Milestone Completion for the SMP Node Affinity Subproject on the Single Metadata Server Performance Improvements Project of the SFS-DEV-001 contract.

Revision History

Date	Revision	Author
13th July 2012	Original	R. Henwood
13th Sept 2012	IB tests included	R. Henwood

Contents

Introduction

The following milestone completion document applies to Subproject 1.1 – SMP Node Affinity subproject of the Single Metadata Server Performance Improvements within the OpenSFS Lustre Development contract SFS-DEV-001 signed 7/30/2011.

Subproject Description

Per the contract, Implementation milestone is described as follows: “This subproject splits the computing cores available on the Metadata Server (MDS) into a configurable number of compute partitions, and binds the Lustre RPC service threads to run within a specified compute partition. This allows the RPC threads to run more efficiently by keeping data structures in cache memory close to the CPU cores on which they are running, and avoids needles contention on the inter-CPU memory subsystem. SMP Node Affinity also allows individual RPC requests to stay local to a specific compute partition, improving overall efficiency throughout the protocol stack as the number of cores increases.”

Milestone Completion Criteria

Per the contract, Implementation milestone is described as follows: “Contractor shall complete implementation and unit testing for the approved solution. Contractor shall regularly report feature development progress including progress metrics at project meetings and engineers shall share interim unit testing results as they are available. OpenSFS at its discretion may request a code review. Completion of the implementation phase shall occur when the agreed to solution has been completed up to and including unit testing and this functionality can be demonstrated on a test cluster. Code Reviews shall include:

a. Discussion led by Contractor engineer providing an overview of Lustre source code changes
b. Review of any new unit test cases that were developed to test changes

Location of Completed Solution

The agreed solution has been completed and is recorded in the following patches:

Code Review	Commit
3268	ptlrpc: post rqbd with flag LNET_INS_LOCAL
3135	ptlrpc: CPT affinity ptlrpc RS handlers
3133	ptlrpc: partitioned ptlrpc service

2725	o2iblnd: CPT affinity o2iblnd
3252	lnet: re-finalize failed ACK or routed message
2718	ksocklnd: CPT affinity socklnd
3238	lnet: wrong assertion for optimized GET
3193	lnet: tuning wildcard portals rotor
2805	lnet: SMP improvements for LNet selftest
3141	ldlm: SMP improvement for ldlm_lock_cancel
2911	ptlrpc: Reduce at_lock dance
2729	libcfs: CPT affinity workitem scheduler
2824	obdclass: SMP improvement for lu_key
3180	lnet: multiple cleanups for inspection
3114	lnet: allow user to bind NI on CPTs
3113	lnet: Partitioned LNet networks
3091	lnet: cleanup for rtrpool and LNet counter
3078	lnet: Partitioned LNet resources (ME/MD/EQ)
2917	ptlrpc: cleanup of ptlrpc_unregister_service
2912	ptlrpc: svc thread starting/stopping cleanup
3070	lnet: reduce stack usage of "match" functions
3056	lnet: Granulate LNet lock
2895	ptlrpc: partition data for ptlrpc service
3048	lnet: code cleanup for lib-move.c
3043	lnet: match-table for Portals
3010	lnet: code cleanup for lib-md.c
2997	lnet: split lnet_commit_md and cleanup
2983	lnet: LNet message event cleanup
2933	lnet: eliminate a few locking dance in LNet
2932	lnet: parse RC ping in event callback
2930	lnet: router-checker (RC) cleanup
2879	ptlrpc: common code to validate nthreads
2926	lnet: move "match" functions to lib-ptl.c
2925	lnet: allow to create EQ with zero eq_size
2924	lnet: cleanup for LNet Event Queue
2923	lnet: new internal object lnet_peer_table
2878	ptlrpc: clean up ptlrpc svc initializing APIs
2922	lnet: container for LNet message
2921	lnet: abstract container for EQ/ME/MD
2919	lnet: add lnet_*_free_locked for LNet
2558	libcfs: more common APIs in libcfs
2920	libcfs: export a few symbols from libcfs
2523	libcfs: NUMA allocator and code cleanup
2461	libcfs: implementation of cpu partition
2346	libcfs: move range expression parser to libcfs

Demonstrate any new tests that have been developed.

