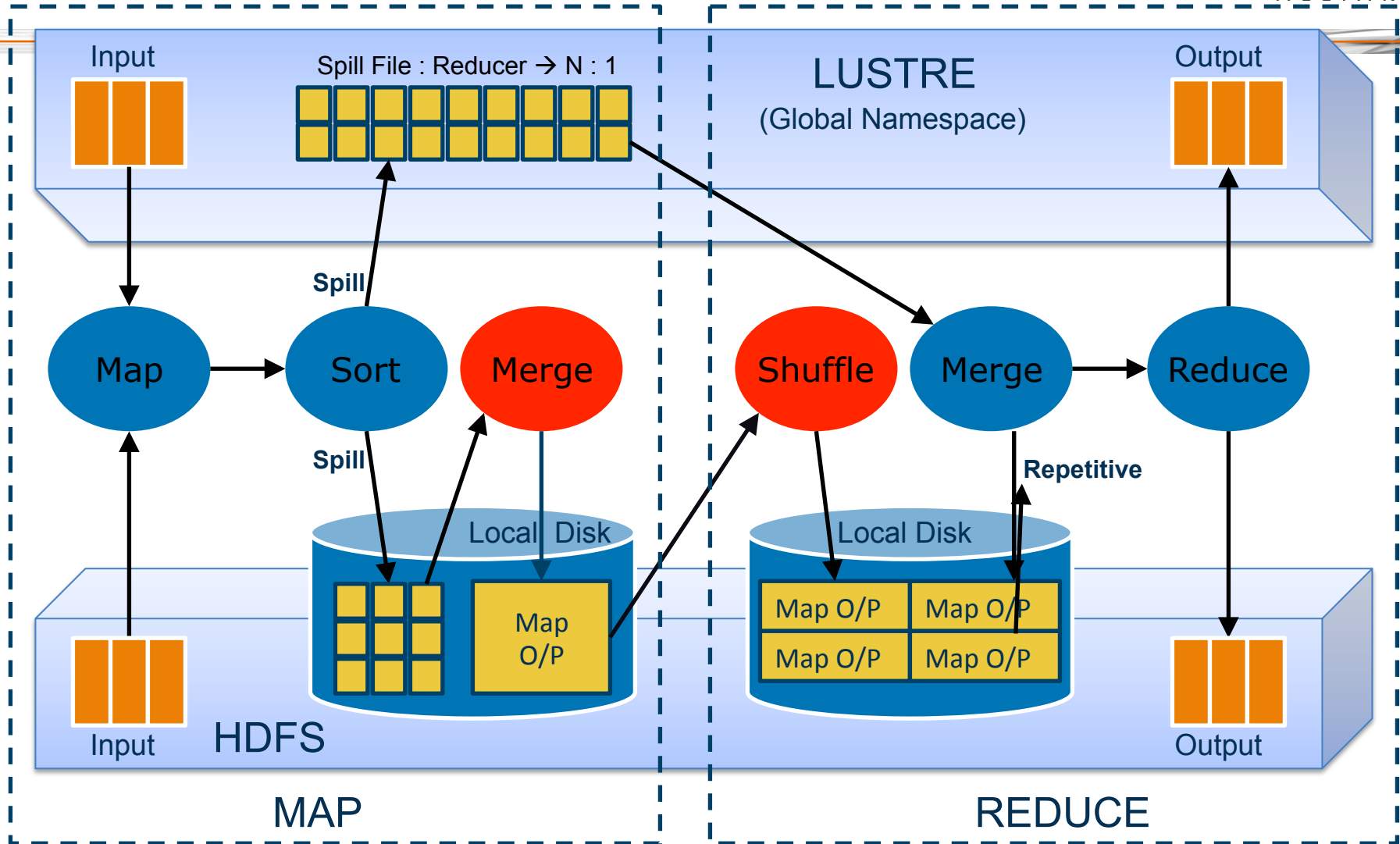
- Run Hadoop* on HPC cluster
- Replace YARN with Slurm
- Single compute cluster used for variety of workloads



