ISC 2021 Digital BoF:
LUSTRE® **in HPC, BigData and AI:**
Status, New Features and Roadmap

Frank Baetke (EOFS / for HPE) / Hugo Falter (EOFS / Partec)
Kevin Harms (OpenSFS / ANL)
Sarah Neuwirth (Goethe-University Frankfurt)
Olaf Gellert (DKRZ)
Peter Jones (EOFS / Whamcloud-DDN)
Jacques-Charles Lafoucriere (CEA)

https://www.eofs.eu/

European Open File Systems - A Societas Europaea
Co-owner of the LUSTRE trademark, logo and assets

EOFS President:
- Frank Baetke (acting for HPE)

EOFS Vice-President:
- Jacques-Charles Lafoucriere (CEA)

Directors of the Administrative Council:
- Hugo R. Falter (ParTec AG)
- Peter Jones (DDN/Whamcloud)

Members of the Administrative Council:
- Eric Monchalin (Atos)
- Jacques-Charles Lafoucriere (CEA)
- Thomas Stibor (GSI)
- Frank Baetke (acting for HPE)
- Johann Lombardi (Intel)
- Arndt Bode (LRZ)

http://opensfs.org

What is OpenSFS?

- OpenSFS facilitates a community around Lustre
  - Organization for both Vendors (Participants) and Users (Members) to discuss features and directions
- Promote Lustre and the Lustre community
- Ensure Lustre remains vendor-neutral and open
- Organize the LUG conference

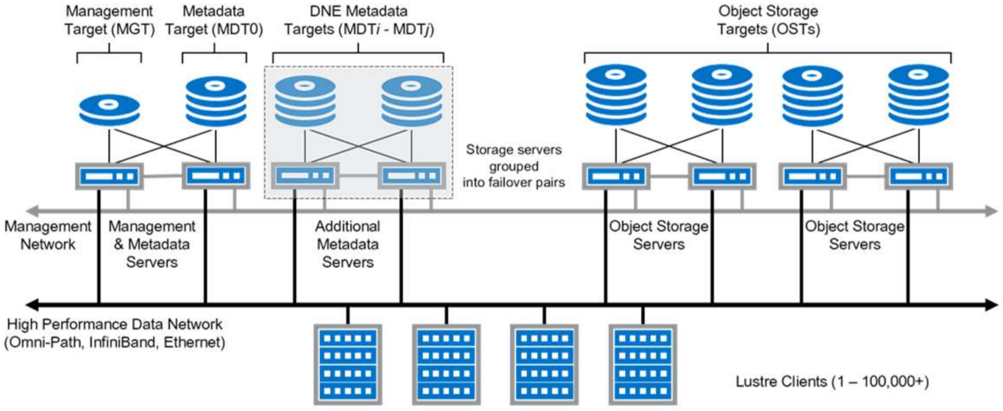
- **Co-owner of the LUSTRE trademark, logo and assets**

AEON COMPUTING
ddn
FUJITSU
Hewlett Packard Enterprise
intel
Microsoft
NetApp

Argonne NATIONAL LABORATORY
arm
BAE SYSTEMS INSPIRED WORK
bp
cea
ConocoPhillips
CSCS
Fermilab
INDIANA UNIVERSITY
iRODS
BERKELEY LAB
Lawrence Livermore National Laboratory
leidos
Los Alamos NATIONAL LABORATORY EST. 1943
NASA
NOAA
OAK RIDGE National Laboratory
SDSC SAN DIEGO SUPERCOMPUTER CENTER
Sandia National Laboratories
SIG SUSQUEHANNA
SRCC
TACC
UF UNIVERSITY of FLORIDA

