Lustre* on Amazon Web Services*

High Performance Data Division

Robert Read

robert.read@intel.com

April 16, 2013

Lustre on Amazon Web Services

- Goal
  - Provide a scalable, shared filesystem for HPC applications on the cloud.
- Lustre Advantages
  - POSIX namespace
  - Maximizes use of available resources
  - Very scalable

Storage on AWS

- Storage Options
  - Ephemeral storage
    - Local storage to the instance
    - Directly attached, fastest option
    - Limited options for size
    - Disappears when instance terminates
  - Elastic Block Storage (EBS)
    - Networked storage
    - Max size 1TB per EBS volume
    - Persistent, can outlive instance
    - Not magic, still suffers from usual storage woes
  - S3 is for durable storage
    - Not coherent

Recent Enhancements

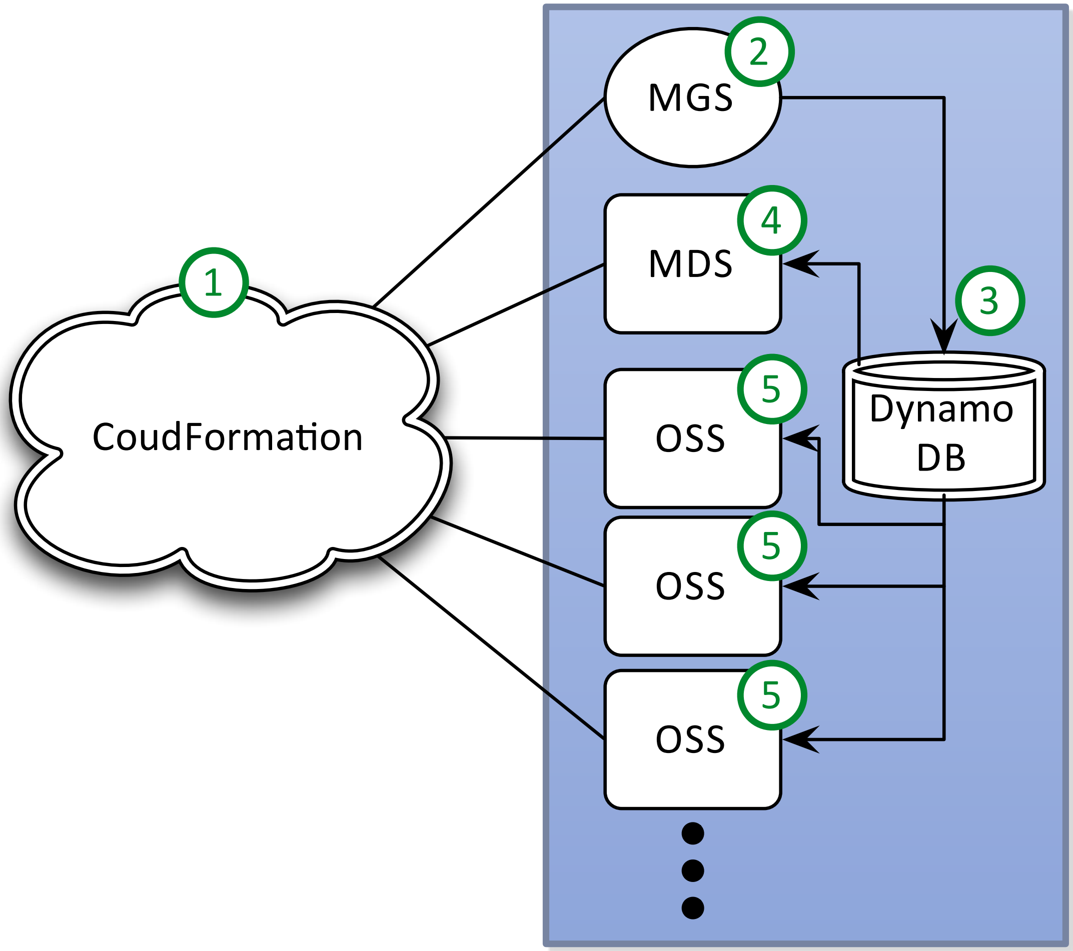
- EBS Optimized instances
  - Dedicated 500Mb/s or 1000Mb/s link for EBS
  - Effectively doubles throughput of a server node
- EBS with Provisioned IOPS
  - 100-2000 IOPS
- High I/O instances
  - 2x 1TB SSD volumes (ephemeral)
  - Can be used Cluster Compute placement group
  - Random IOPS: 120k read, 10k-85k write
  - Sequential IO: 2GB/s read, 1.1.GB/s write
- High Storage instances
  - 24x 2TB disks (ephemeral)
  - Can be used in Cluster Compute placement group
  - Sequential IO: 2.4 GB/s read, 2.6 GB/s write

