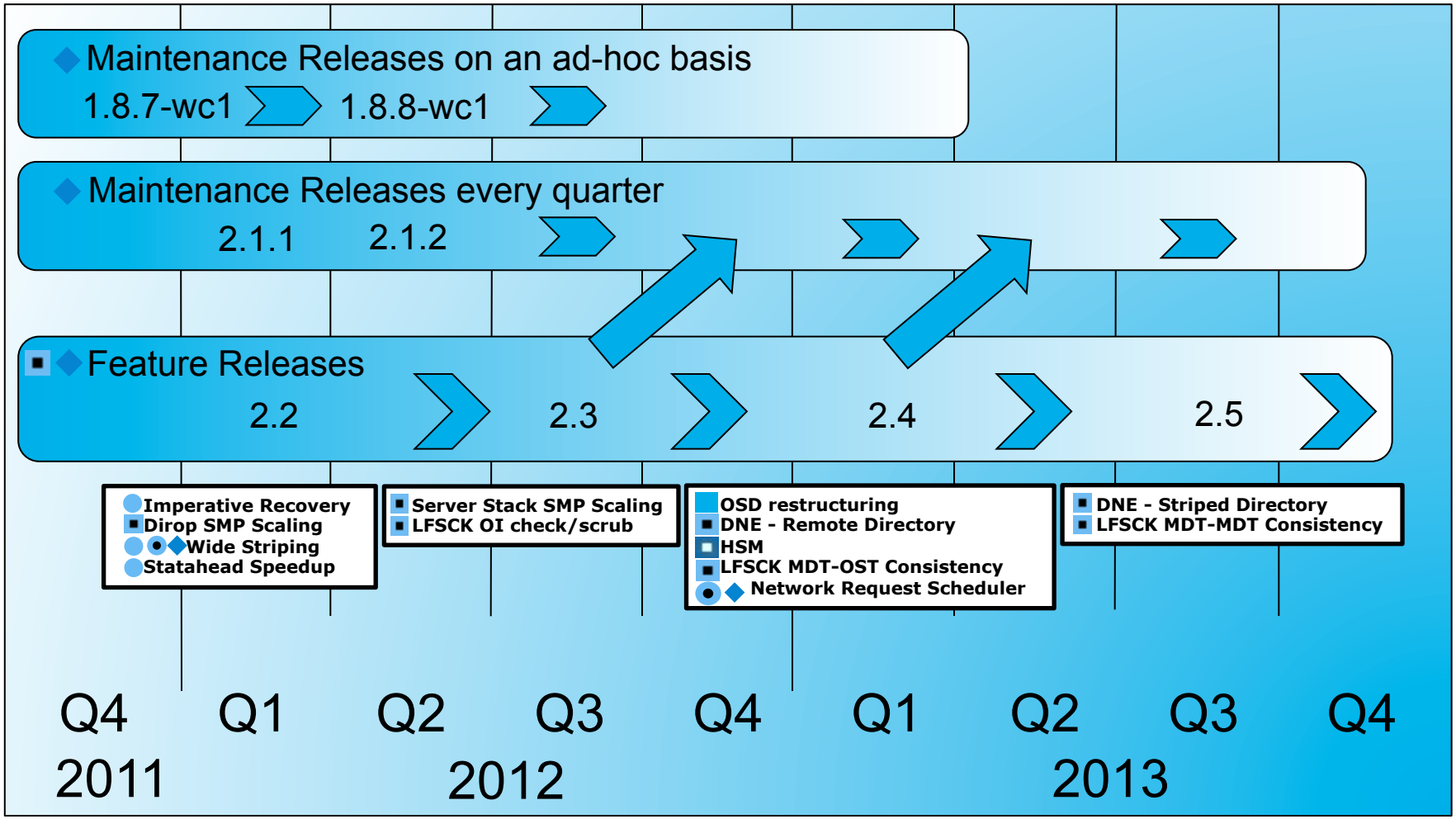
whamcloud

Lustre User Group
Austin, TX
April 2012

Lustre Future Development

- Andreas Dilger
  Principal Lustre Engineer
  Whamcloud, Inc.
  adilger@whamcloud.com

Community Lustre Roadmap



Sponsor for Whamcloud Development and Releases: ORNL OpenSFS LLNL Whamcloud

Third Party Development: CEA Xyratex

Lustre 2.3 - September 2012

- Client code cleanup (EMC/Cray)
  - Finish 2.6.38 dcache scalability changes
  - Port started for 3.1, 3.3 upstream kernels
  - Clean up old kernel portability code, isolate from server code
  - Groundwork for landing Lustre client to upstream kernel
- Single Server Metadata Performance (WC/OSFS)
  - LNET, ptlrpc locking improvements
  - Improved ptlrpc service thread scheduling (MRU)
  - NUMA affinity for ptlrpcd threads (checksums)
- LFSCK Online scrub - OI Rebuild (WC/OSFS)
  - OSD object iterator to efficiently traverse in-use inodes
  - Object Index (OI) verify/rebuild (corruption, or MDT backup/restore)
  - Distributed consistency checking infrastructure