Lustre File System – Architecture and Key Benefits

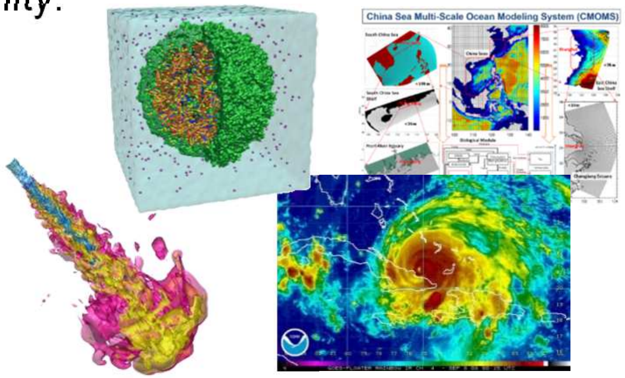
- Lustre is an open-source, global single-namespace, POSIX-complaint, distributed parallel file system
- Key design aspects:
  - Scalability – supports small-scale HPC environments to the very largest high-end supercomputers
  - High file I/O performance through flexible file striping for varying I/O patterns and sizes
  - High-availability – data is stored persistently and reliably, without loss or corruption of information
- Client-server network architecture
- Redundant servers support storage failover
- Capable of Exascale capacities
- Supports high-speed network fabrics
- Community participation:
  - EOFS: https://www.eofs.eu/
  - OpenSFS: https://www.opensfs.org/


Management Target (MGT)
Metadata Target (MDT0)
DNE Metadata Targets (MDT*i* - MDT*j*)
Object Storage Targets (OSTs)
Storage servers grouped into failover pairs
Management Network
Management & Metadata Servers
Additional Metadata Servers
Object Storage Servers
Object Storage Servers
High Performance Data Network (Omni-Path, InfiniBand, Ethernet)
Lustre Clients (1 – 100,000+)

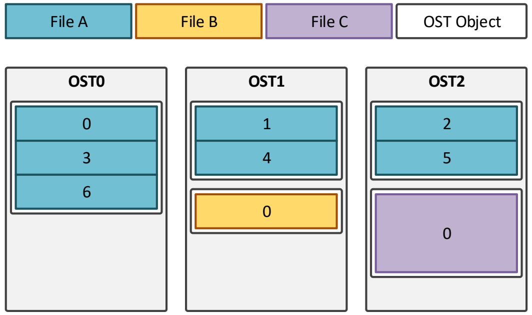
Architectural overview of Lustre building blocks.

Lustre File System – Architecture and Key Benefits

- Object-based storage building blocks to maximize scalability:
  - Metadata is stored separately from file object data.
    ⇒ Each file system can be optimized for different workloads
  - With Lustre DNE (Distributed Namespace), multiple metadata servers can be added to increase the namespace capacity and performance.
  - Additional OSSs can be added to increase capacity and throughput bandwidth.
    ⇒ Max. filesystem size: 512PB (LDISKFS), 8EB (ZFS)
- High I/O performance for data-intensive HPC applications:
  - Files are divided into stripes and stored across multiple OSTs.
  - Progressive File Layout (PFL) enables flexible file layouts for different parallel I/O patterns and sizes.
  - Low overhead for small files, increased bandwidth for large files.
  - A single file system instance can, in aggregate, present up to tens of petabytes of storage to thousands of compute clients, with more than a terabyte-per-second of combined throughput.



Data-intensive application support.



File A | File B | File C | OST Object

OST0: 0, 3, 6

OST1: 1, 4, 0

OST2: 2, 5, 0

File striping, RAID-0 pattern.