Optimized Shuffle: HDFS vs Lustre*



OPENFABRICS
ALLIANCE

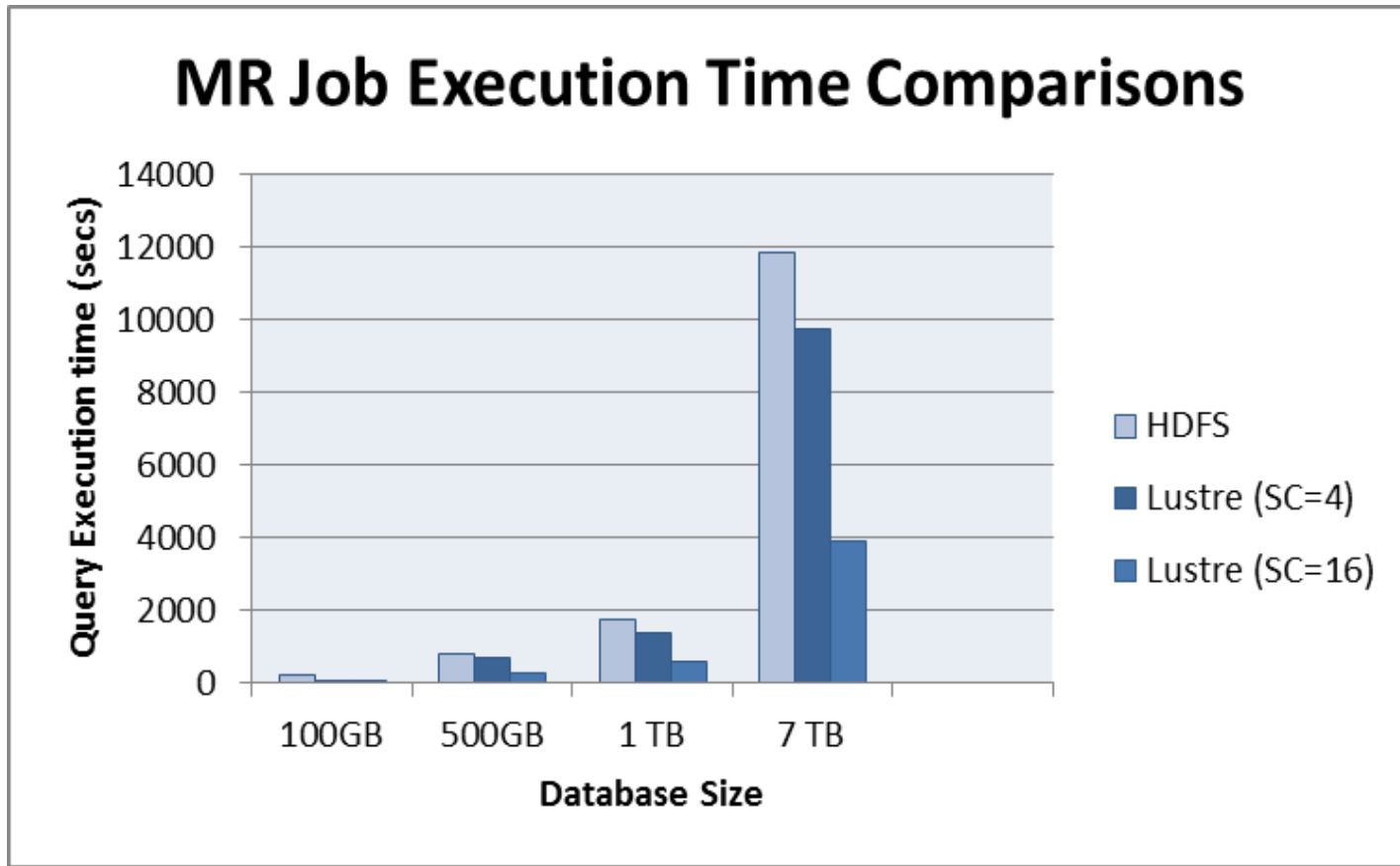


* Some names and brands may be claimed as the property of others.

Performance Comparison

- Tata Consultancy Services performed comparison in 2014
 - “Performance comparison of Lustre* and HDFS for MR implementation of FSI workload using HDDP cluster hosted in the Intel BigData Lab in Swindon (UK) and Intel® Enterprise Edition for Lustre* software”
 - Intel® EE for Lustre = 3 X HDFS for single job
 - Intel® EE for Lustre = 5.5 X HDFS for concurrent workload
- <http://insidebigdata.com/2014/09/29/performance-comparison-intel-enterprise-edition-lustre-hdfs-mapreduce/>

