LUG 2019:
Lustre New User Training

Michael Timmerman

Hewlett Packard Enterprise

michael.timmerman@hpe.com

Class Agenda

- Intro (15 Min)
- What is Lustre (30 Min)
- Architecture (60 Min)
- Break (15 Min)
- Installation/Operations (30 Min)
- Users and Applications (30 Min)
- References and Community Resources (15 Min)
- Questions (45 Min)

Intro

The Lustre New Users Training is targeted for users familiar with High Performance Computing but are new to the Lustre file system.

The goal of the class is to acquaint the attendee with the history of Lustre, it's architecture, the concepts of installation and operations, the how to for common user and application interfaces, and a short list of reference and community resources.

- Me
- You
- Bathrooms and other important stuff
- Questions
  - ask or not to ask
  - when to ask or when not to ask

What is Lustre?

- Marketing Speak
- Technical Mumblings
- Yesterday – A very brief history of Lustre
- Today – Current status
- Tomorrow – Futures timeline

Marketing Speak

- Open source storage architecture for clusters with 1000s of clients
  - Open source GPLv2
  - Freely distributed and in many cases bundled and resold as appliances, engineered solutions, etc.
- POSIX compliant parallel file system
- Highly scalable for both storage and bandwidth
  - 10 TB/sec
  - 1 trillion files
  - 512 PiB of space
- 70% - 75% of the top 100 systems run Lustre

Technical Mumblings

- Built on ldiskfs (ext4 variant) or zfs file systems
- Heterogeneous networking (Infiniband and Intel Ominpath®)
- High-availability
- Capacity growth
- User, group and project quotas
- Policy engine (Robinhood)
- Controlled file layout

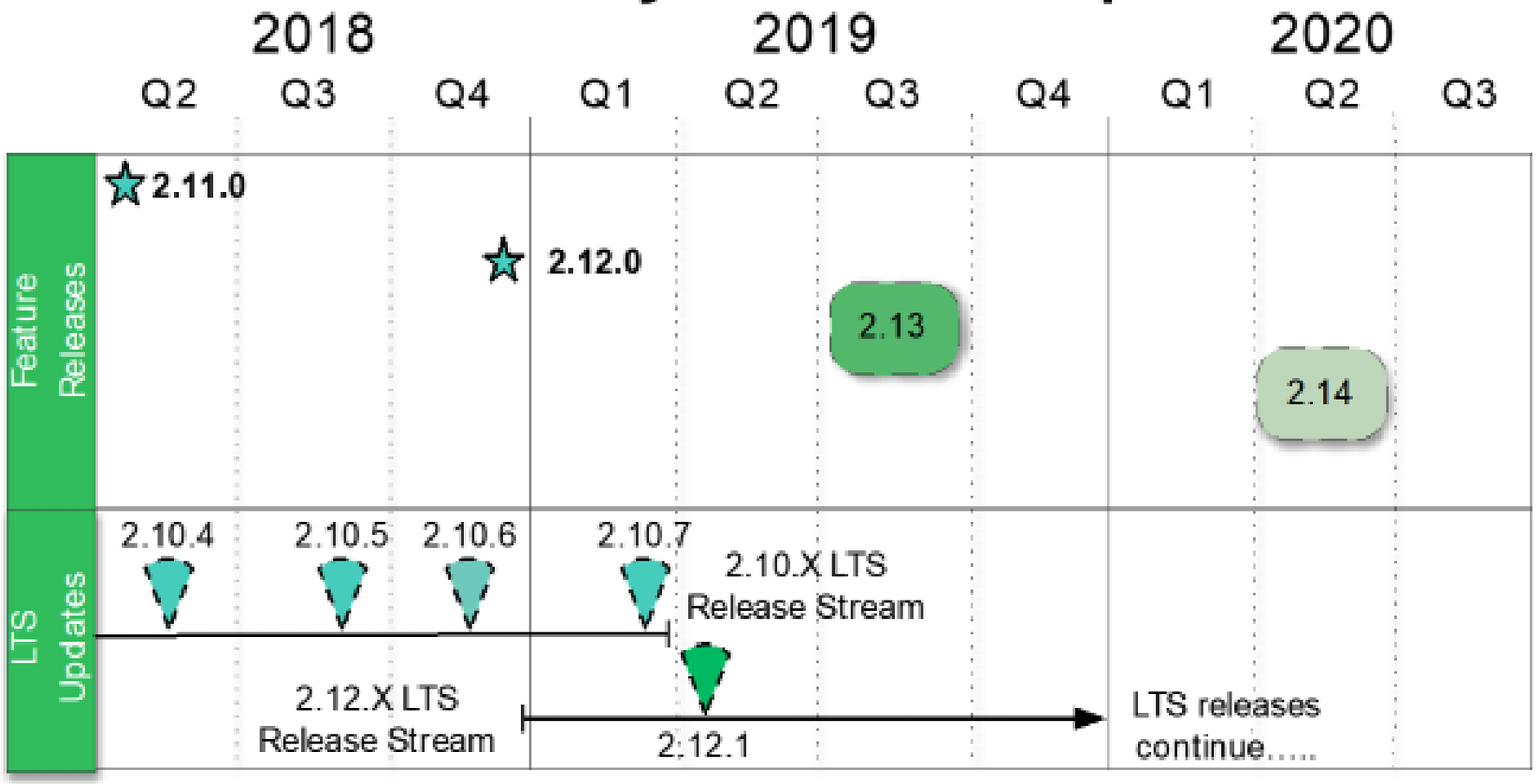
Yesterday –
A very brief history of the Lustre “Hot Potato”

