TEST PARAMETERS AND FEATURE TEST PLANS

James Nunez, Intel HPDD
LAD Developer's Day
September 2016
Agenda

• Patch Test Parameters
• Feature Test Plan - Proposal
Patch Test Parameters
Overview

Continuous Integration

Developer → Gerrit

Gerrit → Autotest

Autotest → Maloo

Maloo → Lustre

Gerrit Gatekeeper → Maloo

Reviewers → Maloo

Jenkins

Build

Test

Review

Merge

*Other names and brands may be claimed as the property of others.
Stage 2: Test

Gerrit notifies the developer of test success or failure

- If a test fails, developer must analyze the failure and mark reason in Maloo
  - If patch caused failure, the patch must be fixed and resubmitted
    - Associate failed test(s) with LU ticket of the patch itself
  - If patch didn't cause failure, tests need to be restarted in Jenkins or Maloo
    - Associate failed test(s) with existing LU ticket, or Raise a new LU ticket
- A failed test session is not retriggers automatically
Stage 2: Test Notes

All test-groups must pass for a patch to be considered for landing

- The following test-groups are currently run in parallel test sessions
  - review-ldiskfs (~4 hours)
  - review-zfs-part-1 (~5 hours)
  - review-zfs-part-2 (~6 hours)
  - review-dne-part-1 (~8 hours)
  - review-dne-part-2 (~8 hours)

- Times for each test session can vary by more than an hour depending on node used

- Test-groups are composed of one or more Lustre test-suite
Changing Test Parameters

How:
  • Gerrit commit messages

Why:
  • Control what tests are run
  • Control Lustre* configuration
  • Control test environment variables

What:
  • Each Test-Parameters: line is in *addition* to the normal test sessions
Tests replay-vbr 41, 41, 4k and 10b and replay-single test 25 were added to the ALWAYS_EXCEPT list because they were failing on el7. The issues causing those failures have been fixed and landed in http://review.whamcloud.com/#/rc/14938/.

The replay-vbr and replay-single tests need to be re-enabled.

Test-Parameters: trivial testlist=replay-vbr, replay-single

Signed-off-by: James Nunez <james.a.nunez@intel.com>
Control Testing

Build Lustre* only, no testing:
- Test-Parameters: forbuildonly

Test only, do not land:
- Test-Parameters: fortestonly

“Trivial” changes only:
- Test-Parameters: trivial
- Test-Parameters: trivial testlist=conf-sanity
- Test-Parameters: trivial testgroup=full

Additional test configurations/runs for intermittent failures:
- Test-Parameters: testlist=conf-sanity,conf-sanity,conf-sanity
- Test-Parameters: testlist=conf-sanity,conf-sanity,conf-sanity,conf-sanity
Control Lustre* Configuration

Linux Distribution:

```bash
Test-Parameters: clientdistro=el6.7 ossdistro=el7 \ mdsdistro=sles11sp3
```

File system and server count:

```bash
Test-Parameters: mdtfilesystemtype=ldiskfs \ ostfilesystemtype=zfs mdscount=2 mdtcount=4
```

Interoperability:

```bash
Test-Parameters: clientjob=lustre-b2_7 mdsjob=lustre-b2_8 \ ossjob=lustre-b2_8 clientbuildno=38 serverbuildno=13
```

Other configurable file system parameters:

- combinedmdsmgs, clientcount, osscount, ostcount, clientarch=ppc
Control Test Suite Variables

Run SLOW tests:

Test-Parameters: envdefinitions=SLOW=yes

Specify test subset:

Test-Parameters: envdefinitions=ONLY=32 testlist=conf-sanity

Run tests with feature enabled during development:

Test-Parameters: envdefinitions=FILESET=subdir testlist=sanity
Test-Parameters: envdefinitions=SHAREDKEY=true testlist=sanity-sec
Feature Test Plan Proposal
Feature Test Plan - Proposal

Purpose

• Inform the community about a new Lustre* feature
• How to use the feature
• How was the feature tested

Attach to JIRA* ticket

Based on IEEE 829 “IEEE Test Plan Template”

Required for landing features
Feature Test Plan Information

Introduction

Documentation

Installation and Set up

What to Test

• Unit Testing
• Functional and Regression Testing
• Manual Testing

What to Test (continued)

• Performance Testing
• Scalability Testing
• User Interface
• Testing failure conditions

What not to test

Test Environment
Document Info and Introduction

Introduction

• a detailed description of the feature or a high-level description of the feature with a reference where the reader can get a detailed description of the feature

• the problem statement - why the feature is being developed or what problem will it solve

• expected use cases of the feature that motivated your development

• any requirements that the feature must meet

• JIRA tickets that track the progress of this feature. If there is a single parent ticket that lists all work being done in other JIRA tickets, listing this parent ticket is sufficient.

• describe any interactions between this new feature and existing features
Feature Documentation

Lustre Manual updates (LUDOC)


Man page updates

- Create a separate lfs-foo.1 or lctl-foo.8 page for subcommands
Feature Installation and Set-Up

How to install the feature
  – Including any external components needed for the feature to work

How to setup and configure the file system to make use of the feature

How to enable the feature

How to monitor the feature including
  – How to determine if the feature is working, running, stopped or has encountered a problem
  – Include any /proc or /sys files to monitor to see if the feature is progressing or working

How to, and when should the feature be turned off or paused
What to Test and What Not to Test

Unit tests
Functional and regression testing
Manual testing
Performance testing
Scalability testing
User interface components
Failure conditions to test (often omitted)
What not to test?
Environmental Needs

Hardware requirements to properly run or test this feature

Software requirements, outside of the file system, that need to be installed to properly use or test this feature

Does another feature need to be employed to test this feature?
  – e.g. your feature needs Kerberos enabled or only works with DNE

Non-standard testing tools required
References

Changing Test Parameters


Test Plan Template

• Will upload to lustre.org if there is support for this proposal
Questions?

james.a.nunez@intel.com
Legal Information

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer or learn more at intel.com.

Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. Consult other sources of information to evaluate performance as you consider your purchase. For more complete information about performance and benchmark results, visit http://www.intel.com/performance.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

Statements in this document that refer to Intel's plans and expectations for the quarter, the year, and the future, are forward-looking statements that involve a number of risks and uncertainties. A detailed discussion of the factors that could affect Intel's results and plans is included in Intel's SEC filings, including the annual report on Form 10-K.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel does not control or audit third-party benchmark data or the web sites referenced in this document. You should visit the referenced web site and confirm whether referenced data are accurate.

Intel and the Intel logo are trademarks of Intel Corporation in the United States and other countries.

*Other names and brands may be claimed as the property of others.

© 2016 Intel Corporation.