(Numbers as reported by Amazon)

Deploying Lustre on AWS

- Custom Lustre Server AMI
  - Centos 6.3
  - Lustre master (pre-2.4)
- Deploy cluster with CloudFormation
  - m1.xlarge (4 core, 15GB) + EBS Optimized
  - One Availability Zone
- New filesystem is assembled as nodes boot
- Minimal coordination through DynamoDB

Loosely Coupled Lustre Initialization



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

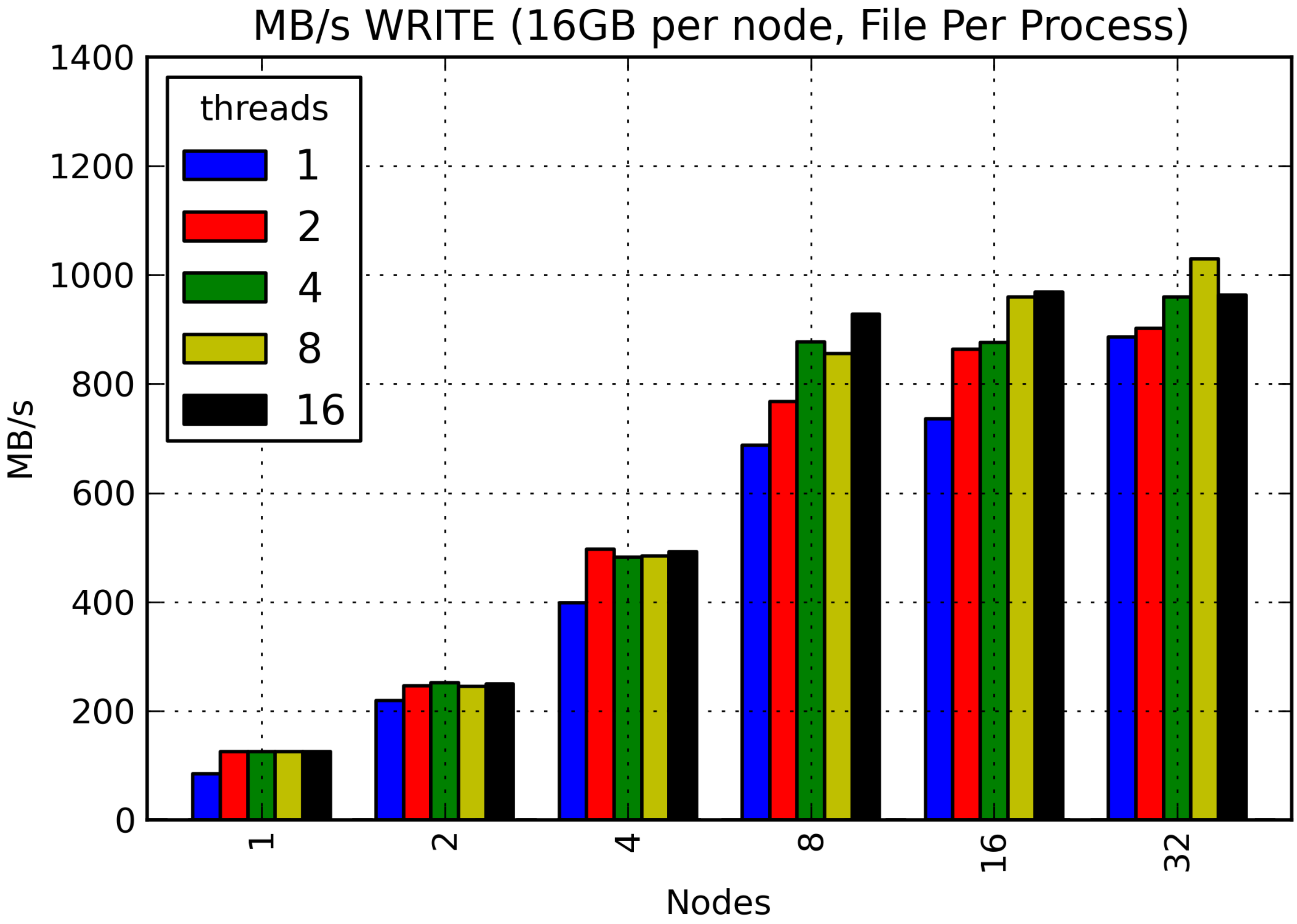