- 2003 – Installed at LLNL for production
- 2004 – Cluster File System (CFS) is founded (v1.2)
- 2007 – Sun acquired CFS (v1.6)
- 2010 – OpenSFS founded
- 2010 – Oracle acquired Sun (v2.0)
- 2011 – Whamcloud formed with former Sun/Oracle folks (v2.1)
- 2011 – Xyratex Lustre team formed
- 2012 – Intel acquired Whamcloud (v2.5)
- 2013 – Xyratex acquired Lustre assets from Oracle
- 2014 – Seagate bought Xyratex
- 2017 – Cray acquired “Xyratex” from Seagate
- 2018 – Intel sold off the Lustre team and assets to DDN, who created a new division – Whamcloud
- Has anything happened this morning?
- Pretty nice history on the Wikipedia.org site

Today – Current status

- Lustre 2.10.7 released – long term stability version
- Lustre 2.12.1 released – will be the long term stability version
- Lustre 2.13.0 development – scheduled release Q3 2019

Lustre Community Roadmap



LEGEND:

Expected Timeline | Timeline TBD | Completed | LTS Branch

2.11
- Data on MDT
- FLR Delayed Resync
- Lock Ahead

2.12
- Lazy Size on MDT
- LNet Health
- DNE Dir Restriping

2.13
- Persistent Client Cache
- LNet Selection Policy
- Self Extending Layouts

2.14
- FLR Erasure Coding
- Health Monitoring
- DNE Auto Restriping

* Estimates are not commitments and are provided for informational purposes only
* Fuller details of features in development are available at http://wiki.lustre.org/Projects

Architecture

- Architectural Strategy
- Building Blocks
  - Overview
  - MGS/MGT
  - MDS/MDT
  - OSS/OST
  - Lustre Clients
  - LNET Routers
  - Robinhood
- Networking
  - Infiniband
  - Ethernet
  - Other choices

