Demonstration Milestone for SMP Node Affinity

Overview

This document describes the work required to demonstrate that the SMP Node Affinity code meets the agreed acceptability criteria. The SMP Node Affinity code is functionally complete. The purpose of this final demonstration phase is to show that the code provides an enhancement to Lustre when used in a production-like environment.

The purpose of demonstration is to show appropriate functionality of the sub-project. This shall be done through execution of test cases designed to prove the Acceptance Criteria defined during the Solution Architecture.

Acceptance Criteria

SMP Node Affinity will be accepted as properly functioning if:

- Improving metadata performance (creation/removal/stat) with these conditions:
  - Reasonable number of clients (16+ clients)
  - Fat MDS (8+ cores)
  - Narrow stripecount file only (0, 1, 2, 4 stripecount)
- No performance regression for single client metadata performance
- No functionality regression

Baseline

The baseline measurement for this Demonstration will be Lustre 2.2 GA. SMP Node Affinity patches are included in Lustre Master the Lustre Master version will be compared against Lustre 2.2 GA for the purposes of demonstrating acceptance.

Hardware

MDS node:

- 2 x Intel Xeon(R) X5650 2.67GHz Six-core Processor (2-HT each core), which will present as 24 CPUs on Linux
- 24GB DDR3 1333MHz Memory
- 2PT 40Gb/s 4X QSFP InfiniBand adapter card (Mellanox MT26428)
- 1 QDR IB port on motherboard
- SSD as external journal device (INTEL SSDSA2CW120G3), SATA II Enterprise Hard Drive as MDT (single disk, WDC WD2502ABYS-02B7A0)

OSS:

- 2 x AMD Opteron 6128 2.0GHz Eight-Core Processor
- 16GB DDR3 1333MHz Memory
- 2PT 40Gb/s 4X QSFP InfiniBand adapter card (Mellanox MT26428)
- 1 QDR IB port on motherboard
- 3 x 1TB SATA II Enterprise Hard Drive (single disk, WDC WD1003FBYX-01Y7B0)

Client:

- Quad-Core Intel E5507 2.26G/4MB/800
- Mellanox ConnectX 6 QDR Infiniband 40Gbps Controller (MTS3600Q-1BNC)

- 12GB DDR3 1333MHz E/R memory

Network:

- Infiniband between all the nodes: MTS3600Q managed QDR switch with 36 ports.

Test Methodology

The following tools will be used to provide a production-like load. Performance of the MDS will be measured using LNet selftest and mdtest.

1. LNet selftest

Note: In-order to generate a suitable load, high "concurrency" is required on the client side. High concurrency in this environment is measured as over one thousand RPCs from 16 clients.

2. mdtest

- multiple mounts on each client.
  - each thread works under a private mount.
  - sufficient workload for "shared directory" is achievable because target directory has separate inode for each mount on client.
    - ⚠ It is possible to generate a high working load by turning off `mdc_rpc_lock` on all clients. `mdc_rpc_lock` however only works for directory per thread. For the shared directory case operations will be serialized by VFS mutex on client side. For this reason multiple mounts are chosen as they perform well in both cases.
- Verify server working load by checking total number of threads on MDS.
  - Service threads are created on demand. If there are sufficient active RPCs on server side the service threads number should reach the upper-limit.
  - If active RPCs are more than upper-limit of service threads number, they will not generate extra workload to filesystem or ptlrpc service threads. Each thread can handle one RPC at a time so the additional RPCs are just queued on the service.

Test File System Configuration

- 16 Clients, each with 32+ threads.
- 1 MDS, 2+ OSS (6 OSTs on each OSS).
- Test repeated three times. Median and max are recorded.
- Test completed with both Lustre 2.2 and Lustre-Master.

Test Results

LNet selftest results:

- aggregation selftest ping rate on server
  - Client [1, 2, 4, 8, 16] X Thread [32, 64]
- single client selftest ping rate
  - Client [1] X Thread [1, 2, 4, 8, 16, 32, 64]
- aggregation bandwidth of "selftest 4K BRW read & write" on server
  - Read: Client [1, 2, 4, 8, 16] X [32, 64] threads
  - Write: Client [1, 2, 4, 8, 16] X [32, 64] threads

mdtest results:

- all following tests should run for mknod, 1-stripe, 2-stripe, 4-stripe.
- total file number.
  - 256K files.
  - 1 million files.
- 32 mounts for each client, directory per thread.
  - Client [1, 2, 4, 8, 16] X 32 threads create/stat/unlink.
  - Directory per thread.
  - Shared directory for all threads.

Test Duration

LNet selftest

- 2 hours to setup environment, prepare scripts
- 2 releases, 2 hour for each release
- TOTAL: 6 hours

mdtest

- 4 hours to setup environment, prepare scripts
- 2 releases (v2_2, master)
- 2 different total files (256K, 1 million)
- 4 stripe setting (0, 1, 2, 4)
- 2 (dir-per-thread, shared dir) X 6 (1, 2, 4, 8, 12, 16 clients) = 12
- 3 (repeat)
- assume each round take 4 minutes
- TOTAL: 4 * 60 + 2 * 2 * 4 * 12 * 3 * 4 = 2544minutes = 42.4 hours

Estimated total time

- 6 + 42.4 = 48.4 hours

With these estimates, a minimum of 4 working days is required. One additional day should be included as a risk reserve. A further 1-2 days to summarize data.