SMP Node Affinity does not require new functional tests as this project is a performance enhancement.

During the course of development, two small changes were made to the existing tests.

1. Force enable multiple CPU partitions for autotest. By default, libcfs will create multiple CPU partition only for system with > 4 CPU cores. It is preferential to run test with multiple CPU partitions for all SMP machines. A patch was developed to always enable multiple CPU partitions on systems with multiple cores.
2. Minor issue fixes. Now multiple CPU partitions are provided modifications to the tests were required to work around brittle interractions between autotest and the procfs subsystem.

These changes are recorded as http://review.whamcloud.com/#change,3288

The completion of these modified tests is recorded as https://maloo.whamcloud.com/test_sessions/076bf58e-ca29-11e1-9192-52540035b04c

A subsequent test on IB is included in Appendix 2 recorded at https://maloo.whamcloud.com/test_sessions/2912130e-fd4f-11e1-b09c-52540035b04c

Demonstration of SMP Node Affinity functionality.

After landing the final patch, the complete test framework is recorded as completing at the following record:

https://maloo.whamcloud.com/test_sessions/076bf58e-ca29-11e1-9192-52540035b04c

The result detail is recorded in Appendix 1.

Conclusion

Implementation has been completed according to the agreed criteria.

Appendix 1 Autotest results on TCP/IP

Session for group review (fat-intel-3vm6, liang)

Uploaded by: Whamcloud Autotest.
Reason: landing.
12 test sets passed out of 12.
Code review references

- gerrit:3288
  id: b365fcb82a38761a4c40ff09ed653b7654a77d9e
  change_no: 3288
- jira:LU-1607
  id: LU-1607

Test sets

Name	Test group	Test host	Branch	Arch / Lustre Version	Run at (UTC)	Duration	Subtests passed	Bugs	Links	User	Status
mmp	review	fat-intel-3vm6	• master	• x86_64,server,el6,inkern • x86_64,client,el5,inkern	2012-07-10 00:10:34	177	10/10		gerrit:3288, jira:LU-1607	liang	PASS
lnet-selftest	review	fat-intel-3vm6	• master	• x86_64,server,el6,inkern • x86_64,client,el5,inkern	2012-07-10 00:05:12	319	1/1		gerrit:3288, jira:LU-1607	liang	PASS
lustre-rsync-test	review	fat-intel-3vm6	• master	• x86_64,server,el6,inkern • x86_64,client,el5,inkern	2012-07-09 23:59:21	342	14/14		gerrit:3288, jira:LU-1607	liang	PASS
sanity-sec	review	fat-intel-3vm6	• master	• x86_64,server,el6,inkern • x86_64,client,el5,inkern	2012-07-09 23:56:24	177	7/7		gerrit:3288, jira:LU-1607	liang	PASS
sanity-quota	review	fat-intel-3vm6	• master	• x86_64,server,el6,inkern • x86_64,client,el5,inkern	2012-07-09 23:16:57	2358	35/35		gerrit:3288, jira:LU-1607	liang	PASS
insanity	review	fat-intel-3vm6	• master	• x86_64,server,el6,inkern • x86_64,client,el5,inkern	2012-07-09 22:53:10	1419	11/11		gerrit:3288, jira:LU-1607	liang	PASS
replay-ost-single	review	fat-intel-3vm6	• master	• x86_64,server,el6,inkern • x86_64,client,el5,inkern	2012-07-09 22:42:07	654	12/12		gerrit:3288, jira:LU-1607	liang	PASS
recovery-	revie	fat-intel-	• maste	• x86_64,server,el6,inker	2012-07-09	2085	55/55		gerrit:328	liang	PASS

