Lustre 2.0 Alpha Release Test Plan

Author	Date	Description of Document Change	Client Approval By	Client Approval Date
Minh Diep	4/6/2009	First draft		

I. Test Plan Overview

This test plan will be used to verify the Lustre 2.0 for Alpha release. The goal is to find and resolve defects in the Lustre 2.0 codes for Alpha release.

Executive Summary

Lustre QE team will require a test plan that can verify Lustre 2.0 for Alpha release.

- Create a test plan to verify Lustre 2.0 for alpha release
- Inputs from the QE, and 2.0 C-team will be required
- The output will be a state of the build to help decide if Lustre 2.0 is ready for Alpha
- The following clusters will be used for testing:
  Jackie1, Jackie3, Yala1, Yala2 in Yala automation

To achieve a stable quality for Lustre 2.0 alpha release, testing must be focused on basic functionality and stability.

Problem Statement

The quality and stability of Lustre HEAD branch has not reached an acceptable level for alpha release. Therefore, Lustre team has decided to review and resolve all issues that can bring the quality of Lustre 2.0 to alpha level. This means the release will be focused on 1 architecture and 1 distribution with limited support.

Goal

The goal is to get Lustre 2.0 for alpha release at LUG.

Success Factors

Lustre 2.0 alpha release must pass acceptance-small test suite. All 2.0 alpha blockers must be resolved before release.

Testing Plan

Define the setup steps that need to happen for the hardware to be ready? Who is responsible for these tests?
For manual testing on clusters at Broomfield, the testing steps are as follows: 1) reserve test cluster time through Cluster Scheduler 2) setup the test cluster by running oslo on lts-head node 3) configure Lustre file system and start running the tests 4) send test results to Buffalo by running send_report.pl on lts-head node. QE team in Lustre group is responsible for setting up the test environment, running the tests, vetting and reporting the test results. For automated testing, the testing steps are as follows: 1) submit YALA testing request in YALA 2) vetting the test results on Buffalo. QE team is also responsible for the automated testing.

Specify the date these tests will start, and length of time that these test will take to complete.
Acceptance -small will be run every day

Specify (at a high level) what tests will be completed?	
Functional Tests	acceptance small test suite
Performance Tests	IOR
Stress Tests	Simul

Test Cases

Functional Test Cases		
All of the test cases in acceptance small test suite:		
no.	Test Case	Description
1.	RUNTESTS	basic regression tests with unmounting/remounting
2.	SANITY	tests that verify operation under normal operating conditions
3.	DBENCH	dbench benchmark for simulating N clients to produce the filesystem load
4.	BONNIE	Bonnie++ benchmark for creation, reading, and deleting many small files
5.	IOZONE	Iozone benchmark for generating and measuring a variety of file operations
6.	FSX	file system exerciser
7.	SANITYN	tests that verify operations from two clients under normal operating conditions

8.	~~LFSCK~~	tests e2fsck and lfsck to detect and fix filesystem corruption
9.	LIBLUSTRE	runs a test linked to a liblustre client library
10.	~~RACER~~	test for filesystem race conditions by concurrently creating, moving, deleting, etc. a set of files
11.	REPLAY_SINGLE	tests that verify recovery after MDS failure
12.	CONF_SANITY	tests that verify various Lustre configurations (including wrong ones), where the system must behave correctly
13.	RECOVERY_SMALL	tests that verify RPC replay after communications failure
14.	REPLAY_OST_SINGLE	tests that verify recovery after OST failure
15.	REPLAY_DUAL	tests that verify recovery from two clients after server failure
16.	INSANITY	tests multiple concurrent failure conditions
17.	SANITY_QUOTA	tests that verify filesystem quotas
18.	SANITY_SEC	tests that verify Lustre identity features
19.	REPLAY_VBR	tests that verify version base recovery features
20.	PERFORMANCE_SANITY	performance tests ported from CMD3 test suites (using mdsrate)

Performance Test Cases		
no.	Test Case	Description
1.	IOR	Industry HPC IO performance benchmark for testing performance of parallel file systems using various interfaces and access patterns.

Stress Test Cases		
no.	Test Case	Description
1.	Simul	An MPI coordinated test of parallel filesystem system calls and library functions. It was designed to perform filesystem operations simultaneously from many nodes and processes to test the correctness and coherence of parallel filesystems.

Supported Architectures for Alpha Release

Architectures
x86_64

Supported Distributions and Kernels for Alpha Release

Distribution	Kernel
RHEL 5	2.6.18-128.1.1.el5

Platforms, Network Types for Testing

	RHEL 5
x86_64	√
TCP (1GigE)	√
IB (OFED 1.3.1)	√

Proposed Time Line

Testing will be conducted daily until Alpha release at LUG (4/16/2009)

Scale Testing

Large scale testing will be performed on Nessie clusters with only IOR run to verify basic application on scale cluster works.

Automation Testing

Automation will be used by SUN QE to support this time line. The automation is not for external use. Acceptance small test suite is available in the Lustre source tree and can be used by the Open Source community.

Not Tested

QE will not test all Lustre configurations (ie, 1Cx1Mx1O, 1Cx1Mx2O, 1Cx1Mx3O...)
QE will not test all type of NIC and HBA including its drivers
QE will not test all machine configuration (ie, # of cpu, memory size, storage capacity...)
QE will not cover all industry performance tests.

There are several tests in acceptance-small test suite which target large scale failover. These tests will not be run prior to alpha release. These are:
Recovery-mds-scale, Recovery-double-scale, Recovery-random-scale

II. Test Schedule

Lustre 2.0.0 alpha

1. CVS tag: HEAD
2. Time Plan:
    - Start date: 03/24/2009
    - End date: 04/17/2009
1. Tracking Tickets on Bugzilla:
    - Lustre 2.0.0 release tracker: Bug 10695
    - Lustre 2.0.0 blocker: Bug 18996
1. New Landed Features: none
2. Supported Kernels:
    - 2.6.18-128.1.1.el5 (RHEL 5),

1. Test Matrix:

		YALA	**YALA**		
	Platform / **Test case**	RHEL5/x86_64	RHEL5/x86_64 *		
YALA	acc-sm	[2][3][9]	[2][9]		
YALA	IOR	[2][3][9]	[2][9]		
YALA	Simul	[2][3][9]	[2][9]		
Parinay	IOR at scale	[2][9]	[2][9]		

- TCP network run will be conducted several time instead of daily

NOTE:

[2] - Run test on patchless client.
[3] - Run test over IB (OFED 1.3.1) network.

[9] - Run test with flock locking enabled (mounting Lustre client(s) with "flock" option).

III. Test Plan Approval

- Review date for the Test Plan review with the 2.0 C-team
    -