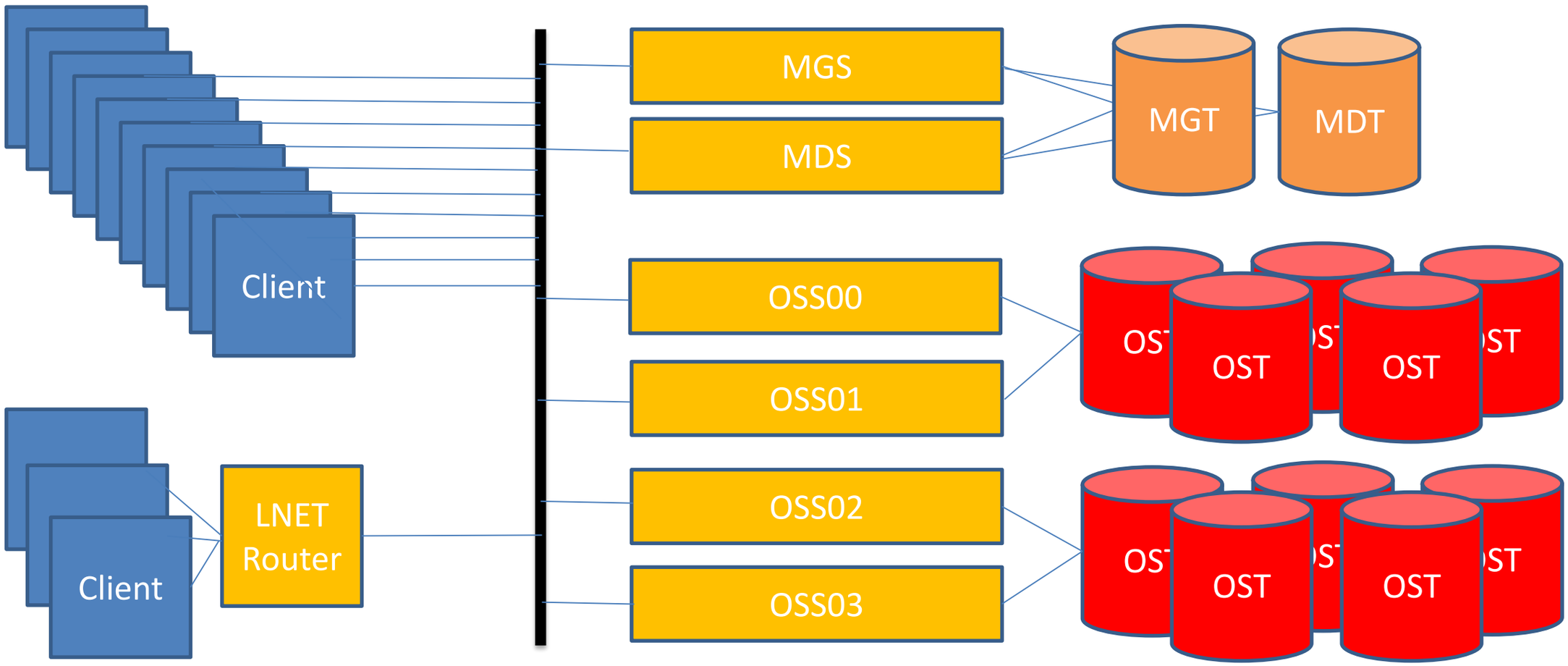
Architectural Strategy

- Client server model
- Pairwise failover
- Separate Metadata & Data Servers
- Works with commodity hardware
- Kernel based
- Linux only

Overview

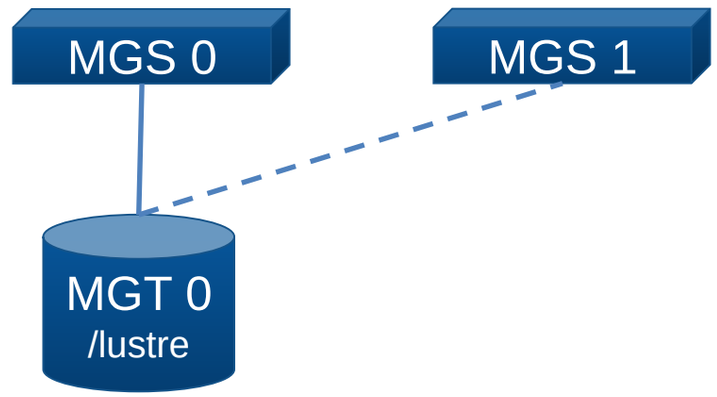
- Five Basic Components
  - **Management Server (MGS)** - Handle cluster configuration
  - **Metadata Servers (MDS)** - Manages file naming / directories attributes in the file system
  - **Object Storage Server (OSS)** - Provides file I/O services
  - **File system clients** - Usually compute nodes running Linux
  - **Lustre Network abstraction (LNET)** - Handle the network interface implementation for high performance access and communication

Overview (cont.)



