LUSTRE ROADMAP UPDATE

April 28, 2008

Bryon Neitzel
Solution Architecture Manager, Lustre Group
Sun Microsystems, Inc.

Overview

- Identify **major features** in next 4 Lustre releases
  - 1.6.5, 1.8, 2.0, 3.0
  - Not meant to be a complete feature list
- Provide a **brief** description of those features
  - Several excellent presentations tomorrow about feature details.
- Show list of **future** features
  - These aren't yet scheduled into a specific release
  - In some cases, work is already underway

Lustre Release Taxonomy

- Historically, Lustre release numbers did not accurately reflect major changes to the product
- We're **transitioning** to more conventional taxonomy (x.y.z)
  - x: Major – architectural changes
  - y: Minor – new features
  - z: Maintenance – bug Fixes
- For example:
  - > Lustre 1.6.5 – bug fixes
  - > Lustre 1.8 – Version Based Recovery
  - > Lustre 2.0 – Userspace Servers

Code Bases

- HEAD
  - Leading (bleeding) edge of feature development
  - Several new, unreleased features are already in HEAD
    - Kerberos, CMD, etc
  - Not yet ready for prime time
- b1_6
  - This is the branch in our repository from which the 1.6 releases will be delivered
  - Made decision to also release 1.8 off this code branch
  - All 1.8 features will also be in HEAD

Some Major Requirements

- ORNL Baker/Spider system
  - 1 Pflop, 240GB/sec BW, router performance
- HPCS
  - 40K file creates/sec, 1 trillion files, data integrity
- NRL
  - gov level security, distributed file systems, capacity, windows client, WAN performance, search
- Lustre Summit
  - usability, recoverability,multi-clustered environments

Lustre 1.6.5

- Timeframe: imminent
- Defects
  - Can see all defects at **bugzilla.lustre.org**
  - Search on "**165-tracking**" (blockers on 13881)
- Adaptive Timeouts
  - **Major feature** in a bug release?
    - Turned off by default
  - Fully tested, **thanks to LLNL**
  - Production ready for those who choose to run this

Lustre 1.8

- Planned for late **summer 2008**
  - Based on **1.6 code base**, not HEAD
  - HEAD is usually the code branch for next release
- Adaptive Timeouts (AT)
  - **turned on by default**
- Version Based Recovery (VBR)
- Commit On Share (COS)
- OST Pools
- Interoperability Changes

Lustre 1.8

- Version Based Recovery (VBR)
  - Problem: all clients need to participate in recovery or all could be evicted
  - Allows clients that don't participate in initial recovery to **re-establish locks**
- Commit on Share (COS)
  - Problem: If client B depends on transactions completing by client A, and client A doesn't recover, client B also fails
  - When dependent transactions are detected, these are **committed quickly**, in order to avoid this situation during recovery

Lustre 1.8

- OST Pools
  - Lustre community effort led by **CEA**
  - A name associated with **a set of OSTs**
  - Will make object placement definitions **more flexible**
    - A directory or file can be restricting to striping within a pool
    - Pools can be assigned to specific clusters by client Network ID
    - Pools can be assigned to specific users or groups by UID/GID
    - Pools can be assigned to specific types of files by filename (e.g. *.mpg) to allow different striping for some files without requiring a default EA for all files in the output directory
  - **Independent** of the ZFS pools implementation

Lustre 1.8

- Adaptive Timeouts
  - **Modify RPC timeouts** based on server load
  - Timeouts **increase** as server load increases, **decrease** as server load decreases
  - Timeouts are **based on node health** rather than fixed duration
- Interoperability Changes
  - Will add support for **2.0 features** like new networking protocol, and fids
  - Allows 1.8 clients to **talk to 2.0 servers**, and 2.0 server to talk to 1.8 clients
  - http://arch.lustre.org/index.php?title=Interoperability_fids_zfs

Lustre 2.0

- Planned for **Dec 2008**
  - Based on HEAD branch
  - CMD code will be in this code base, **but disabled**
- Major new version of Lustre that introduces **substantial architectural changes** and features
  - Userspace Servers, ZFS, Solaris, Security, Replication, Server logs, HSM (Hierarchical Storage Management)
- Migration features may not available in 2.0
  - New ZFS deployments will be supported but existing deployments will have to wait for migration tools
  - ZFS and ldisfs OSTs can exist in same file system

Lustre 2.0

- State of Development
  - Userspace Servers, ZFS, Solaris
    - started in April 2007, Alpha in Jan 2008
  - Security
    - Mostly complete, developed on HEAD so didn't make 1.8
  - Replication & Change Logs
    - 1 - 2 developers, code complete by September
  - HSM (Hierarchical Storage Management)
    - CEA is primary developer
  - Windows Native Client
    - OSR is primary developer

