2015 Computer System, Cluster and Networking Summer Institute

Team Saffron

Lustre Testing on Tamirs

MDT and VM tests

June 5th, 2015

Mike Mason, HPC-3

Chris Mitchell, HPC-3

Brad Settlemyer, HPC-5

LA-UR-15-24108

Overview

- Introductions
- Overview of Parallel File Systems
  - Lustre
- Kove memory box as MDT
- Overhead of MDS monitoring
- Client VMs to increase performance

Mike Mason

- HPC-3, production group
- Tech Lead for HPC Monitoring
  - How do you monitor 10 clusters, 4 petabyte file systems and all the infrastructure?
- File Systems Administrator
  - 5 Lustre file systems, total 12 PB
  - 4 Sonexion (Lustre), total 80 PB
  - 5 Panasas, total 7 PB
  - NFS home/project space

Christopher Mitchell

- HPC-3, Production Infrastructure Team
- Systems Architect & File Systems Admin
  - What does the next generation system look like, when can we order it, how does it go together?
- File Systems Administrator
  - NFS Servers (home/projects)
  - Lustre
  - Data Transfer Node System

Brad Settlemyer

- HPC-5 - Systems Integration Group
- Research Scientist
  - Maintain ties to production systems
    - Work with HPC-3 and code teams anytime I can
  - Improve the way LANL purchases leadership-class computers and file systems
    - Data-driven analysis of data center
  - Improve the way LANL operates and uses leadership-class computers and file systems
    - Participate in the I/O research community
    - Follow commercial trends closely

Overview of Parallel File Systems

LA-UR-15-24108

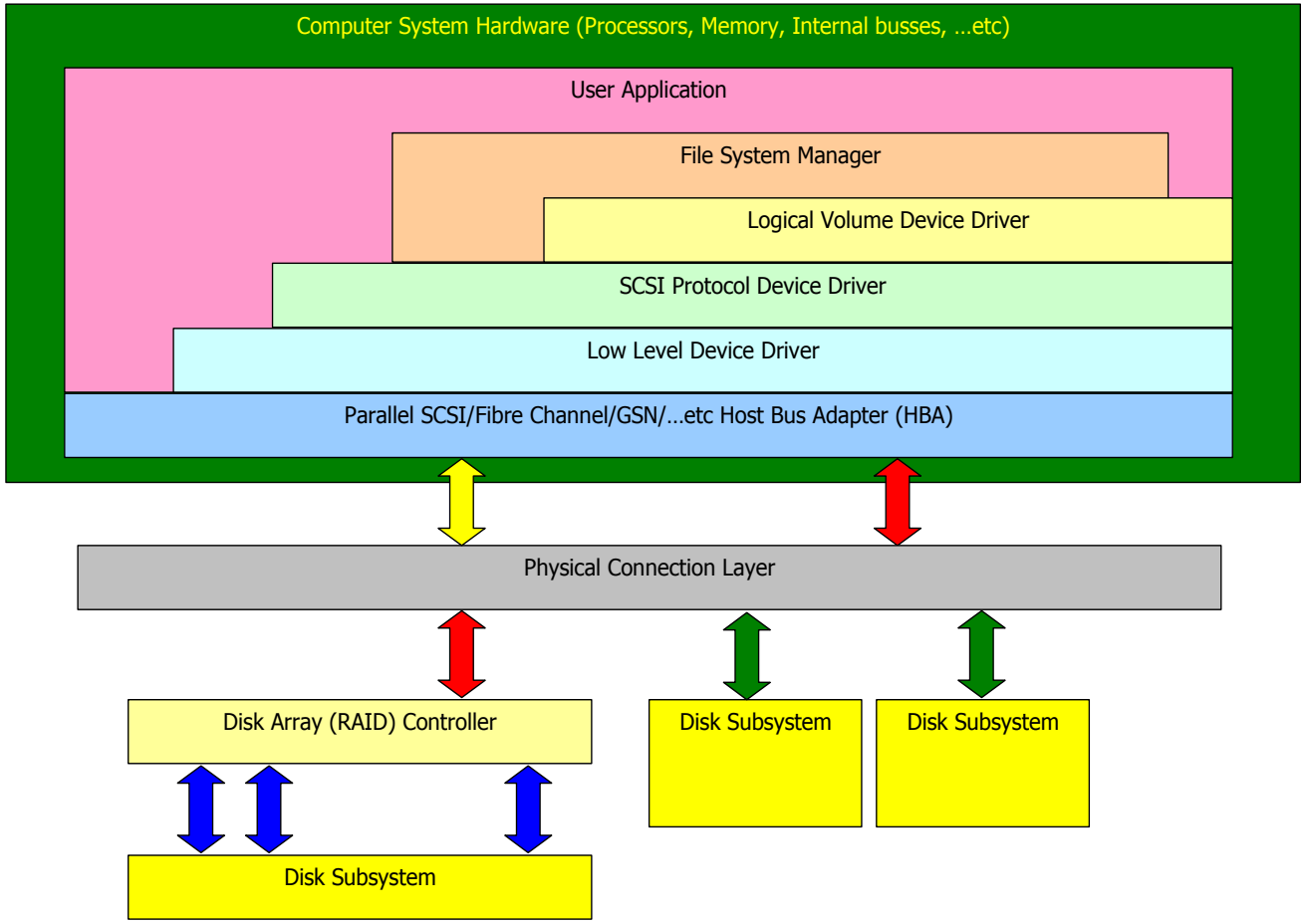
Why Use a File System?