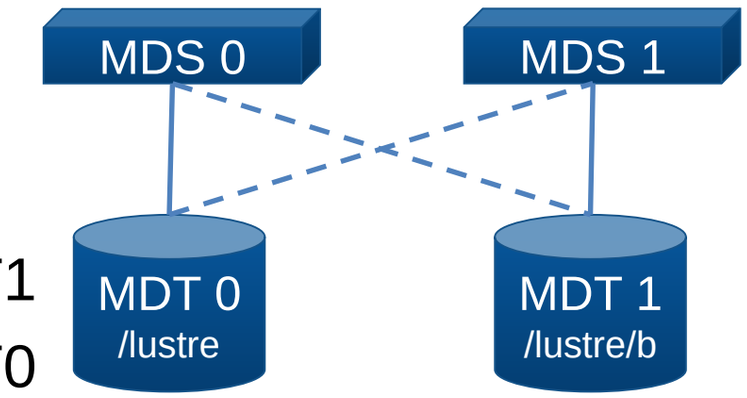
Management Server (MGS)

- MGS
  - Stores information on a MGT
  - Manages cluster configuration database.
  - Can be co-located or separate from MDS
  - Typically an active/passive configuration



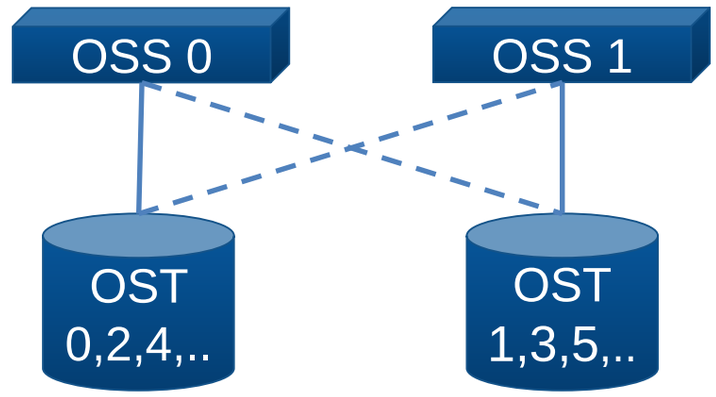
Metadata Servers (MDS)

- MDS
  - Provides the namespace services and stores the metadata information on MDT(s)
  - Distributed Namespace (DNE)
    - Multiple MDTs supported by one or more MDSs
    - Increased performance and increased metadata capacity
    - The root directory is always located on MDT0.
    - For the file system /lustre with 2 directories, a & b
      - Files in /lustre/a would reside in MDT0
      - Files in /lustre/b would reside in MDT1
  - High Availability
    - MDS0 is active for MDT0 and passive for MDT1
    - MDS1 is active for MDT1 and passive for MDT0



Object Storage Server (OSS)

- OSS
  - Provides file Read/Write services and store data in one or more Object Storage Target (OST)
  - Configured in pairs with shared storage
  - Typically 1 – 8 OSTs per OSS system
    - Based off system performance and requirements
    - OSTs can only be mounted on one server at a time



File System Clients

- POSIX compliant file system access layer
- Talks to:
  - MDS for metadata services
  - OSS for data
- User access
- Manages the file striping across multiples OSTs
- System function: Robinhood, HSM copytools
- Gateways to NFS, CIFS, …

Lustre Network (LNET)

- LNET
  - Lustre uses its own network abstraction layer called LNET for communicating with all nodes within a cluster
  - Networking layer that provides the Lustre Network Drivers (LND) for heterogeneous networks
  - Provides unique network ID (NID) for every interface of supported network type on every node in Lustre cluster
  - Properly configured LNET can give more than 80% bandwidth of raw network
  - LNET has to be specifically told about which interface to use
  - LNET itself does not do interface bonding, but it works well with bonded TCP/IP interfaces (Ethernet)
  - Multi-Rail

Lustre Network (LNET)