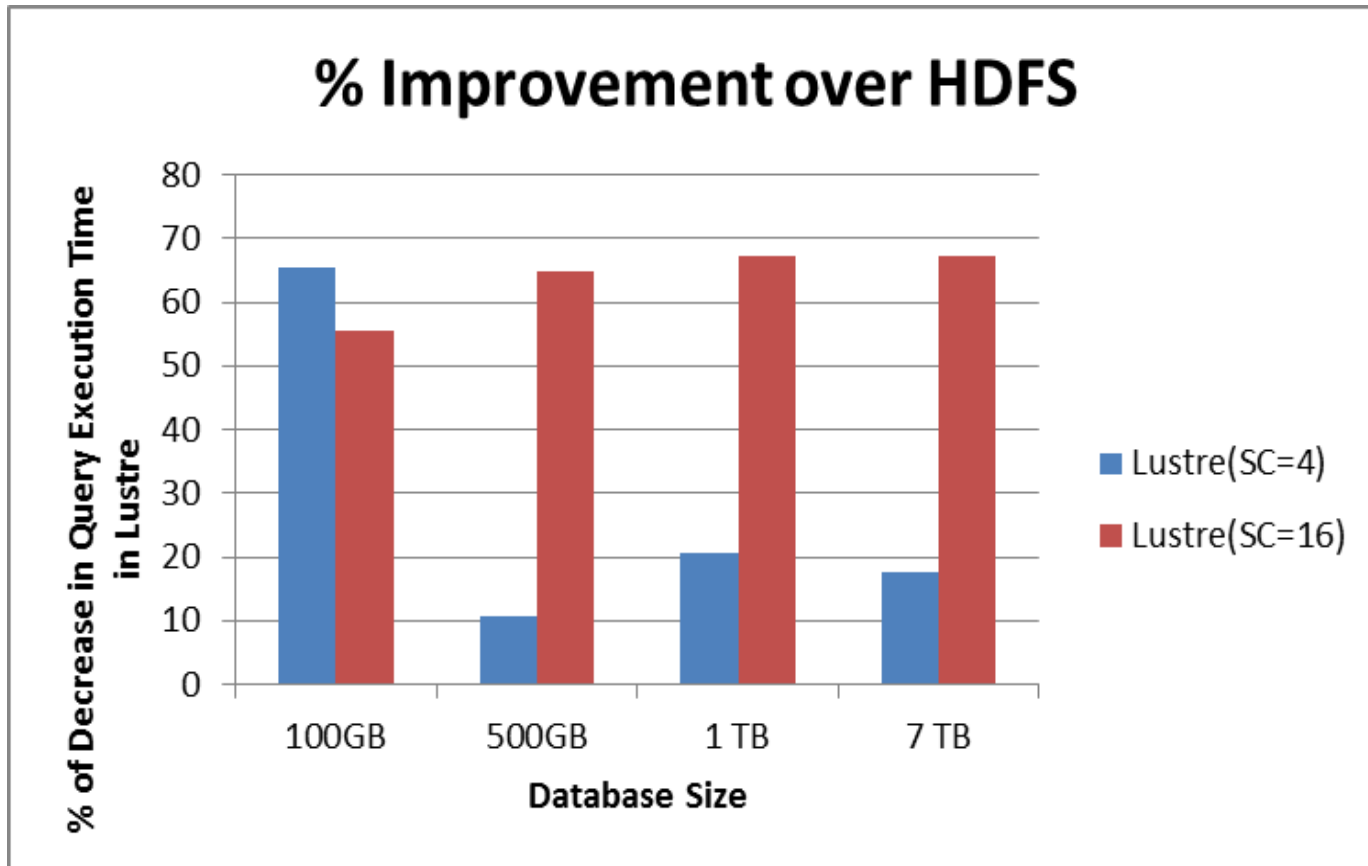
Degree of Concurrency = 1



Intel® EE for Lustre* = 3 X HDFS for optimal SC settings*

http://www.eofs.eu/fileadmin/lad2014/slides/21_Rekha_Singhal_LAD2014_Lustre_vs_HDFS_MR.pdf

Degree of Concurrency = 1



Intel® EE for Lustre* optimal SC gives 70% improvement over HDFS

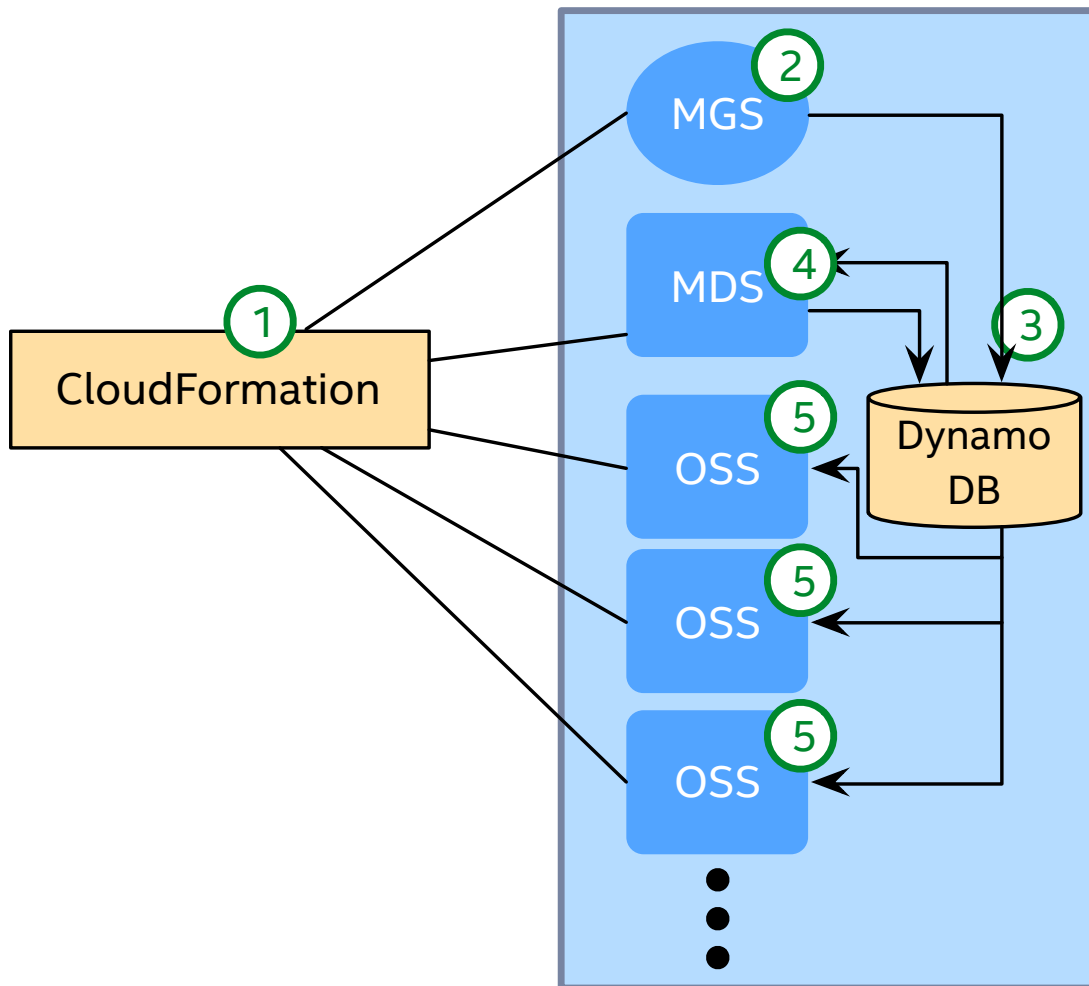
http://www.eofs.eu/fileadmin/lad2014/slides/21_Rekha_Singhal_LAD2014_Lustre_vs_HDFS_MR.pdf

