



Lustre* Manual

High Performance Data Division

Richard Henwood

LUG13: April 17th 2013

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

Overview

What is in the Lustre manual this instant?

*Network
Request
Scheduler*

Multiple MDTs!

4MB RPCS

But that's not all...

Multiple MDTs (DNE)

yesterday

Lustre Tuning parameters

yesterday

LFSCK

this morning

Wireshark

this morning

Change-logs

tomorrow

JobStats, mds-survey, tuning, debugging etc, etc, etc

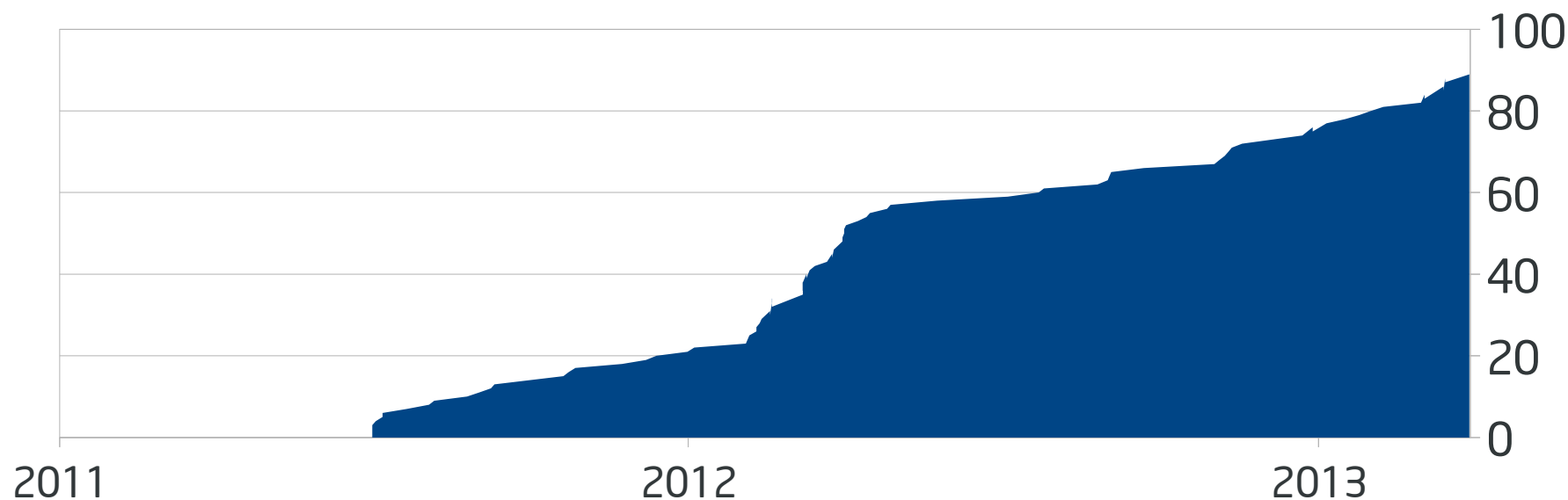
History

For me, it all started with:

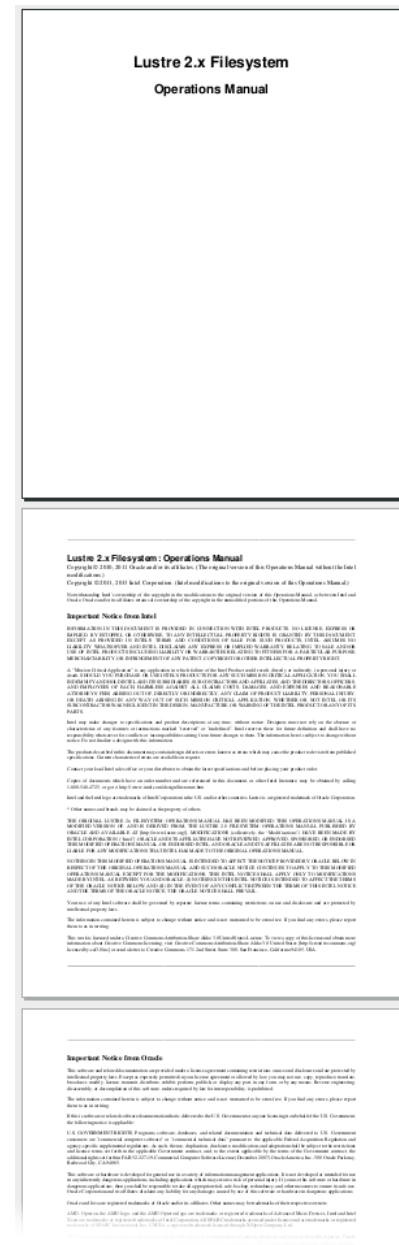
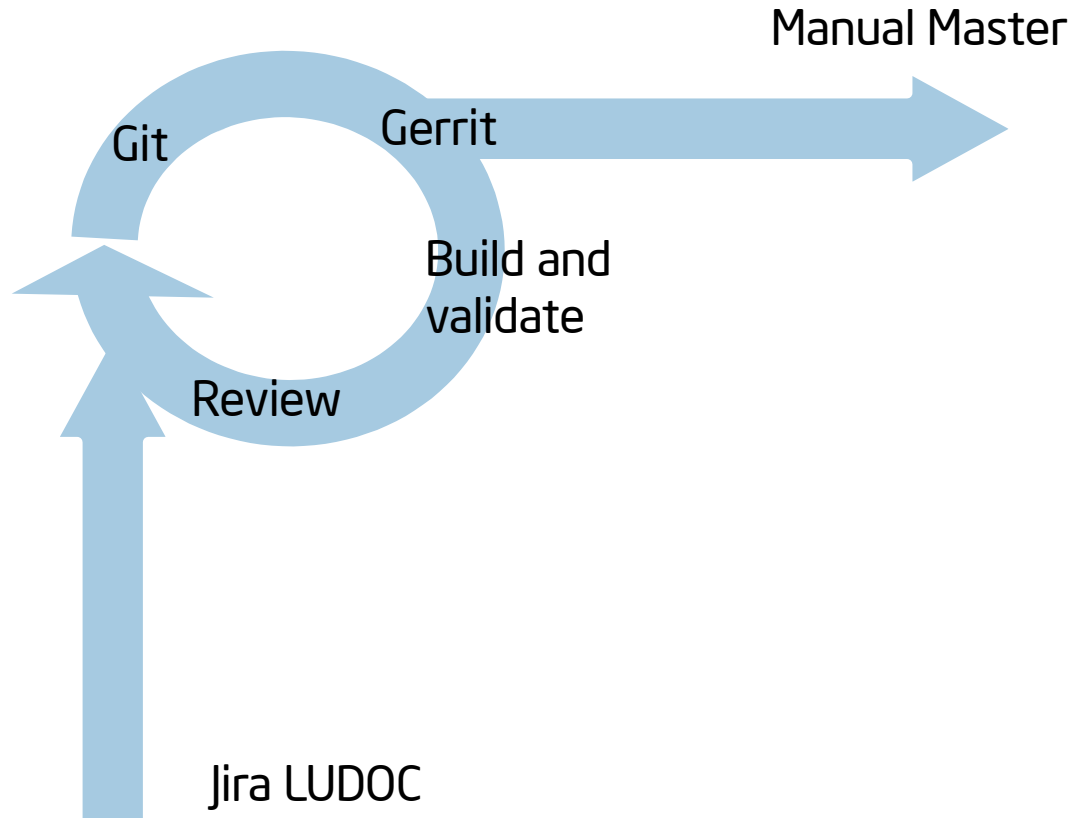
```
# lctl get_param llite.*.stats
snapshot_time          1308343279.169704 secs.usecs
dirty_pages_hits       14819716 samples [regs]
dirty_pages_misses     81473472 samples [regs]
read_bytes              36502963 samples [bytes] 1 26843582 55488794
write_bytes             22985001 samples [bytes] 0 125912 3379002
brw_read                2279 samples [pages] 1 1 2270
ioctl                  186749 samples [regs]
open                    3304805 samples [regs]
close                   3331323 samples [regs]
seek                    48222475 samples [regs]
fsync                   963 samples [regs]
truncate                9073 samples [regs]
setxattr                19059 samples [regs]
getxattr                61169 samples [regs]
```

Current status

- 118000 words, 13000 lines
- Available as pdf, html, epub
- ~90 commits



Contributing Engineering work-flow



Lustre Manual archaeology

Docbook

- Industry standard
- xml
- 400 tags
- xml editing

- Vim
- Emacs
- Bluefish
- Serna
- oXygen XML Editor

NOTE: a good tool will support
`xinclude` and be Docbook5
aware.