- LNET can be controlled by:
  - Specifying module parameters in /etc/modprobe.d/lustre.conf
  - Modules parameters decide which interface to use, NIDs and routing
  - Using “lnetctl” command line interface
    - “lnetctl” allows configuring, starting, stopping, testing LNET

Robinhood

- Software for the monitoring, audit and purging of large POSIX file systems, enables in particular:
  - Generation of detailed reports on the usage profile of a file system
  - Raising of alerts when the objects of the file system fulfil certain given conditions
  - Application of complex purge policies defined by the administrator
  - Careful control and balancing of the occupation rate of discs in a Lustre file system
- Resides on a separate Lustre client capable of running a large MariaDB supporting the collection of data about the file system and it’s contents
- Provides the policy engine functionality for Lustre HSM and being Lustre aware:
  - Process Lustre changelogs to avoid an expensive filesystem scan
  - It can perform list/purge operations on files per OST
  - Aware of OST artifacts like OST usage
- Multi-threaded architecture, developed for use in HPC

Break

- Renew
- Refresh
- Restore
- Reacquaint
- Relieve

- But, be back in 15 minutes!

Installation and Operations

- Lustre Release, OS and Networking Requirements
- Installation Process (not the instructions)
- Operations
  - lctl
  - Startup
  - Mounting
  - Shutdown
  - Logfiles
  - Quotas
  - Lustre HSM
  - Robinhood
- Performance Testing

Lustre Release, OS and Networking Requirements

- Lustre Server Release
  - Supported OS and release levels
  - Can be built for others, within limits
  - Special networking requirements, Mellanox OFED levels
- Lustre Client Release
  - Released with server
  - Can be built for more OS and Networking releases
  - Usage not tied to the server level, but keep it close or you can lose functionality

Installation Process (not the instructions)

- Needs Assessment
  - Big or little files?
  - 1Ks or 1Ms or 1Bs of files?
  - Hot or Cold storage?
  - Access patterns
  - Defensive I/O
  - Sizing: Speed vs Space vs Cost (you only get to pick two)

Installation Process (not the instructions)

- Hardware/OS Install
  - Installation: Rack it, Stack it, and Plug it all in
  - Firmware
  - Load and Configure the Operating System
    - Patches
    - Drivers
    - Define storage: ldiskfs or ZFS
    - Network connections
    - “If it is not broke, don’t fix it” has come to an end
      - Must install security patches
      - Must move to current release
- HW and Network validation
  - Drive/RAID groups
  - Network connections

Installation Process (not the instructions)

- Lustre software download
  - Whamcloud
  - OpenSFS
  - EOFS
  - Vendors
- Release build or build your own?
  - Servers
  - Clients

Installation Process (not the instructions)

- Lustre Software
  - Networking
    - Define the network
  - Lustre Servers
    - Load Lustre server software
    - Configure
      - Lnet
        - » Load Lustre and Lnet modules
      - MGS(s)
        - » MGT
      - MDS(s)
        - » MDT(s)
      - OSS(s)
        - » OST(s)
  - Lustre Clients
    - Load Lustre client software
    - Configure lnet and load modules
    - Mount the Lustre file system
- Benchmark the filesystem for a performance baseline

Operations

- lctl
- Startup Lustre
- Mount the filesystem
- Shutdown Lustre
- Logfiles
- Quotas
- Robinhood

Operations - lctl

- Administrative CLI to Lustre
- lctl is your administrative friend and “lctl help” gets you started
  - Controls
    - Network (also lnetctl)
    - Devices
    - Debugging Control
    - Change Logs
    - Testing
    - Pools
  - User probably will use just lfs

Operations – lctl dl

```
mds # lctl dl
0 UP mgs MGS MGS 9
1 UP mgc MGC192.168.0.10@tcp e384bb0e-680b-##### 5
2 UP mdt MDS MDS_uuid 3
3 UP lov lustre-mdtlov lustre-mdtlov_UUID 4
4 UP mds lustre-MDT0000 lustre-MDT0000_UUID 5
5 UP osc lustre-OST0000-osc lustre-mdtlov_UUID 5
6 UP osc lustre-OST0001-osc lustre-mdtlov_UUID 5
7 UP osc lustre-OST0002-osc lustre-mdtlov_UUID 5
8 UP osc lustre-OST0003-osc lustre-mdtlov_UUID 5
9 UP osc lustre-OST0004-osc lustre-mdtlov_UUID 5
10 UP osc lustre-OST0005-osc lustre-mdtlov_UUID 5
```