Current Lustre Filesystems - Mistral

- **Installed 2015 / 2016**
  - ClusterStor CS9000
    - *21 PB diskspace / 124 OST / 5 MDT / 6TB HDD*
  - ClusterStor L300
    - *33 PB diskspace / 148 OST / 7 MDT / 8TB HDD*
  - Infiniband FDR
- **Usage**
  - HOME / SCRATCH / WORK on one filesystem
  - Extension of WORK on second filesystem
    - *Approx. 300 projects on WORK*
  - Current usage ~84% on both filesystems
  - Approx. 1.200.000.000 inodes used

Next Lustre Filesystems - Levante

- **Installation Summer / Fall 2021**
  - DDN EXAScaler
    - *1x NVMe based filesystem for HOME (100 TB)*
    - *1x HDD based filesystem for WORK (120+ PB*
    - *1x NVMe / HDD mixed filesystem for testing (200 TB NVMe / 3.7 PB HDD) SCRATCH*
    - *Progressive File Layout*
    - *Infiniband HDR (100Gb)*
    - *Planned inode capacity 4.000.000.000*

- **Challenge**
  - Data-migration of about 44PB+ from previous Lustre filesystems to this system

Lustre 2.12.x LTS

Lustre 2.12.6 went GA on Dec 9th

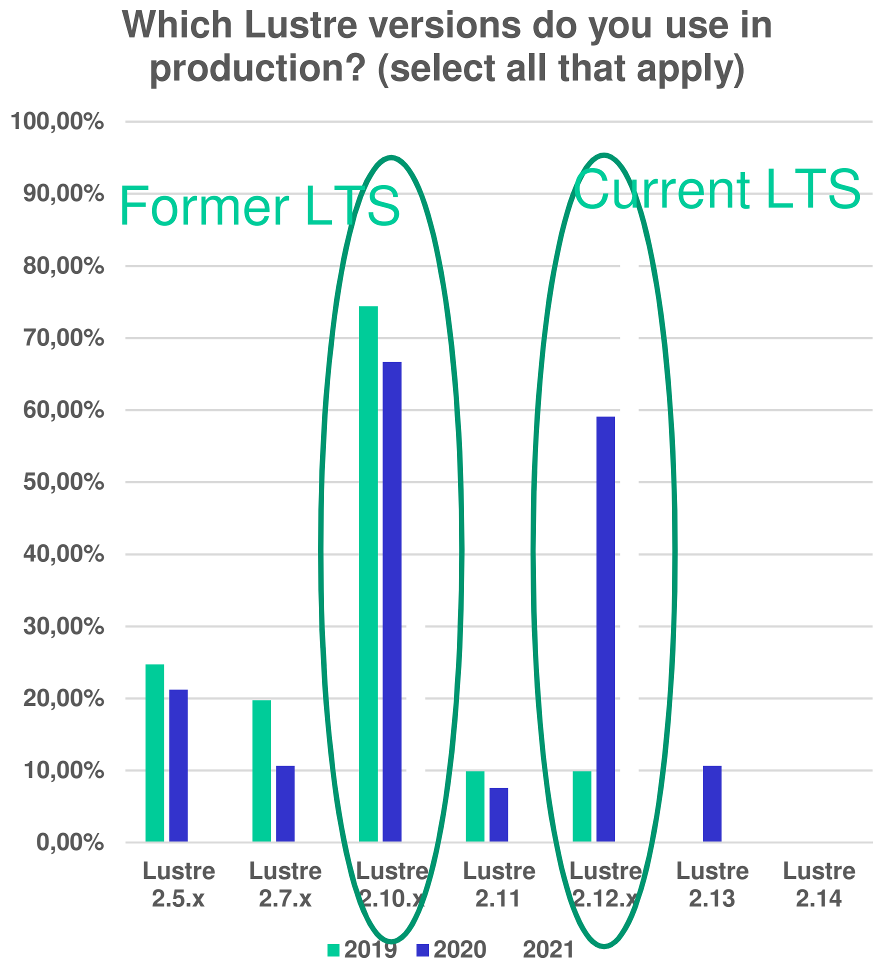
- RHEL/CentOS 7.9 servers/clients
- RHEL 8.3/SLES12 SP5/Ubuntu 20.04 clients
- http://wiki.lustre.org/Lustre_2.12.6_Changelog