- Execute Code
  - Linux uses Fork and Exec model
  - Exec leverages Mmap
- Write data file
  - Most common operation on parallel file systems
  - Includes a lot of hidden metadata
- Read data file
  - More general than Exec
  - Most common operation on traditional file systems
- It's reliable!

What is a File?

- Three parts to a file!!!
  - Inode/metadata
  - File data
  - Entry in parent directory
- A directory is a file
  - The file data is a formatted list mapping names to inodes
- Root of many problems in large file systems is that directory entries must be unique
  - Leads to serialization of many file system tasks

Storage Hierarchy



Computers and Hard Drives

- Just as a Computer Cluster is a lot of individual computers glued together with software working as a computing unit
- A File System is a lot of individual hard drives glued together with software working as a storage unit

LA-UR-15-24108
Operated by Los Alamos National Security, LLC for the U.S. Department of Energy's NNSA

Shared File Systems

Traditional Network File Systems

- Multiple Clients
  - Clients maintain state
- Single Server State
  - Multiple servers possible, but must maintain a single shared state
- Data stored into single view of storage
  - Parallelism only in backend
- NFS, Samba/CIFS, QFS

Distributed/Parallel File Systems

- Multiple Clients
  - Servers maintain state (why?)
- Multiple Servers
  - Server's manage federated storage
- A single client may access multiple servers in parallel
- Better aggregate throughput