Operations - Startup

- Load any network and storage modules
- Load Lustre and LNET modules
- Start MGS, mount mgt
- Start MDS(s), mount mdt(s)
- Start OSS(s), mount ost(s)
- Start Robinhood & HSM copytools if required
- Mount the file system on clients

Operations - Mounting

- Automatic or Manual is operational decision
- Make sure the network is up and modules loaded
- /etc/fstab entry

Operations - Shutdown

- Unmount the Lustre filesystem on clients
  - Stop Robinhood and HSM copytool servers and unmount Lustre
- Unmount the MGT + MDTs
- Unmount the OSTs
- Unload the Lustre and LNET modules
- On zfs systems, shutdown zpools
- Shutdown system

Operations – Stats and Logs

- /var/log/messages – on all mgs/mds/oss/routers/client systems
- Lustre Debug Log – circular buffer
- http://wiki.lustre.org/Diagnostic_and_Debugging_Tools
- Traditional system/network utilities for tracing
- http://wiki.lustre.org/Lustre_Monitoring_and_Statistics_Guide
- Robinhood
- Importance of a periodic benchmark

Operations - Quotas

- User, Group and Project quotas
- Administration via lfs and lctl commands
- No single point of administration
  - Commands executed on MGS, MDS, OSS and clients
- Accuracy
  - Megabyte resolution
  - File system needs to be quiet.
- Hard/Soft – Block and/or Inode limit
  - Soft limit with a grace timer
  - Block quota consumed by OST
  - Inode quota consumed by MDT

Operations – Lustre HSM

- Launch the copytool on each agent node to connect to your HSM storage
- If your HSM storage has POSIX access this command will be of the form:

  lhsmtool_posix --daemon --hsm-root $HSMPATH --archive=1 $LUSTREPATH

Operations - Robinhood

- Start/Stop/Status using systemctl
- Configuration file
- Log files
  - Robinhood
  - MySQL/MariaDB
- MySQL or MariaDB is your friend, learn how it works
  - Administration tools, MySQL workbench
  - Befriend a DBA
- Database Maintenance
  - Backups
  - Performance tasks
    - ANALYZE TABLE
    - OPTIMIZE TABLE
    - Long running queries

Performance Testing

- System level tests:
  - http://wiki.lustre.org/Testing_HOWTO
  - http://wiki.lustre.org/LNET_Selftest
- User level tests
  - ior: MPI based test of IO performance
    - http://wiki.lustre.org/IOR
  - mdtest: MPI based test of metadata performance
    - http://wiki.lustre.org/MDTest