Lustre Manual archaeology

Docbook tags: an example with <replaceable>

```
<title>Synopsis</title>
<screen>
    lctl lfsck_stop -M |
        --device <replaceable>MDT_device</replaceable> \
            [-h | --help]
</screen>
</section>
```

27.4.1.2.1. Synopsis

```
lctl lfsck_stop -M | --device MDT_device \
    [-h | --help]
```


Lustre Manual specifics

```
<* conditional='l23'>Lustre 2.3| specific text.</*>
```

25.2. Binding MDS Service Thread to CPU Partitions

introduced in Lustre 2.3

With the introduction of Node Affinity (Node Affinity) in Lustre 2.3, MDS threads can be bound to particular CPU Partitions (CPTs). Default values for bindings are selected automatically to provide good overall performance for a given CPU count. However, an administrator can deviate from these setting if they choose.

- `mds_num_cpts=[EXPRESSION]` binds the default MDS service threads to CPTs defined by `EXPRESSION`. For example `mdt_num_cpts=[0-3]` will bind the MDS service threads to `CPT[0, 1, 2, 3]`.

Contributing continued...

- Low barriers to entry with submissions to

<http://jira.hpdd.intel.com/browse/LUDOC>

- Lots of opportunities for improvements, large and small

What to expect in future:

Manual for the Lustre File system - Mozilla Firefox

Manual for the Lustre File system

file:///home/rhenwood/manual/old_manual/tmp/en-US/html-desktop/index.html

Preface

- 1. About this Document
 - 1.1. UNIX Commands
 - 1.2. Shell Prompts
 - 1.3. Related Documentation
 - 1.4. Documentation, Support, and Training
- 2. Revisions

I. Introducing Lustre

- 1. Understanding Lustre
 - 1.1. What Lustre Is (and What It Isn't)
 - 1.1.1. Lustre Features
 - 1.2. Lustre Components
 - 1.2.1. Management Server (MGS)
 - 1.2.2. Lustre File System Components
 - 1.2.3. Lustre Networking (LNET)
 - 1.2.4. Lustre Cluster
 - 1.3. Lustre Storage and I/O
 - 1.3.1. Lustre File System and Striping
- 2. Understanding Lustre Networking (LNET)
 - 2.1. Introducing LNET
 - 2.2. Key Features of LNET
 - 2.3. Supported Network Types
- 3. Understanding Failover in Lustre
 - 3.1. What is Failover?
 - 3.1.1. Failover Capabilities
 - 3.1.2. Types of Failover Configurations
 - 3.2. Failover Functionality in Lustre
 - 3.2.1. MDT Failover Configuration (Active/Passive)

Table 5.1. Inode Ratio to be considered

LUN/OST size	Inode ratio	Total inodes
over 10GB	1 inode/16KB	640 - 655k
10GB - 1TB	1 inode/68kiB	153k - 15.7M
1TB - 8TB	1 inode/256kB	4.2M - 33.6M
over 8TB	1 inode/1MB	8.4M - 134M

You can specify the number of inodes on the OST file systems using the following option to the **--mkfs** option:

```
-N num_inodes
```

Alternately, if you know the average file size, then you can specify the OST inode count for the OST file systems using:

```
-i average_file_size / (number_of_stripes * 4)
```

For example, if the average file size is 16 MB and there are, by default 4 stripes per file, then **--mkfsoptions='-i 1048576'** would be appropriate.

Note

In addition to the number of inodes, file system check time on OSTs is affected by a number of other variables: size of the file system, number of allocated blocks, distribution of allocated blocks on the disk, disk speed, CPU speed, and amount of RAM on the server. Reasonable file system check times (without serious file system problems), are expected to take five and thirty minutes per TB.

For more details on formatting MDT and OST file systems, see [Section 6.4, "Formatting Options for RAID Devices"](#).

5.3.4. File and File System Limits

Table 5.2, "File and file system limits" describes file and file system size limits. These limits are imposed by either the