Lustre 2.3 - continued

- Job Scheduler ID statistics tracking (WC)
  - Extract Job ID from client environment
  - Scheduler-dependent, tunable environment variable name
  - Send Job ID with every RPC to server
  - Track per statistics, like per-client stats
- OSD Restructuring - Object Filter Device (WC/LLNL)
  - OST/obdfilter code layered on top of OSD API
  - Will allow OST to benefit from Parallel Directory Operations
  - Allow running OST on top of ZFS
  - Presentation on this later today
- Client Checksum improvements (Xyratex)
  - Kernel assembly/hardware assistance
  - Performance-based algorithm selection
  - Presentation on this later today
- Many other projects underway
  - Not scheduled for release/landing until they are ready

http://wiki.whamcloud.com/display/PUB/Lustre+Community+Development+in+Progress

Lustre 2.4 - March 2013

- OSD Restructuring - LOD/OSP (WC/LLNL)
  - Remove VFS usage from MDS/MGS
  - Allow running MDT, MGT on ZFS
- LFSCK Online scrub - MDT-OST consistency (WC/OSFS)
  - Verify MDT inode to OST object references, or recreate missing objects
  - Verify each OST object is referenced by MDT inode, or move to lost+found
  - Verify no OST object referenced by two inodes, or copy/unlink/quarantine
- Distributed Namespace - Remote Directory (WC/OSFS)
  - Split namespace for subdirectory trees (e.g. /home/{user})
  - Scale metadata size/performance beyond single MDS
  - Presentation on this later today
- Network Request Scheduler (WC/Xyratex)
  - Reorder read/write operations for more optimal/repeatable disk order
  - Infrastructure for request prioritization (QOS, throttling, etc.)
  - Presentation on this later today
- 4MB Bulk RPC transfers (Xyratex)
  - Submit larger IO requests to disk
  - Less round-trip latency for WAN usage

Lustre 2.4 - ZFS OSD (WC/LLNL)

- Leverage many features immediately
  - Robust code with 10+ years maturity
  - Data checksums on disk + Lustre checksums on network
  - Online filesystem check/scrub/repair - no more *e2fsck*!
  - Scales beyond current ldiskfs object/filesystem limits
  - Drive commodity JBOD storage without RAID hardware
  - Integrated with flash storage cache (L2ARC)
- More ZFS features to leverage in the future
  - Snapshots, end-to-end data integrity, datasets
  - Active open development community (Delphix, Joyent, Illumos)
- Build Lustre servers without kernel patches
- Compatible with 1.8 clients
  - Minor issues fixed, and/or handled on server

Lustre 2.4 - HSM (WC/CEA)

- Originally developed by CEA France
- Simple archive back-end interface
  - Copy a file to archive, notify MDS it is finished
  - Initially supports HPSS and POSIX APIs
    - POSIX copytool can archive to any “filesystem”
    - HPSS copytool available to HPSS users
- Infrastructure useful for other projects
  - Layout lock for dynamic inode layouts
    - Data migration between storage pools/tiers
    - Asynchronous data mirroring
  - Policy engine to to provide automation interface
    - Integrate with Lustre ChangeLog to avoid scanning
    - Manage space, tiers, users, directory trees, file types

Lustre 2.5 - September 2013

- Distributed Namespace - Shard/Stripe dir (WC/OSFS)
  - Split a single large directory over multiple MDTs
  - Better size/performance for single directory
- LFSCK Online scrub - MDT-MDT consistency (WC/OSFS)
  - Verify parent->child remote directory
  - Verify master->slave directory shards

- OpenSFS TWG prioritizing other features
  - Requirements gathered from OpenSFS members, community
  - Developing consensus on short list of features
  - OpenSFS members vote to select priority feature funding
  - Meeting this Wednesday afternoon
  - One more good reason to join OpenSFS

Lustre 2.5+ - Storage Management

- Asynchronous Object Mirroring/Migration
  - File mirroring is critical to long-term availability
  - Selective mirroring of objects, within/across OST pools
  - Migration to balance OST usage, empty old OSTs
- Storage Tiers
  - Extension of existing OST pools
  - Different types of storage (flash, SAS, SATA) or locations
  - Add per-pool quotas to control usage/access permission
  - Leverage HSM infrastructure (copytool, policy engine)
- Complex File Layouts
  - Different layouts/pools for extents of the same file
  - Incrementally change striping as file grows
- Useful separately, powerful together

Lustre 2.5+ - Management/availability

- LNET configuration/robustness
  - Better tools, online routing configuration changes
  - LNET channel bonding, IPv6
- Scalable fault detection/Health Network
  - Reduce ping overhead (avoid clients*servers pings on idle system)
  - Faster notification of client/server failure = faster recovery/response
- Improved Lustre Management
  - Improved configuration tools, interfaces, and robustness
  - Better application programming interfaces
- Administrative shutdown of servers
  - Flush client caches/locks before server shutdown
  - Faster recovery of clients for controlled restarts
  - Transparent protocol changes possible
  - Simplified upgrade code = improved reliability

In every feature release

- Ongoing bug fixing
- Handling kernel API changes
- Performance improvements
- Other minor features
- Usability fixes

- Feature releases with funding from OpenSFS

Thank You

- Andreas Dilger
  Principal Lustre Engineer
  Whamcloud, Inc.
  adilger@whamcloud.com

ZFS on Linux Licensing Answers

- ZFS is NOT a derived work of Linux

  "It would be rather preposterous to call the Andrew FileSystem a 'derived work' of Linux, for example, so I think it's perfectly OK to have an AFS module, for example." – Linus Torvalds

  "Our view is that just using structure definitions, typedefs, constants, macros with simple bodies, etc., is NOT enough to make a derivative work. It would take a substantial amount of code (coming from inline functions or macros with substantial bodies) to do that." – Richard Stallman (The FSF's view)
- ZFS module is Open Source Software
  - Even proprietary binary modules tolerated (Nvidia, GPFS, etc)
- Companies already support OpenSolaris ZFS
  - Nexenta, Joyent, may other vendors via Illumos
  - CDDL provides ZFS patent indemnification