Lustre 2.0

- Userspace Servers + ZFS == Solaris

> User space servers
  - Move Lustre server code from kernel to user space
  - Easier management of servers
  - Problems with servers won't require rebooting system
  - Portable to other types of hardware

> ZFS
  - End to end integrity via checksums
  - Higher file system limits with ZFS
  - Faster failover
  - Built-in snapshots
  - Efficient RAID and RAIDZ
  - Integrated volume management

Lustre 2.0

- Security (GSS/Kerberos)
  - Support **GSSAPI framework** in Lustre.
    - GSSAPI is an IETF standard that addresses the problem of many similar but incompatible security services in use today.
  - Support **Kerberos 5** as a mechanism of GSSAPI.
  - Support **user authentication** and integrity/privacy protection for messages between clients and MDS's.
  - Based on **MIT** implementation

Lustre 2.0

- Replication & Change Logs
  - **Replication - used to propagate changes from a master server to a separate target file system**
    - Synchronize frequent, short epochs
    - Data and attributes synchronized once per epoch
  - **Change Logs**
    - Maintain an operation log of namespace operations
    - Active log and one or more staged journals
    - Insertions, deletions and rename operations
    - Maintain a log of updated inodes per epoch
    - Log on MDS only (use mtime to determine changed files)

Lustre 2.0

- HSM
  - Lustre Community effort led by **CEA**
  - Will **interoperate** with existing storage systems
    - Single namespace to represent all backend storage
  - **No strong binding** - user space tool to copy in/out
  - **Transparent** to end user
  - All files always visible through Lustre
    - Files may reside in Lustre, backend storage, or both
  - Metadata is **always up to date**
    - Will add a migration status flag
  - Performance penalty only during cache miss

Lustre 2.0

- Network Request Scheduler
  - Like a **disk elevator** for the storage cluster
  - **Manages** incoming RPC requests
  - **Re-orders** IO request execution
    - Avoids client starvation
    - Presents optimized workload to backend filesystem
  - **Change** number of requests in-flight
    - Manages latency seen by each active client
    - Limits request buffering on the server
  - Currently finishing a simulator to model IO from client to server

Lustre 2.0

- Windows Native Client
  - Current Windows access is via CIFS (**pCIFS**)
  - **Native Lustre client** for Windows:
    - **Faster** performance
    - Native Windows behavior,(eg. FAT, NTFS)
    - Support for **Windows Server 2008** and **Windows Vista**
      - (32 bit and 64 bit (x64) versions)
    - Developed in partnership with **OSR**,
      - OSR is a leader in Windows file system development
        - (will use the OSR FSDK)
    - As a result, it's not open source

Lustre 3.0

- Planned for Summer, 2009
- Clustered Metadata
  - Allows metadata operations to be **distributed** over several servers
  - Can **increase MD ops** throughput by adding servers
  - Allows **scaling** performance with commodity hardware
- Migration capability to ZFS
  - Tools that support moving data from ldiskfs to ZFS will be available.

Future Features

- Filesets
- Flash Cache
- Migration
- Proxy Servers
- Solaris Client
- Sub-tree Locking
- Write Back Cache

Future Features

- Filesets
  - Collection of files on which operations can be performed
  - Replication might use this to clone subsets of a FS
- Flash Cache
  - Anticipating the change in storage hardware
  - Allow very fast writes to cache, move data more slowly to disk

Future Features

- Migration
  - Many features need intelligent migration capability:
    - HSM
    - Replication
    - Space Management
- Sub-tree Locking
  - Allow client to take locks on complete sub-directory
  - Useful for features like Writeback Cache

Future Features

- Proxy Servers
- **Definition**: A proxy server is a remote Object Storage Server that can cache data for remote users
  - Will keep current copy of data accessed by remote users
  - Data is local and shared to all remote users at a single location
  - Lustre locking is used to keep cached data coherent with file system
  - Ideal for small groups of users with WAN link to remote data center

Future Features

- Write Back Cache
- **Definition:** a client-side cache to hold MD operations.
  - Allows a client to make directory changes without immediately updating the metadata server (MDS)
  - Changes are collected on the client and sent en masse to MDS
  - Avoids round trip latency for each MDS operation; improves performance, especially for remote users
  - Entire directories are locked while a single client performs operations in that directory (sub-tree locking)

Other significant items

- Client IO Rewrite
  - not really an explicit feature, but a product improvement none-the-less
-
- Lustre End of Life Dates
  - 1.4 - June 30, 2009
  - 1.6 - December 31, 2009

bryon@sun.com