Lustre 2.12.7 coming soon

- RHEL 8.4 client support
- Support for MOFED 5.x

Timing for the next LTS release being discussed

- Recent CentOS changes adding complexity
- Mini-survey for this BOF!
  - https://www.surveymonkey.com/r/CHMDGYT



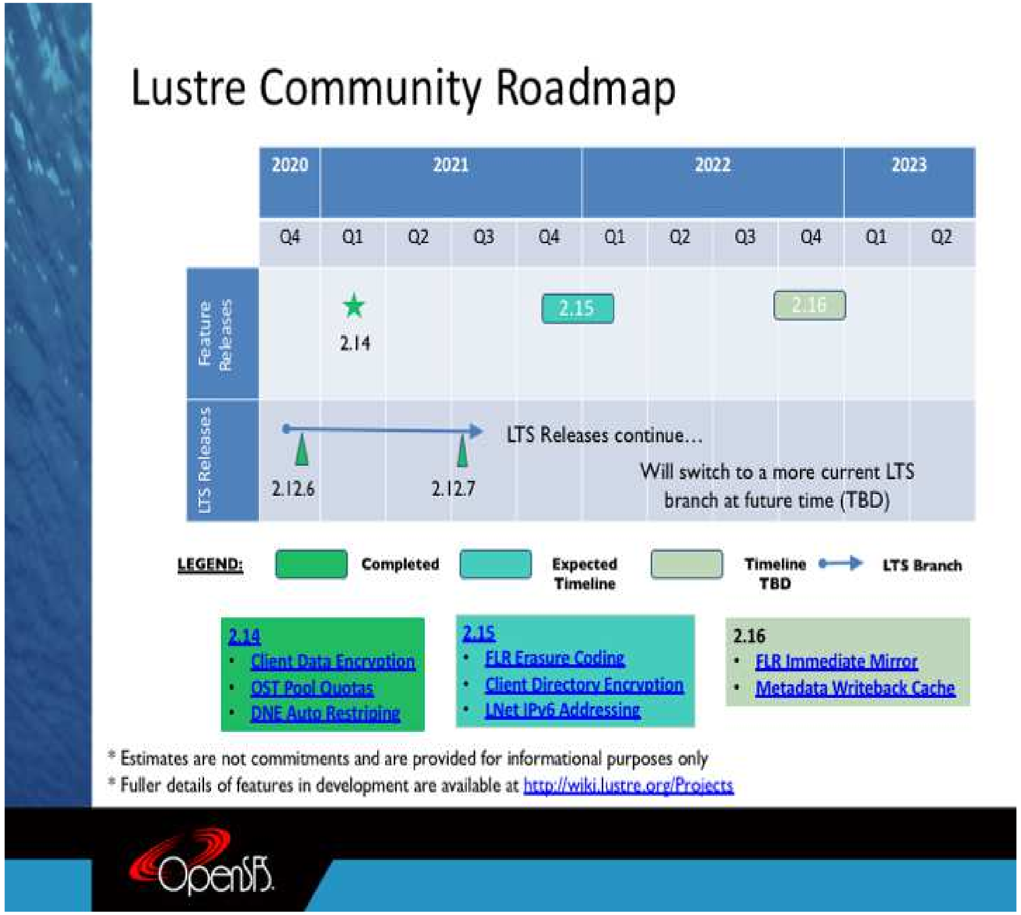
Lustre Major Releases

Lustre 2.14 went GA Feb 19th

- OS support
  - RHEL 8.3 servers/clients
  - RHEL 8.3/SLES15 SP2/Ubuntu 20.04 clients
- Number of useful features
  - Client-side Data Encryption
  - OST Pool Quotas
  - DNE Auto Restriping
- http://wiki.lustre.org/Release_2.14.0

Lustre 2.15 targeting Q4 release

- Client-side filename Encryption
- LNet IPv6 Addressing
- http://wiki.lustre.org/Release_2.15.0