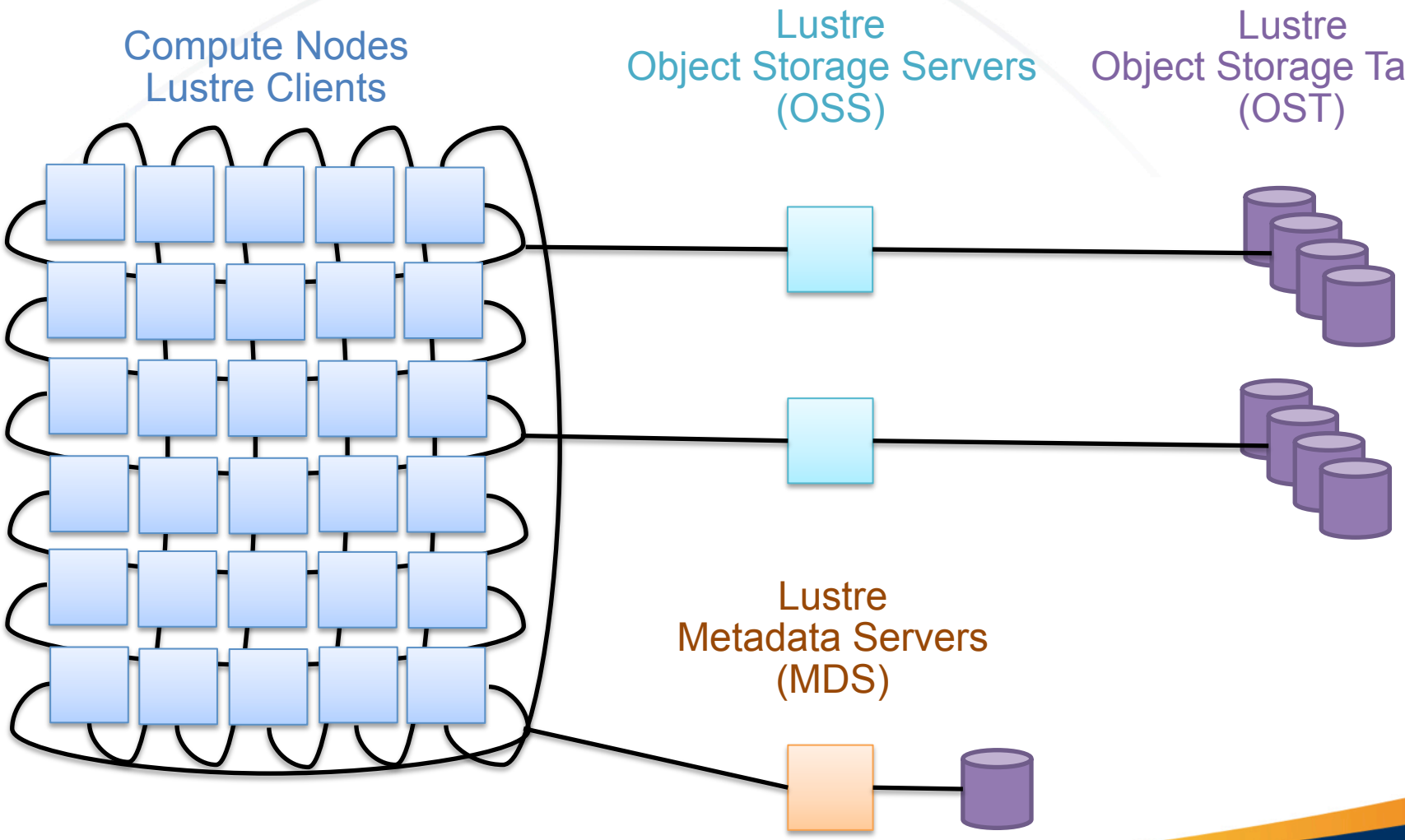
Comparing Parallel File Systems

Feature	Lustre	GPFS	PanFS	OrangeFS	Ceph
Open Source?	**Yes**	*No*	*No*	**Yes**	**Yes**
Support diverse storage?	**Yes**	*No*	*No*	**Yes**	**Yes**
High Performance?	**Yes**	**Yes**	*No*	*No*	*No*
High Reliability?	**Yes**	**Yes**	**Yes**	*No*	*Maybe*
Support High Capacity?	**Yes**	**Yes**	**Yes**	*No*	**Yes**
Easy administration?	*No*	*No*	**Yes**	*No*	**Yes**
Posix Support?	**Yes**	**Yes**	**Yes**	*No*	*No*
Support capacity growth?	**Yes**	*Maybe*	**Yes**	*No*	**Yes**

Lustre Terminology

- Basic Lustre Abstractions
  - Lustre Clients
    - Computation nodes that mount or access the Lustre file systems
  - MetaData Servers (MDS)
    - Manages inodes, directory entries, attributes, and layout info
    - Typically stores data locally in a MetaData Target (MDT)
  - Object Storage Servers (OSS)
    - Used to service I/O requests and manage locks for locally stored data
  - Object Storage Targets (OST)
    - Used by OSS to store file data
    - Formatted as Ldiskfs (ext4 w/ patches)
    - Also supports ZFS