Intel Cloud Edition for Lustre*



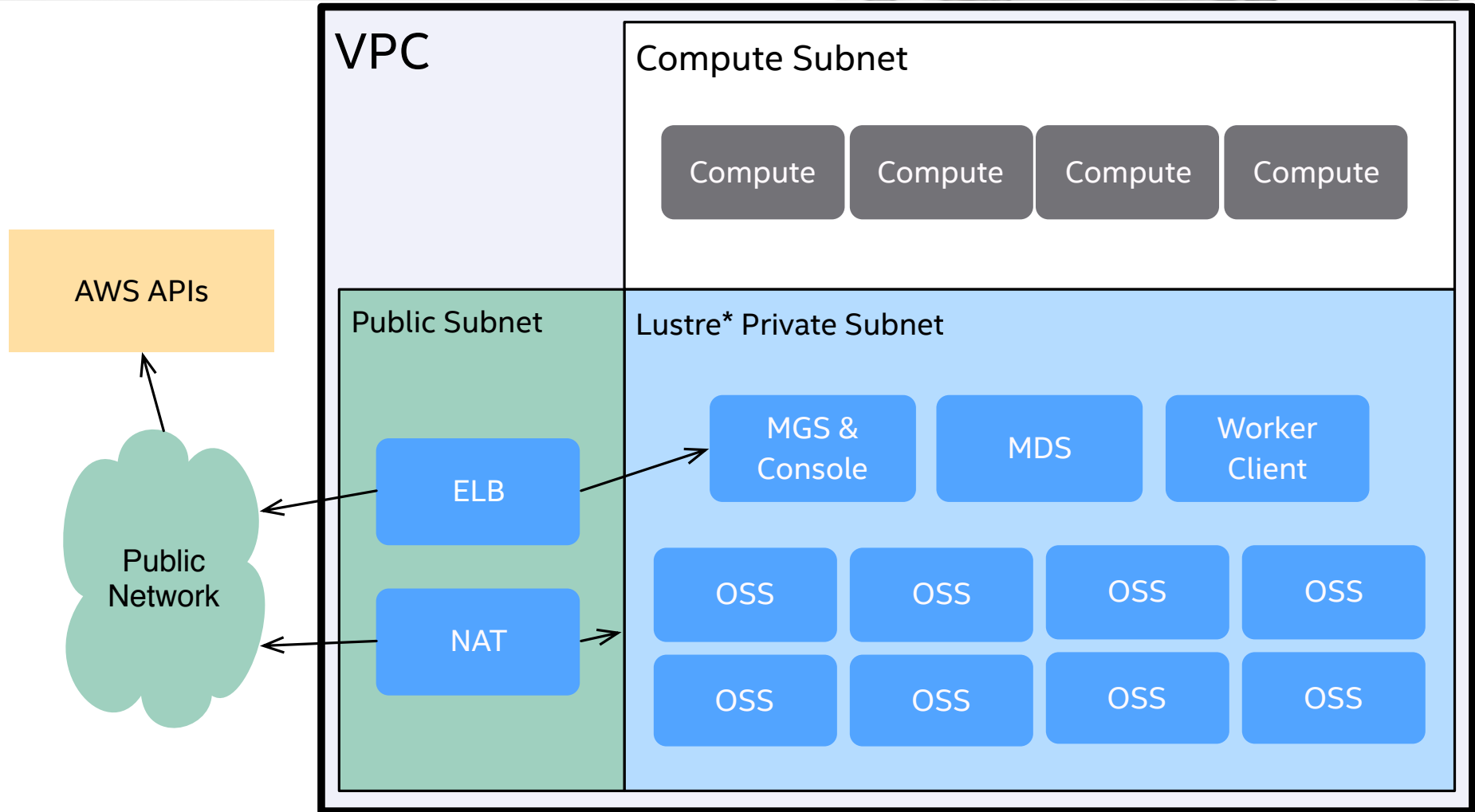
- Three Intel Lustre AMIs available on AWS Marketplace*:
 - Community Version
 - Global Support
 - Global Support HVM
- Lustre v2.5.3
- Automated Lustre configuration
- Lustre monitoring with Ganglia and LMT/Itop
- Automated S3 data import
- Global Support includes additional capabilities:
 - Intel Lustre Support
 - High Availability(requires VPC)
 - Enhanced Networking (HVM + VPC)
 - Higher performance instance types

Automated Lustre* Deployment



- ① CloudFormation creates a stack of AWS resources from a template
- ② MGS Initializes itself
- ③ MGS updates DB with NID
- ④ MDS formats MDT, registers with MGS, updates DB.
- ⑤ OSSs format local targets, updates DB

Using Virtual Private Cloud



Automated S3 Import

- Option to import an S3 bucket into a new Lustre^{*} filesystem
- Initially only the file metadata is imported
- File contents are retrieved from S3 on demand and stored on Lustre

Lustre* HA in the cloud

- HA template is available
 - Storage is managed independently of instances
- Failure scenario
 - AWS AutoScaling detects and terminates a failed instance
 - AutoScaling creates a new instance
 - New instance identifies orphaned storage and network interface
 - “Adopts” the storage and NIC
 - Target is brought back online and recovers

ICE-L Monitoring tools

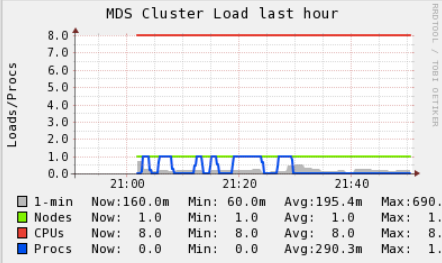
MDS (physical view)