Lustre Benchmarks

- Initial benchmarking to “kick the tires”
- Focus on micro-benchmarks
  - IO bandwidth
  - creates/sec
- More thorough evaluation of various options in progress

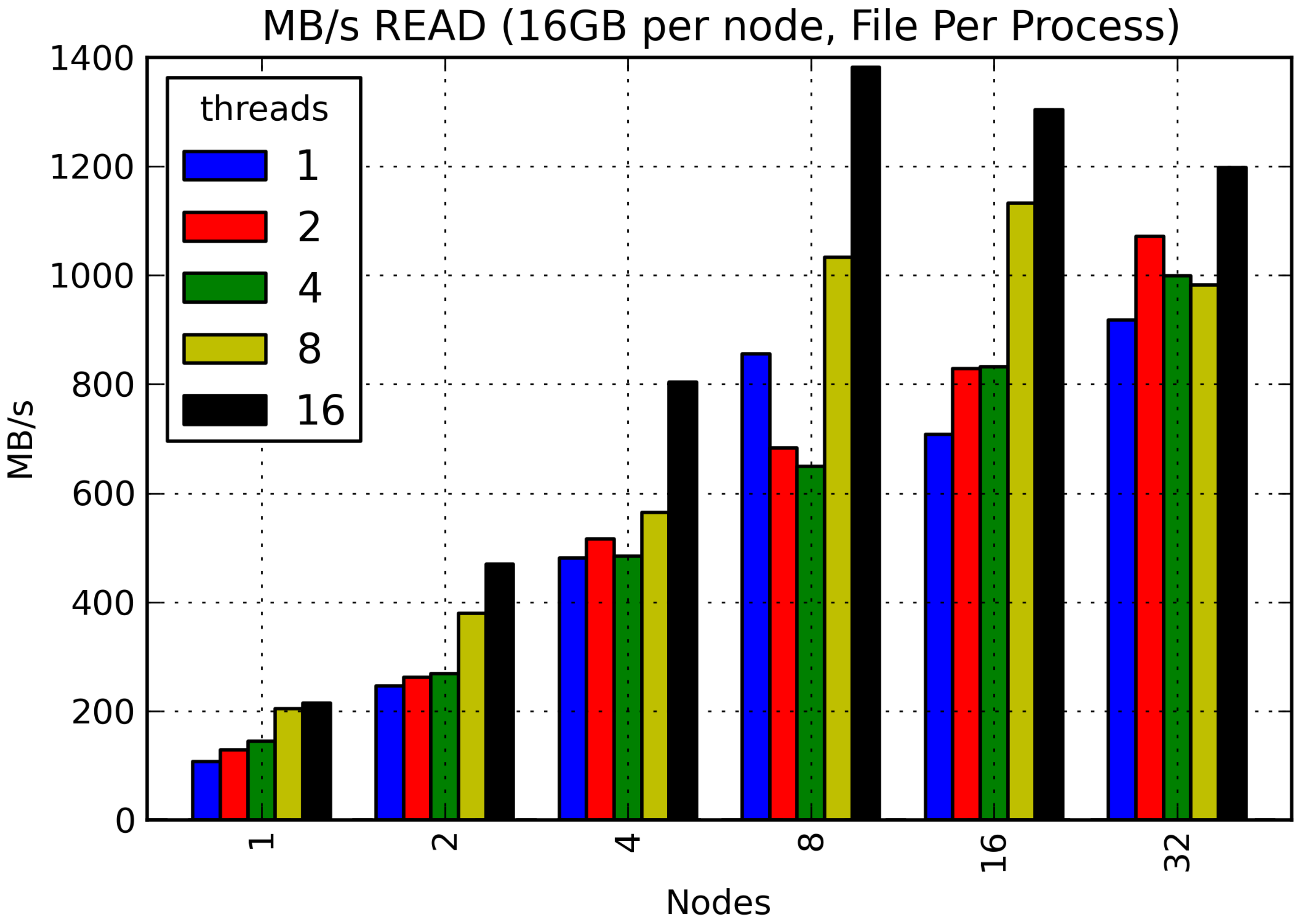
IOR Benchmarking Configuration

- MDS
  - m1.xlarge
  - 8x 40GB EBS volumes
  - RAID0
- 10 OSS
  - m1.xlarge
  - 4x 100GB EBS volumes
  - RAID0
- 32 Clients
  - m1.xlarge
  - 1 to 16 threads
  - 16GB per client