Basic Lustre Configuration

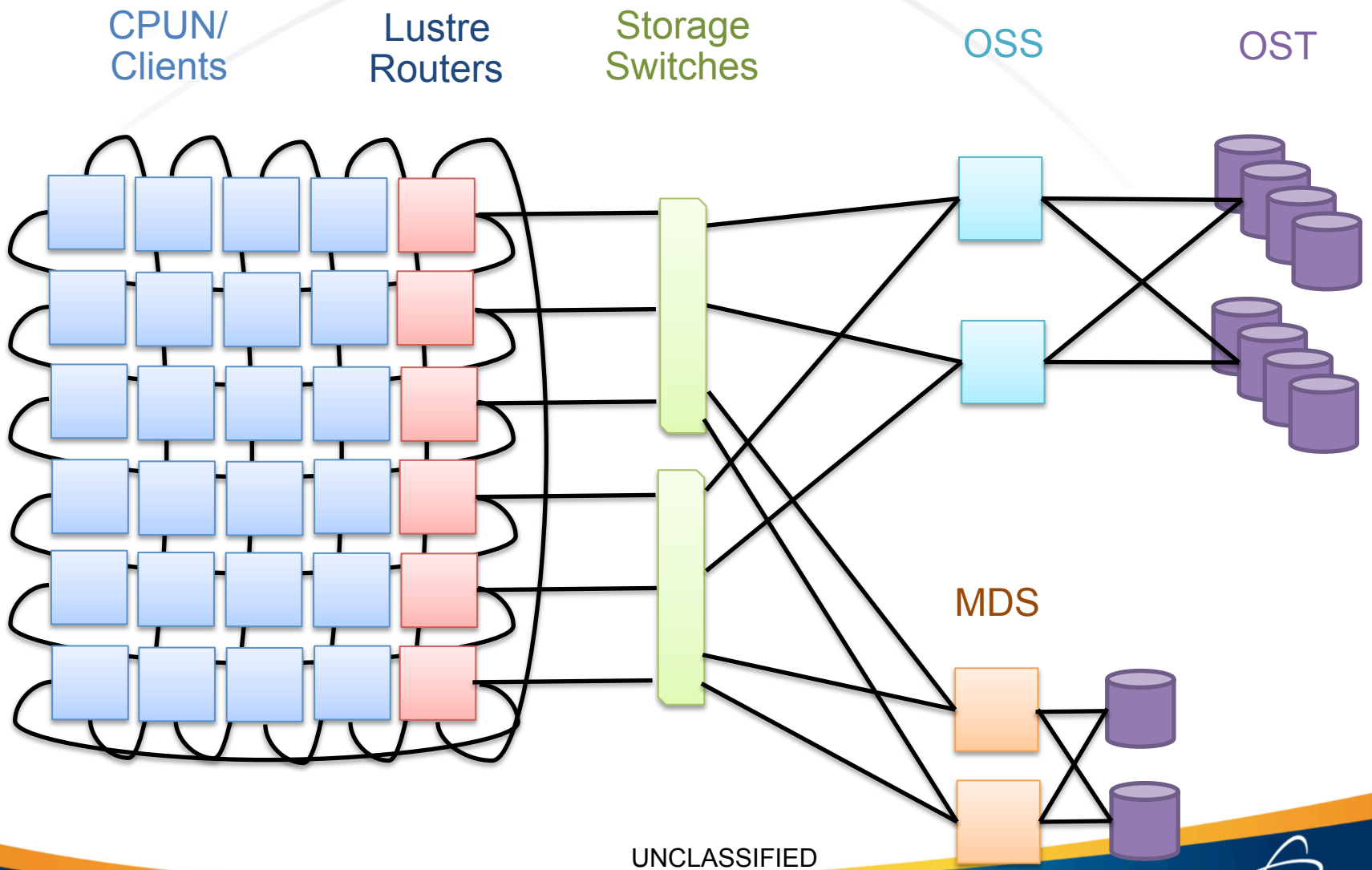


LA-UR-15-24108

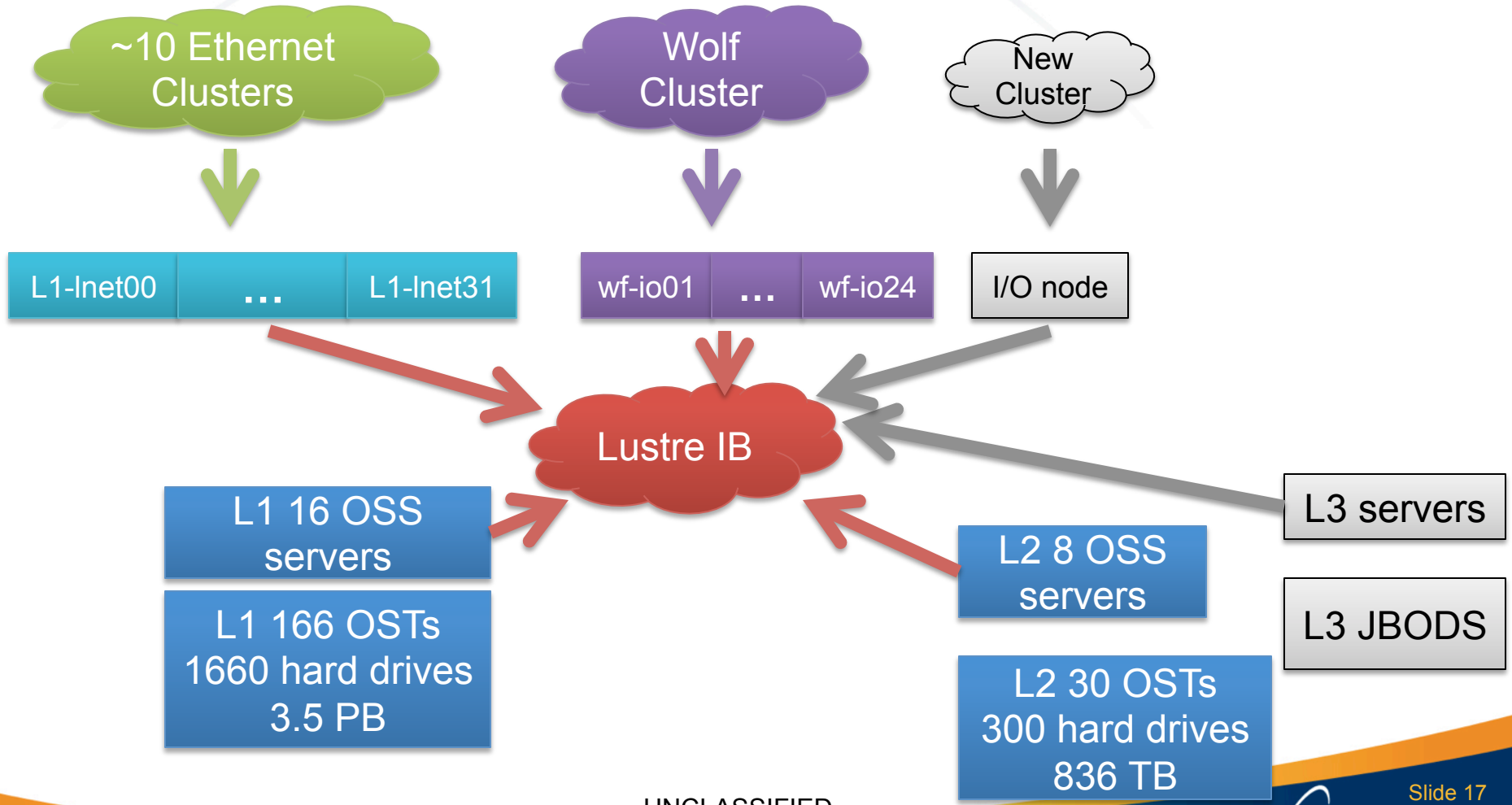
Further Lustre Terminology

- Separation of Server and Target for failover
- Lustre speaks LNET
  - All messages passed by Lustre are called RPCs
  - Supports SRP (SCSI RDMA protocol)
- Lustre Routers translate between networks
  - e.g., Cray Gemini to Lustre Infiniband network
  - Necessary for center-wide file systems
- Multiple metadata servers
  - Distributed metadata is available in Lustre
  - ORNL simply provides multiple mounts

Realistic Lustre Configuration



LANL Open Lustre Infrastructure



Our Projects

LA-UR-15-24108

Kove xpd as MDT

- Kove xpress disk (xpd)
  - The XPD is a server appliance that comes loaded with 1TB of RAM and is backed by a dedicated UPS to make the memory non-volatile.
    - Hard Drive array installed in system to save RAM contents when battery runs low.
    - 6x QDR Infiniband connections from the node to the cluster fabric (6x 32Gbps links)

Kove xpd as MDT

- Lustre MDT
  - The storage target that holds the metadata for the Lustre file system –
    hence MDT = MetaData Target
    - This metadata includes things like file names, directory paths, creation time, file size, etc.
  - All Lustre file system operations require one or more metadata requests to complete and the faster these operations are, the faster the overall request completes.
- Leverage the fact that the xpd is a box of RAM that is persistent and see what speed improvement is possible when the MDT is on the xpd vs disk.

MDS data collection

- Lustre metadata operations are "*slow*"
  - This is relative – we want to create 1 million files in 6 seconds
- How do we make it faster?
  - If this had an easy answer, it wouldn't have required 10 years of engineering just to get to slow
    - Lustre must be reliable, and adding multiple metadata servers makes reliability harder

So how do we make metadata ops faster?

- **Data!** The more, the better
  - Determine what the most common operations are ***during periods of slowdown***
    - Don't optimize the idle loop!
  - Collecting data isn't free, it's additional work on an already *slow* system
  - Measure the costs, figure out what we can collect cheaply, and collect it
  - Share it with the research community!

Client VMs to Increase Performance

- Lustre has some problems with single client performance
  - No matter how big the pipe from the node
    - The network connection 1G, 10G or more
    - The lustre client will max out before the available bandwidth is used
- We can get around this by putting multiple clients on a single node
  - Using Virtual Machines
    - Each VM will act as a Lustre client
    - We saturate the pipe with multiple VMs
  - How many VMs per node should we have?
  - What's the best way to setup VMs and distribute work?

Questions?

- About Lustre
- About Us
- About What We Do