CPUs Total: **8**
 Hosts up: **1**
 Hosts down: **0**

Current Load Avg (15, 5, 1m):
 1%, 2%, 2%

Avg Utilization (last hour):
 2%

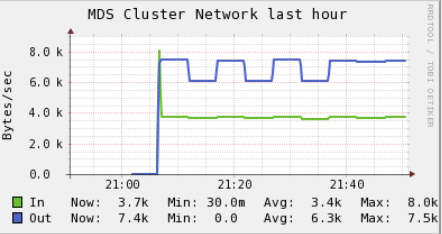
Localtime:
 2013-09-09 21:50



MDS Cluster Load Last hour

Legend: 1-min (grey), Nodes (green), CPUs (red), Procs (blue)

Now: 160.0m, Min: 60.0m, Avg: 195.4m, Max: 690.0m



MDS Cluster Network Last hour

Legend: In (green), Out (blue)

Now: In 3.7k, Out 7.4k; Min: In 30.0m, Out 0.0; Avg: In 3.4k, Out 6.3k; Max: In 8.0k, Out 7.5k

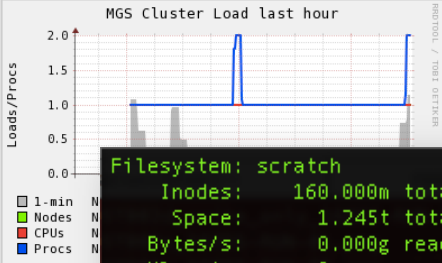
MGS (physical view)

CPUs Total: **1**
 Hosts up: **1**
 Hosts down: **0**

Current Load Avg (15, 5, 1m):
 27%, 54%, 113%

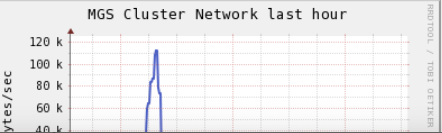
Avg Utilization (last hour):
 26%

Localtime:
 2013-09-09 21:50



MGS Cluster Load Last hour

Legend: 1-min (grey), Nodes (green), CPUs (red), Procs (blue)



MGS Cluster Network Last hour

Legend: In (green), Out (blue)

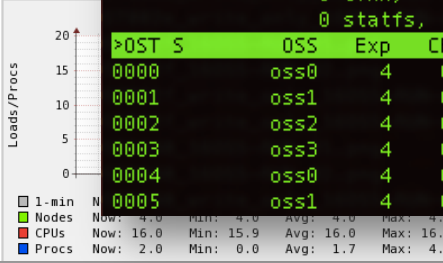
OSS (physical view)

CPUs Total: **16**
 Hosts up: **4**
 Hosts down: **0**

Current Load Avg (15, 5, 1m):
 0%, 0%, 0%

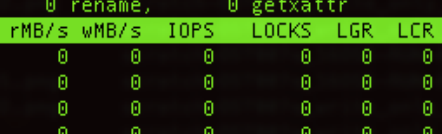
Avg Utilization (last hour):
 0%

Localtime:
 2013-09-09 21:50



OSS Cluster Load Last hour

Legend: 1-min (grey), Nodes (green), CPUs (red), Procs (blue)



OSS Cluster Network Last hour

Legend: In (green), Out (blue)

```

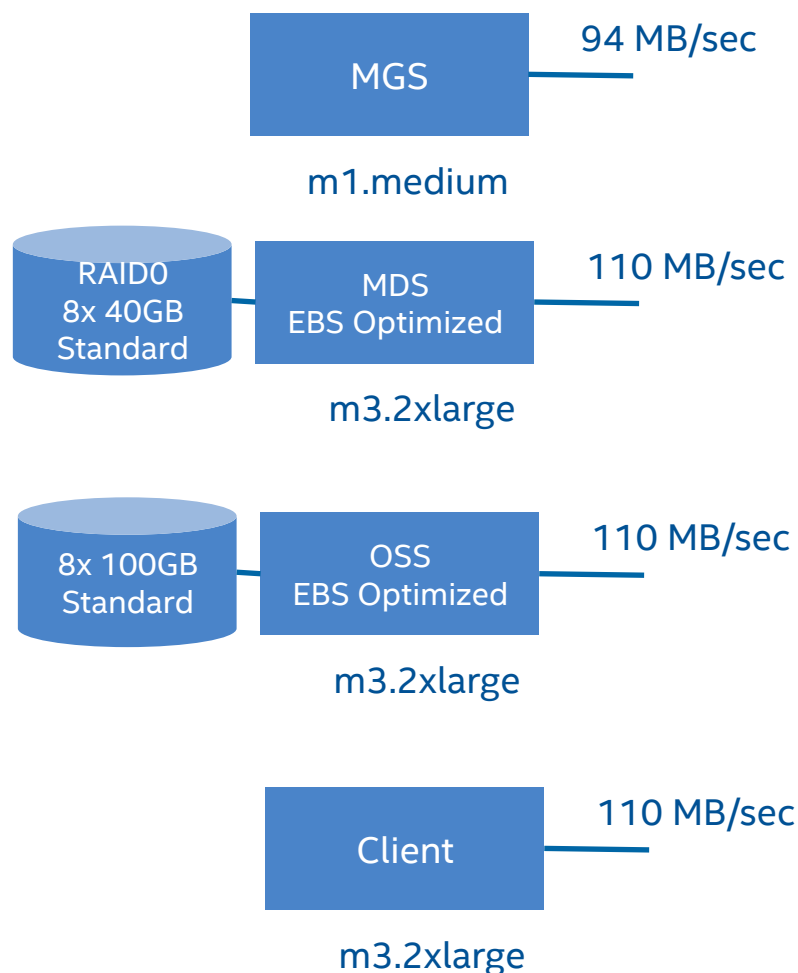
Filesystem: scratch
Inodes:      160.000m total,    0.000m used ( 0%),   160.000m free
Space:       1.245t total,    0.014t used ( 1%),   1.232t free
Bytes/s:     0.000g read,     0.000g write,       0 IOPS
MDops/s:     0 open,         0 close,            0 getattr,    0 setattr
              0 link,        0 unlink,          0 mkdir,     0 rmdir
              0 statfs,      0 rename,          0 getxattr

>OST S      OSS      Exp  CR rMB/s  wMB/s  IOPS  LOCKS  LGR  LCR  %cpu  %mem  %spc
0000      oss0      4    0    0    0    0    0    0    0    0    8    1
0001      oss1      4    0    0    0    0    0    0    0    0    8    1
0002      oss2      4    0    0    0    0    0    0    0    0    8    1
0003      oss3      4    0    0    0    0    0    0    0    0    8    1
0004      oss0      4    0    0    0    0    0    0    0    0    8    1
0005      oss1      4    0    0    0    0    0    0    0    0    8    1
    
```