Sequential Write (FPP)



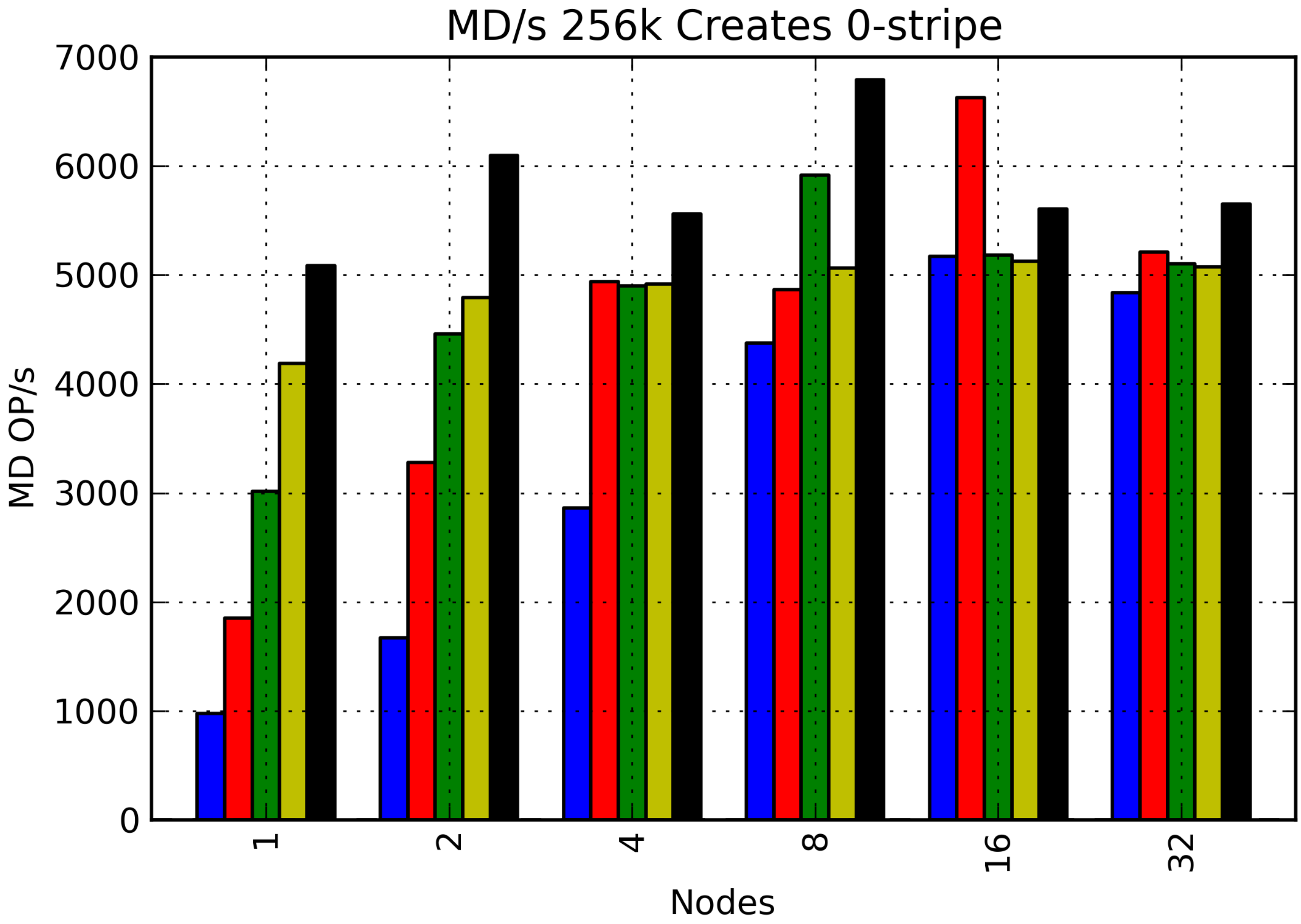
Sequential Read (FPP)