Performance Testing - ior

```
IOR-3.1.0: MPI Coordinated Test of Parallel I/O

Began: Mon Apr  8 08:39:08 2019
Command line used: /home/mptimme/paper/ior -a POSIX -i 3 -d 30 -b 512g -t 1g -F -o
1/r@2/r@3/r@4/r@5/r@6/r@7/r@8/r@9/r@10/r@11/r@12/r@13/r@14/r@15/r@16/r@17/r@18/r@19/r@20/r
Machine: Linux ast1197

Test 0 started: Mon Apr  8 08:39:08 2019
Summary:
             api                = POSIX
             test filename      = 1/r@2/r@3/r@4/r@5/r@6/r@7/r@8/r@9/r@10/r@11/r@12/r@13/r@14/r@15/r@16/r@17/r@18/r@19/r@20/r
             access             = file-per-process
             ordering in a file = sequential offsets
             ordering inter file= no tasks offsets
             clients            = 20 (1 per node)
             repetitions        = 3
             xfersize           = 1 GiB
             blocksize          = 512 GiB
             aggregate filesize = 10240 GiB

access    bw(MiB/s)  block(KiB) xfer(KiB)  open(s)    wr/rd(s)   close(s)   total(s)   iter
------    ---------  ---------- ---------  --------   --------   --------   --------   ----
write     33812      536870912  1048576    130.01     310.11     180.11     310.12     0
read      34324      536870912  1048576    136.62     305.49     168.87     305.49     0
remove    -          -          -          -          -          -          1.34       0
write     33499      536870912  1048576    132.98     313.02     180.04     313.02     1
read      35634      536870912  1048576    138.26     294.26     156.00     294.26     1
remove    -          -          -          -          -          -          1.36       1
write     58013      536870912  1048576    0.000917   180.75     52.30      180.75     2
read      67871      536870912  1048576    0.000607   154.49     28.56      154.49     2
remove    -          -          -          -          -          -          1.35       2

Max Write: 58012.73 MiB/sec (60830.76 MB/sec)
Max Read:  67871.25 MiB/sec (71168.16 MB/sec)

Summary of all tests:
Operation   Max(MiB)   Min(MiB)  Mean(MiB)     StdDev    Mean(s) Test# #Tasks tPN reps fPP reord reordoff reordrand seed segcnt blksiz xsize aggsize API
RefNum
write       58012.73   33498.62   41774.61   11482.80  267.96166 0 20 1 3 1 0 1 0 0 1 549755813888 1073741824 10995116277760 POSIX 0
read        67871.25   34324.17   45943.16   15514.72  251.41637 0 20 1 3 1 0 1 0 0 1 549755813888 1073741824 10995116277760 POSIX 0

Finished: Mon Apr  8 09:08:11 2019
```

Performance Testing - mdtest

```
mdtest-1.9.3 was launched with 40 total task(s) on 40 node(s)
Command line used: /ascldap/users/mptimme/mdtest-master/mdtest -z 2 -b 10 -n 10000 -w 0 -i 3
Path: /lustre/mptimme
FS: 376.2 TiB   Used FS: 0.7%   Inodes: 156.6 Mi   Used Inodes: 0.3%

40 tasks, 399600 files/directories

SUMMARY: (of 3 iterations)
   Operation                     Max            Min           Mean        Std Dev
   ---------                     ---            ---           ----        -------
   Directory creation:      17120.214      16163.383      16521.911        425.833
   Directory stat    :     165620.260     137121.477     154387.566      12392.109
   Directory removal :      28894.752      26902.895      27696.952        861.831
   File creation     :      34566.021      33050.373      33596.068        687.649
   File stat         :     111148.990     101212.401     107006.496       4221.357
   File read         :      77872.801      76639.431      77140.922        529.219
   File removal      :      25290.073      21608.546      23050.123       1605.503
   Tree creation     :       4769.439       3739.982       4416.127        478.269
   Tree removal      :       1051.721        955.792        991.210         42.995

-- finished at 04/23/2019 07:28:46 --
```

Users and Applications

- lfs – User view to the filesystem
- POSIX Parallel File System and How to Use It
  - OSTs
  - Pools
  - Stripes
- Robinhood
- Lustre HSM
- REST, HADOOP and other interfaces?

lfs

- User view and CLI interface to the Lustre file system
- lfs help or man lfs to get started
- Controls or user views into:
  - Striping
  - Quotas
  - Filesystem tools: mkdir, find, df, cp, ls, mv
  - ACLs
  - HSM

POSIX Parallel File System and How to Use It

- Goal is to use all of the disks at the same time
  - Evenly balanced across all OSS servers and OSTs
  - Balanced across all network interfaces
  - What is the best blocking and transfer sizes
- OSTs
  - Underlying storage: SSDs or Spinning Rust, mirrors, RAID,…
- Pools
  - Dedicated disks or grouping of disks with specific characteristics
- Stripes

Robinhood

- Externally maintained database of metadata allowing for policy based administration of a POSIX file system
- Gathers data by:
  - Scanning the file system
  - Reading the Lustre change log – Lustre aware
- Robinhood commands
  - User oriented
    - rbh-find – find clone that queries the robinhood DB
    - rbh-du – du clone that queries the robinhood DB
  - Administration oriented
    - rbh-report – querying command for robinhood policy engine
    - rbh-diff – lists differences between the robinhood database and the filesystem