* Some names and brands may be claimed as the property of others.

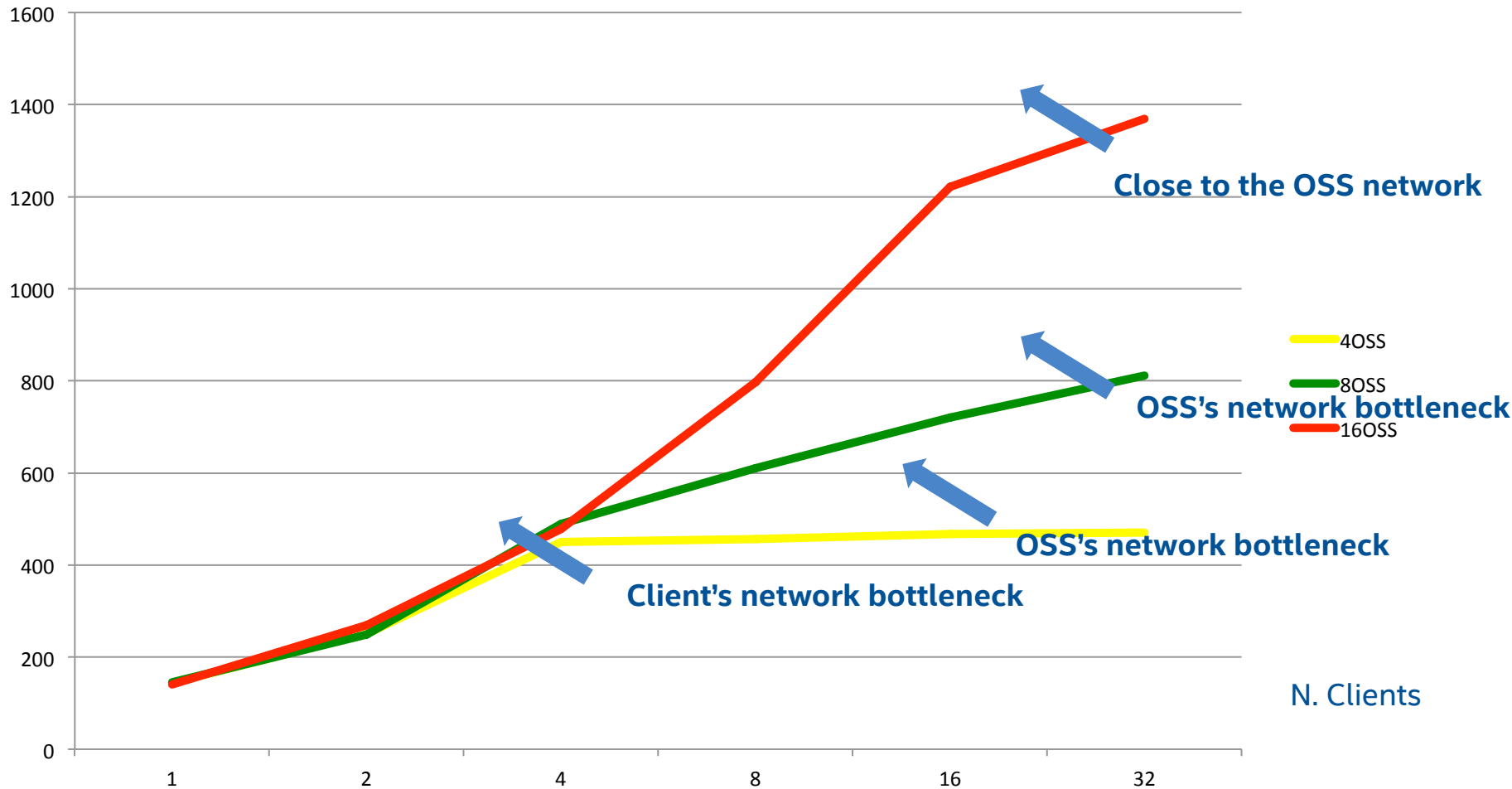
Large File Benchmark

- Using IOR benchmark
- Comparing 3 Lustre cluster configurations
- Increase the number of OSSs
 - 4 OSS
 - 8 OSS
 - 16 OSS
- Configurations of MGS and MDS are fixed
- 1-32 clients