Lustre Community Roadmap

LEGEND: 🟩 Completed 🟩 Expected Timeline 🟩 Timeline TBD → LTS Branch

2.14
- Client Data Encryption
- OST Pool Quotas
- DNE Auto Restriping

2.15
- FLR Erasure Coding
- Client Directory Encryption
- LNet IPv6 Addressing

2.16
- FLR Immediate Mirror
- Metadata Writeback Cache

* 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

Development Drivers

Multiple large Lustre deployments rolling out

- Lustre widely-used in HPC for many years
- New systems continuing to select Lustre (Fugaku, El Capitan, Orion, Perlmutter etc)

AI/ML workloads turning to Lustre

- Sélene system at NVIDIA

Cloud offerings expose Lustre to new markets

- All major CSPs have interest in Lustre

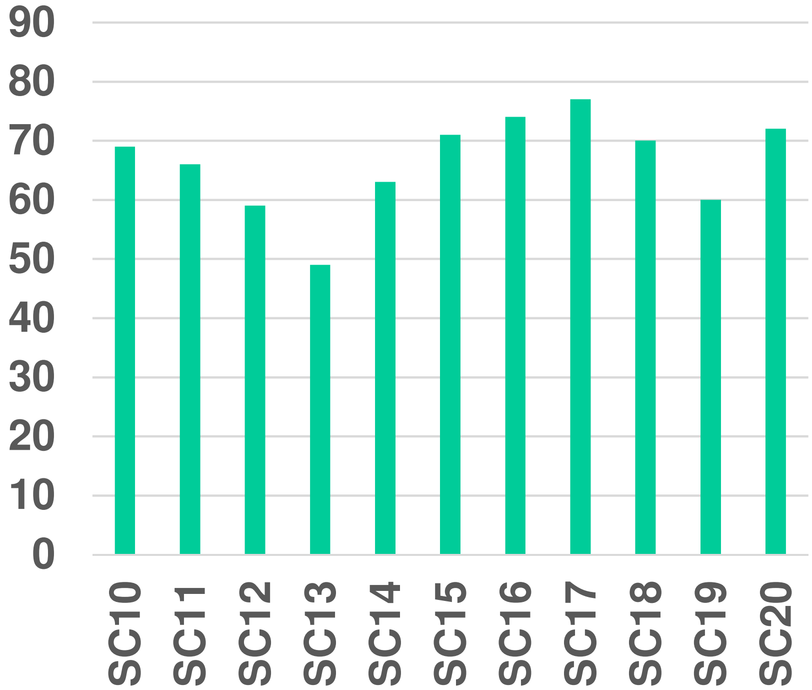
Interactions with kernel community

- Efforts to upstream Lustre client driven changes

See details in Andreas Dilger LUG presentation

- Slides and video available from OpenSFS LUG site

Historic % of Top 100 confirmed using Lustre



Data analysis of top500.org lists

How to get engaged with Lustre developments

Different types of development based on Lustre:

- **Correct a bug**
- **Add a feature**
- **Use Lustre for research work**

Lustre development environment is nice/powerful for developers

- **Very few development tools requirement**
- **Easy to debug**
- **All development can be done in simple virtual machines**
  - A cluster can run on a laptop
- **Lustre community offers a powerful validation test platform**
  - Complete local development platform
- **No need for a large cluster@home**
  - Anyone who want to contribute can easily do a development

Feedback on Lustre « patches »

- **Bug corrections**
  - The simplest way
  - Doing the right/accepted solution is difficult without Lustre experts involvement
    - Initial home made solution is generally not the final one

- **New feature**
  - The hardest way
  - Need community involvement for design and acceptation
    - Design/Development must be done with a close relationship with Lustre experts/community
  - Need a strong commitment from developer to reach Lustre release schedule

- **Lustre for academic research**
  - Lustre is a powerful platform
    - Easy to generates/tests new code
    - But not enough documentation on internals
      - Initial investment is too long for a small development
  - Not really used today

Discussion!