Lustre HSM

- Lustre is not a full featured HSM
  - It provides the hooks and utilities to interface to another full featured HSM system and give Lustre increased HSM functionality
  - Can attach to multiple HSM systems
- Lustre HSM commands
  - lfs hsm_archive /lustre/XYZ
  - lfs hsm_release /lustre/XYZ
  - lfs hsm_restore /lustre/XYZ
  - lfs hsm_remove /lustre/XYZ
  - lfs hsm_cancel /lustre/XYZ
  - lfs hsm_state /lustre/XYZ
  - lfs hsm_set --norelease /lustre/XYZ
  - lfs hsm_clear --noarchive /lustre/XYZ
    - Set/clear flags: NOARCHIVE, NORELEASE, DIRTY, LOST

Lustre HSM - Create Small Demo Files

```
[root@robinhood01 mtimmerman]# cp /bin/bash file_a
[root@robinhood01 mtimmerman]# cp /bin/bash file_b
[root@robinhood01 mtimmerman]# cp /bin/bash file_c
[root@robinhood01 mtimmerman]# cp /bin/bash file_d
[root@robinhood01 mtimmerman]# ls -l
total 942
-rwxr-xr-x 1 root root 960616 Mar 28 10:37 file_a
-rwxr-xr-x 1 root root 960616 Mar 28 10:37 file_b
-rwxr-xr-x 1 root root 960616 Mar 28 10:37 file_c
-rwxr-xr-x 1 root root 960616 Mar 28 10:37 file_d
```

Lustre HSM - Archive Demo Files

[root@robinhood01 mtimmerman]# lfs hsm_archive file_b file_c file_d

[root@robinhood01 mtimmerman]# lfs hsm_state file*

file_a: (0x00000000)

file_b: (0x00000009) exists archived, archive_id:1

file_c: (0x00000009) exists archived, archive_id:1

file_d: (0x00000009) exists archived, archive_id:1

Lustre HSM – Release Demo File

```
[root@robinhood01 mtimmerman]# lfs hsm_release file_c
[root@robinhood01 mtimmerman]# lfs hsm_state file*
file_a: (0x00000000)
file_b: (0x00000009) exists archived, archive_id:1
file_c: (0x0000000d) released exists archived, archive_id:1
file_d: (0x00000009) exists archived, archive_id:1
```

Lustre HSM – Restore Demo File

```
[root@robinhood01 mtimmerman]# lfs hsm_restore file_c
[root@robinhood01 mtimmerman]# lfs hsm_state file*
file_a: (0x00000000)
file_b: (0x00000009) exists archived, archive_id:1
file_c: (0x00000009) exists archived, archive_id:1
file_d: (0x00000009) exists archived, archive_id:1
```

Lustre HSM – Remove Demo File

```
[root@robinhood01 mtimmerman]# lfs hsm_remove file_d
[root@robinhood01 mtimmerman]# lfs hsm_state file*
file_a: (0x00000000)
file_b: (0x00000009) exists archived, archive_id:1
file_c: (0x00000009) exists archived, archive_id:1
file_d: (0x00000000), archive_id:1
```

References

- Official home of Lustre
  - http://lustre.org
- Whamcloud
  - http://whamcloud.com
- OpenSFS (Open Scalable File Systems)
  - http://opensfs.org
- EOFS (European Open File System)
  - http://www.eofs.eu
- HPC and Storage Vendor Websites
- Robinhood
  - https://sourceforge.net/projects/robinhood/

Community Resources

- Wiki
  - https://wiki.whamcloud.com/
  - http://wiki.lustre.org
  - https://en.wikipedia.org/wiki/Lustre_(file_system)
- News Groups
- Lustre Mailing Lists
  - lustre-discuss
  - Lustre slack/IRC
- Conferences – LUG, SC, ISC, LAD

Questions?

- Expert Fielders of Questions
  - Ruth Klundt – Sandia National Laboratories
  - Patrick Farrell – Whamcloud
  - Stephen Champion – Hewlett Packard Enterprise