IOR Sequential Read FPP

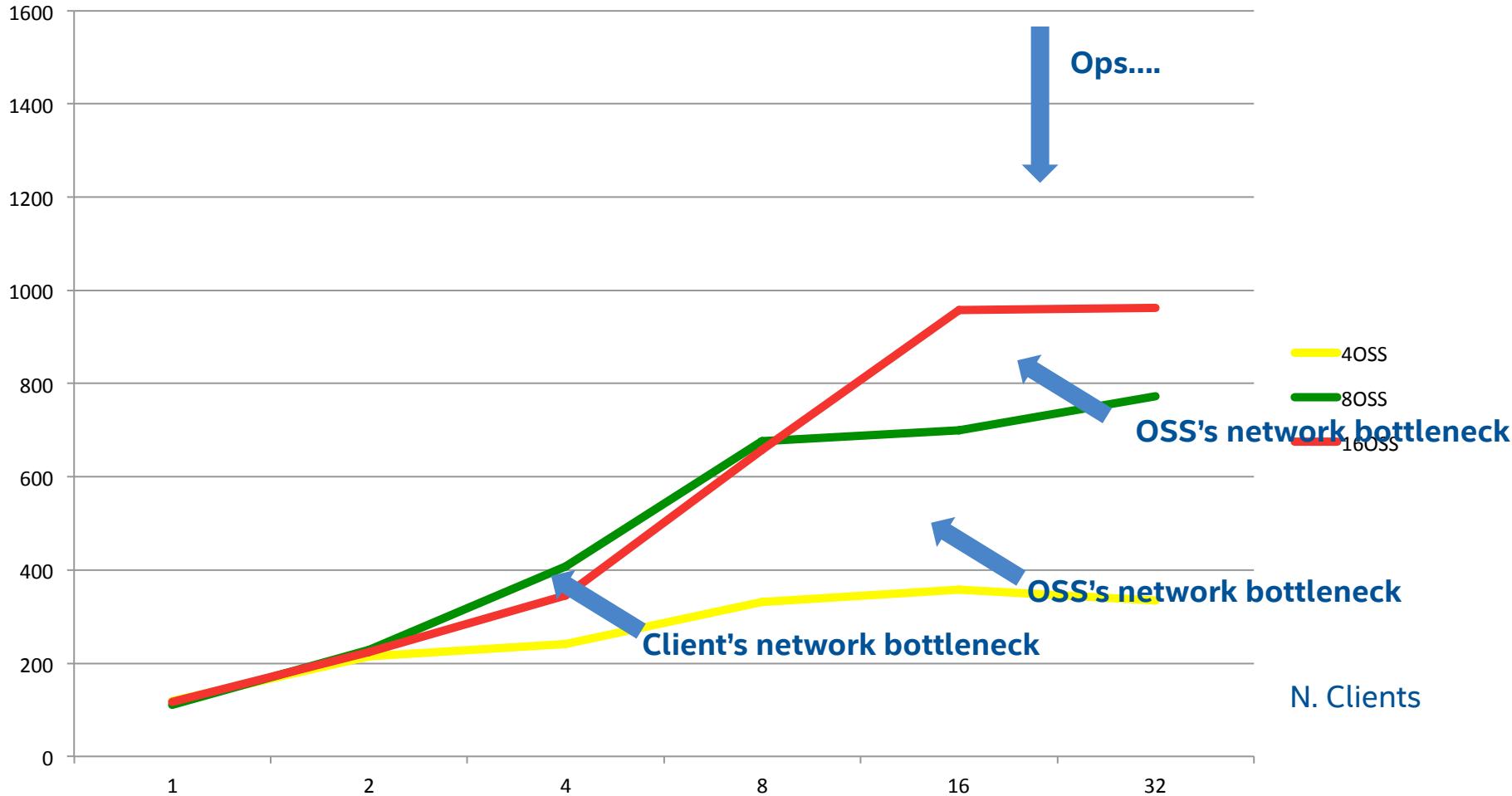
MB/sec



N. Clients

IOR Sequential Write FPP

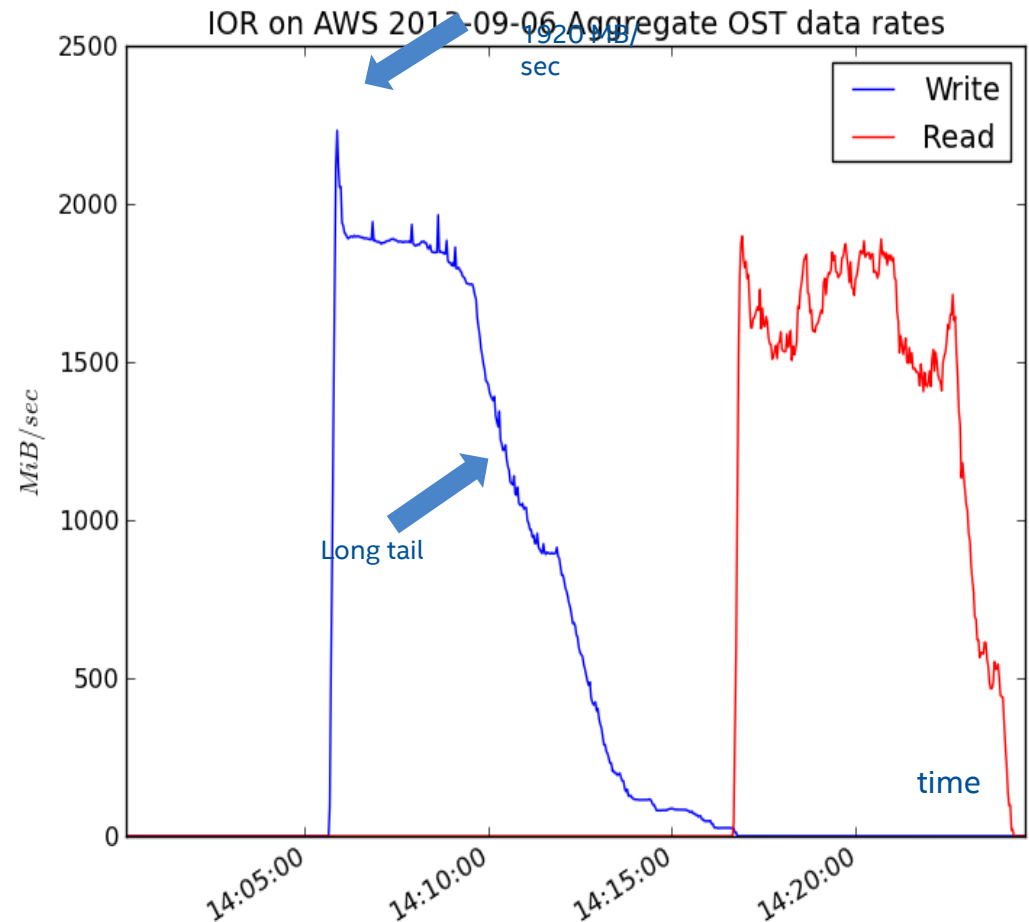
MB/sec



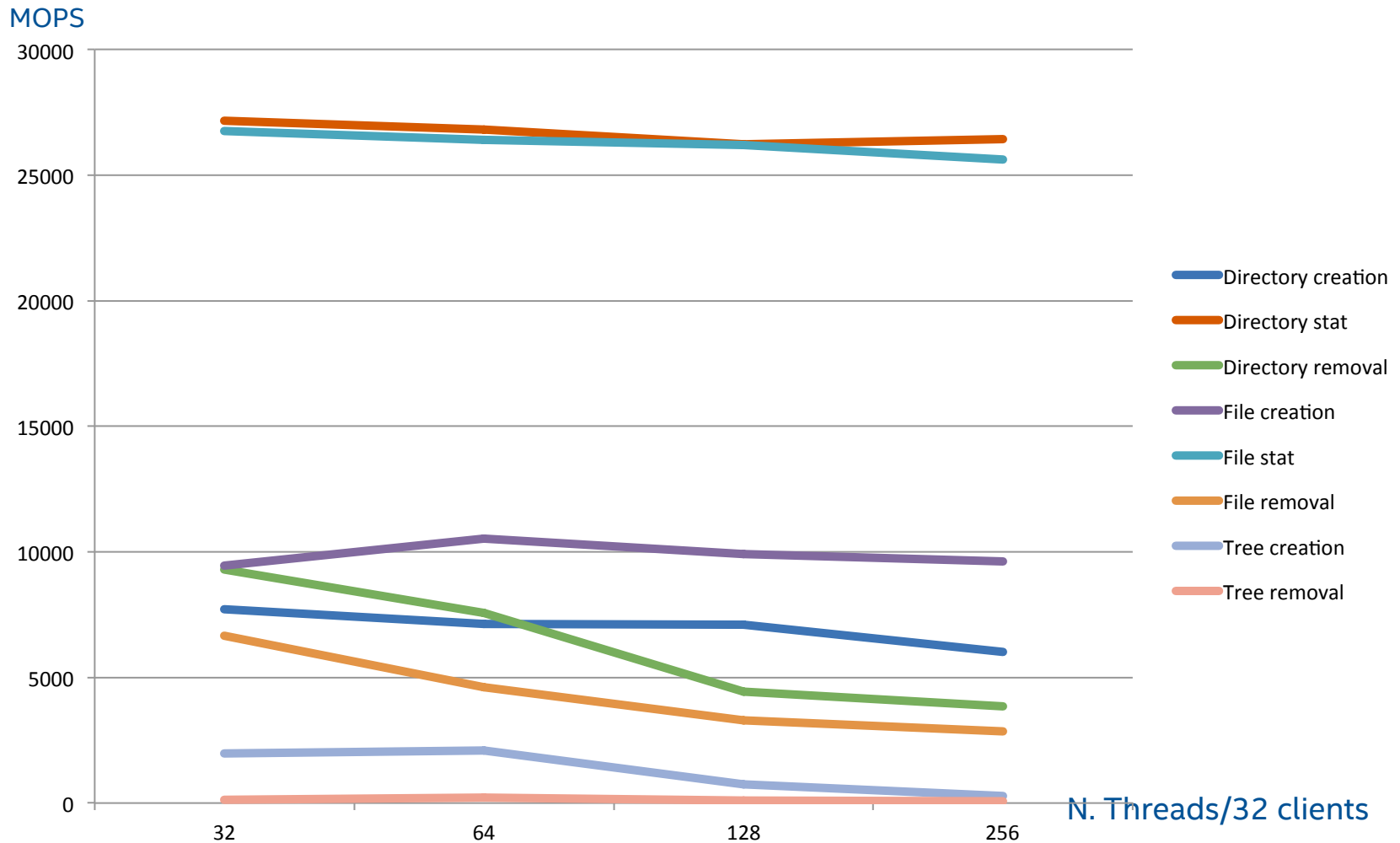
Aggregate Performance During Run

LMT used to record the OST metrics during the IOR run.

With a simple python script we create this graph: “aggregate performance vs time” to analyze the problem.



MDTEST on 16 OSS Cluster Configuration





Thank You



OpenFabrics Software
User Group Workshop

#OFSUserGroup

* Some names and brands may be claimed as the property of others.