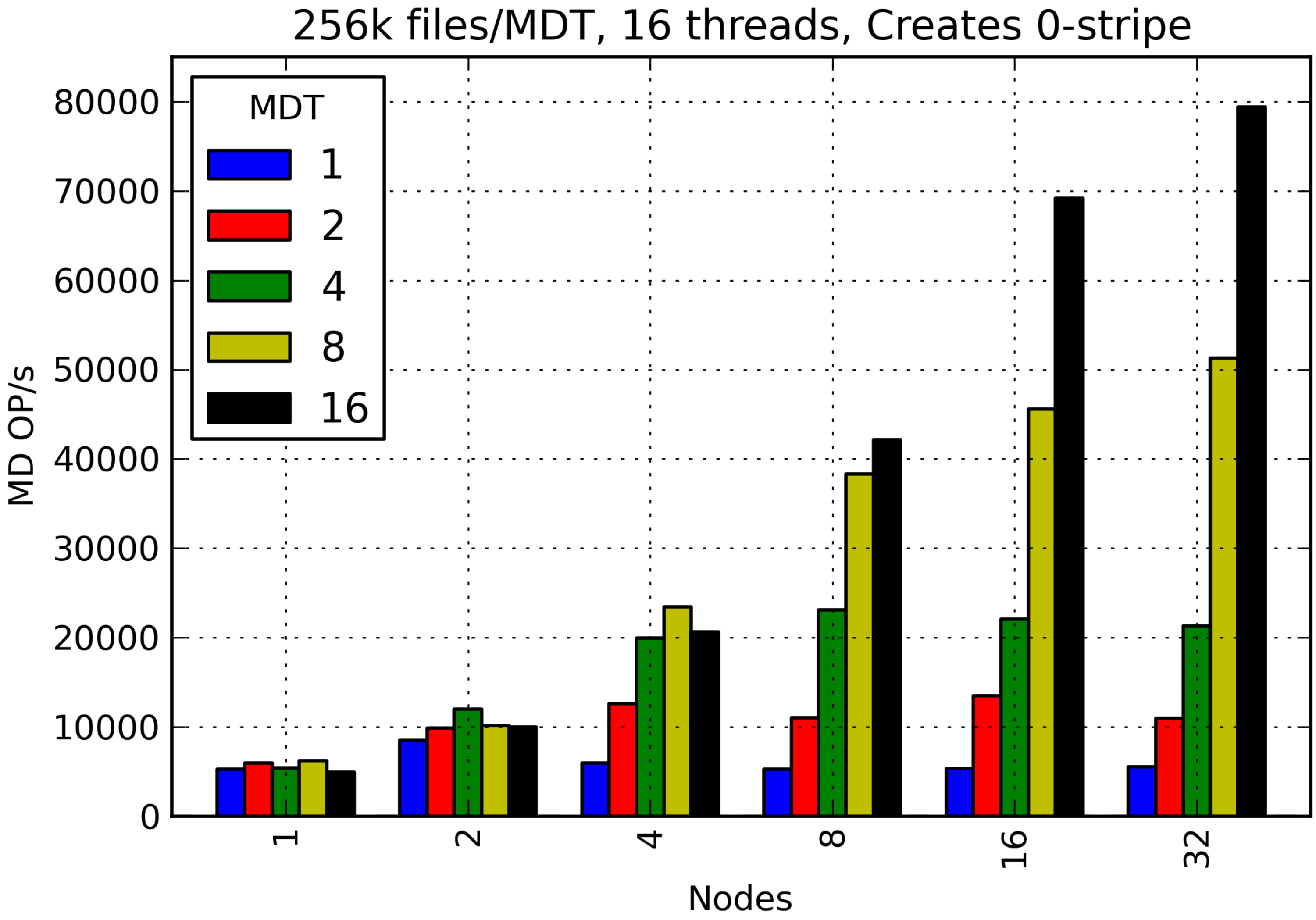
Metadata – mdsrate Configuration

- MDS
  - (same)
- 2 OSS
  - 4x 40GB EBS
- 32 Clients
  - 8 mounts per client
  - Up to 16 threads per client
  - 1 thread per directory

