Lustre User Group 2009

Lustre 1.8 Features

Nathan Rutman
Sun Microsystems

Lustre 1.8 Goals

Focus: deliver a modest maintenance release

- Conservative new features
- Performance & scalability improvements
- Reliability/recovery improvements
- Forward interoperability with Lustre 2.0

Target scale: ORNL Jaguar Cluster

- 10.5PB storage; 240 GB/s I/O throughput goal
- 265,708 processor cores

1.8 Features List

- Adaptive Timeouts
- OSS Read Cache
- Version-Based Recovery
- OST Pools
- 2.0 Conversant (client interop)
- Performance Improvements

Adaptive Timeouts 3055

- Use an adaptive mechanism to set RPC timeouts.
- RPC service time histories are tracked on all servers, and are reported back to clients.
- Clients use this to set future RPC timeout values.
- Early replies prevent timeouts if estimate is incorrect

Adaptive Timeouts Benefits

- Relieves users from having to tune the obd_timeout value.
- Reduces RPC timeouts and disconnect/reconnect cycles.
- Enabler for speedier recovery
- Watchdog timers adapt also
- Scenarios:
  - Slowly changing server loading or network congestion
  - Sudden server / network loading

OSS Read Cache 12182

- Provides read-only caching of data on an OSS.
- Improves Lustre performance when several clients access the same data set, and the data fits the OSS cache
- Low overhead of OSS read cache. No performance impact due to cache misses

OSS Read Cache Benefits

- Allows OSTs to cache read data more frequently
- Improves repeated reads to match network speeds instead of disk speeds
- Provides the building block for OST write cache (small write aggregation).
- Scenarios
  - diskless clients booting from lustre
  - nodes sharing data (3d rendering)

OSS Read Cache Metrics

- Two clients accessing same file:
  - no cache: 77 MB/s
  - with cache: 390 MB/s
- Single client access lots of small files:
  - no cache: 148.6s
  - with cache: 35.7s

Version Based Recovery 10609

Current Recovery

- Requires all clients to replay transactions in original order
- If all clients don't reconnect during recovery window, recovery is aborted

Version Based Recovery

- Allows replay of independent transactions, even with missing clients
- Version conflicts will require client state to be reset
- Soon: delayed clients can reconnect after the recovery window and replay independent transactions

Version Based Recovery Benefits

- Improves the robustness of client recovery operations
- Not all clients are evicted if some miss recovery
- Allows Lustre recovery to work even if multiple clients fail at the same time as the server, if the remaining clients are working independently
- Provides a building block for disconnected client operations

OST Pools 15899

- Pools provide a method to specify an arbitrary group (instead of an index range) of OSTs for file striping purposes
  - Fast vs. slow disks
  - Local network vs. WAN
  - JBOD vs. RAID
  - Specific OSTs for users/groups/applications (by directory)
- Thanks to CEA

OST Pools Benefits

- Allows sets of OSTs to be selected via named groups
- Easier disk usage policy for administrators
- Hardware can be more closely optimized for particular usage patterns
- Pools can separate heterogeneous OSTs within the same filesystem
- Human-readable stripe mappings

```
lfs setstripe --pool scratch /mnt/lustre/workdir
```

Client Interoperability 11824,11930

- Enables Lustre 1.8 clients to work with the new network protocol that will be introduced in the 2.0 release.
- Transparent client, server, network and storage interoperability during migration from 1.6-based clusters to clusters with 2.0-based servers.
- When Lustre 2.0 is released, perform a 'live' upgrade from 1.8 to 2.0 without needing to shut down the system.

Client Interoperability Benefits

- Live upgrade path from 1.6 to 2.0 via 1.8
- Full mixed client / server interop between 1.6 and 1.8
- 1.8 clients work with 2.0 servers
- Shutdown notification
  - Server notifies clients of impending shutdown
  - Clients flush buffers and block ops, simplifying recovery

Performance Improvements

- Client-side SMP performance improvements 10706,11817
  - Decrease superblock contention
  - 5x write improvement on multi(8)-core servers (bonnie, iozone)
- Async Journal Commit on Write 16919
  - Reduces disk seeks in case of limited or disabled write-behind cache on block devices
  - 2-4x write improvement for sequential data streams from a small number of clients (vs 2x for external journal)

Performance Improvements -- coming soon

- LNET SMP Scaling 15379

  Add finer-grained locking into LNET to allow more parallelized ops

- Long-Haul (WAN tuning) 15983

Thank you

THANK YOU