

Mainlining Lustre Client

Tao Peng
EMC Fastdata Group

Outline

- Motivations
- State of Art
- Future Plans

WHY?

- Lustre is very popular in HPC world
 - Six of the top 10 and more than 60 of the top 100 supercomputers in the world have Lustre file systems
- But being outside of the kernel, means:
 - Limited distribution support
 - Limited commercial market users
 - Continuous requests for new kernel support

Benefits of Upstreaming

- Bring Lustre to more and more users
- Better integration with Linux kernel
- Opportunity to include Lustre in standard distributions
- Ease the burden of catching up with continuous new kernel API changes

Overview

- A collaboration of Whamcloud/Intel/EMC
- Many people helped (Cray, ORNL, LLNL, SUSE, Gentoo, kernel community developers, etc.)
- 10+ LU tickets (LU-709, LU-1347, LU-1337, LU-1214, LU-1756, LU-1113, LU-1994, LU-2850, LU-2335, etc.)
- 100+ patches merged. ~10 pending review.
- ~20K LOC touched

Current Status

- The code is already merged in staging directory of upstream Linux kernel
 - drivers/staging/lustre/lustre and drivers/staging/lustre/lnet
 - About 250K LOC
 - 200+ incremental patches applied since first merged
- Depends on CONFIG_STAGING
- Enable CONFIG_LUSTRE_FS and build
- Can only be built as kernel module
 - Libcfs causes dependency circle when built-in

Current Status – work items

- Coding style change
 - Done in upstream. On going in Lustre tree.
- Client/server split up
 - On going
- Obsolete macro cleanup
 - Done in upstream
- Latest kernel support
 - Done in upstream. Ongoing in Lustre tree.
- Kconfig/Kbuild support
 - Done in upstream
- Most were done prior to porting to Linux kernel

Current Status – work items

- Architecture support
 - Still disabled on MIPS/XTENSA/SUPERH
- Libcfs cleanup
 - On going
- Sync patches between upstream and Lustre tree
 - On going
- Many other cleanups required by Kernel maintainers
 - watchdog
 - access_process_vm
 - OBP/MDP/CTXTP

Future Plans

- Libcfs is the next big target to eliminate
 - Much of it is there to support MacOS/liblustre/WinNT
 - As Andreas said, those platform code were planned to be removed in the next Lustre release
 - Expect more changes in this area
- Port patches between upstream kernel and Lustre tree continuously
- Need more testing on the upstream kernel code
 - Test HSM related features
 - EMC also has HSM solution that integrate with Isilon

Future Plans

- Lots of compat code needs to be cleaned up
- CLIO need reshaping to better use kernel's readahead/writeback mechanisms
- When code is cleaned up to some state, we can engage kernel maintainers to review the code and eventually get it out of drivers/staging/ directory
- Long journey, EMC is committed
- Any help is welcome!

How to Get It?

- Just git clone Vanilla kernel tree
 - Git clone
 - [git://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git](https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git)
- Enable CONFIG_LUSTRE_FS
- Make
- Still need user space tools from Lustre tree...

What If I Want to Contribute?

- Greg KH's staging tree is the starting place
 - git clone
`git://git.kernel.org/pub/scm/linux/kernel/git/gregkh/staging`
`g.git`
- Code cleanup
- Patch porting
- Building/Testing
- Any form of contribution is welcomed!

Q&A

Thanks

For more information please contact:

faibish_sorin@emc.com

tao.peng@emc.com

Grau Data XtremStore

- HSM solution for Lustre 2.5 using Isilon backend and PLFS

Grau Data Lustre HSM to Isilon PoC