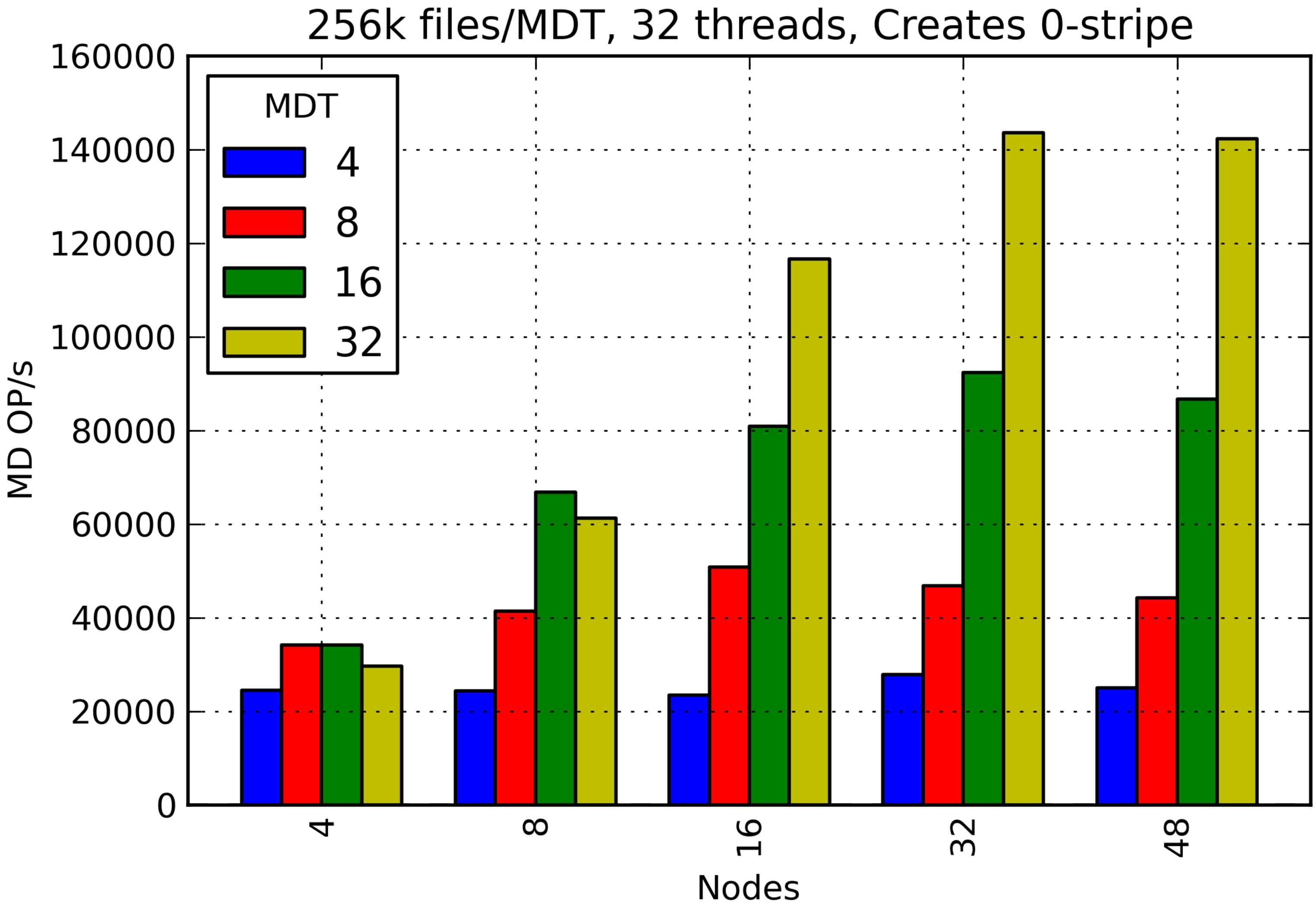
Metadata Performance



Scalable Metadata Performance (DNE)



Scalable Metadata Performance (DNE)



Early Conclusions

- Positives
  - Lustre performs well
  - AWS Architecture allows for scaling as needed
  - New DNE feature is a great fit
  - Fully programmable environment simplifies deployment

- Room for improvement
  - Lustre needs a more dynamic failover capability
  - Data management will be an issue
    - HSM meets S3?

intel®