small	w	3vm6	r	n • x86_64,client,el5,inkern	22:07:14			8, jira:LU-1607		
conf-sanity	review	fat-intel-3vm6	• master	• x86_64,server,el6,inkern • x86_64,client,el5,inkern	2012-07-09 20:48:30	4724	81/81	gerrit:3288, jira:LU-1607	liang	PASS
replay-single	review	fat-intel-3vm6	• master	• x86_64,server,el6,inkern • x86_64,client,el5,inkern	2012-07-09 19:51:15	3435	92/92	gerrit:3288, jira:LU-1607	liang	PASS
sanityn	review	fat-intel-3vm6	• master	• x86_64,server,el6,inkern • x86_64,client,el5,inkern	2012-07-09 19:27:52	1402	107/107	gerrit:3288, jira:LU-1607	liang	PASS
sanity	review	fat-intel-3vm6	• master	• x86_64,server,el6,inkern • x86_64,client,el5,inkern	2012-07-09 18:19:06	4126	421/421	gerrit:3288, jira:LU-1607	liang	PASS

Test nodes

fat-intel-3vm3

Kernel Version:	2.6.32-220.17.1.el6_lustre.g4a711e4.x86_64
Lustre Version:	jenkins-arch=x86_64,build_type=server,distro=el6,ib_stack=inkern
OS:	GNU/Linux
Networks:	tcp
Memsize:	1.96 GB
Lustre Build:	http://build.whamcloud.com/job/lustre-reviews/7631
Architecture:	x86_64
File System:	ldiskfs
Lustre Branch:	master
Node Architecture:	x86_64
Services:	MDS 1
Lustre Revision:	b365fcb82a38761a4c40ff09ed653b7654a77d9e
Distribution:	CentOS release 6.2
Name:	fat-intel-3vm3

fat-intel-3vm4

Kernel Version:	2.6.32-220.17.1.el6_lustre.g4a711e4.x86_64
Lustre Version:	jenkins-arch=x86_64,build_type=server,distro=el6,ib_stack=inkern

OS:	GNU/Linux
Networks:	tcp
Memsize:	1.96 GB
Lustre Build:	http://build.whamcloud.com/job/lustre-reviews/7631
Architecture:	x86_64
File System:	ldiskfs
Lustre Branch:	master
Node Architecture:	x86_64
Services:	OST 6, OST 7, OST 2, OST 3, OST 4, OST 5, OST 1
Lustre Revision:	b365fcb82a38761a4c40ff09ed653b7654a77d9e
Distribution:	CentOS release 6.2
Name:	fat-intel-3vm4

fat-intel-3vm5

Kernel Version:	2.6.18-238.19.1.el5
Lustre Version:	jenkins-arch=x86_64,build_type=client,distro=el5,ib_stack=inkern
OS:	GNU/Linux
Networks:	tcp
Memsize:	1.96 GB
Lustre Build:	http://build.whamcloud.com/job/lustre-reviews/7631
Architecture:	x86_64
File System:	ldiskfs
Lustre Branch:	master
Node Architecture:	x86_64
Services:	Client 1
Lustre Revision:	b365fcb82a38761a4c40ff09ed653b7654a77d9e
Distribution:	CentOS release 5.8
Name:	fat-intel-3vm5

fat-intel-3vm6

Kernel Version:	2.6.18-238.19.1.el5
Lustre Version:	jenkins-arch=x86_64,build_type=client,distro=el5,ib_stack=inkern
OS:	GNU/Linux

Networks:	tcp
Memsize:	1.96 GB
Lustre Build:	http://build.whamcloud.com/job/lustre-reviews/7631
Architecture:	x86_64
File System:	ldiskfs
Lustre Branch:	master
Node Architecture:	x86_64
Services:	Client 2
Lustre Revision:	b365fcb82a38761a4c40ff09ed653b7654a77d9e
Distribution:	CentOS release 5.8
Name:	fat-intel-3vm6

Appendix 2 Autotest results on IB

Session for group review (client-23-ib, liang)
Uploaded by: Whamcloud Autotest.

Reason: landing.

12 test sets passed out of 12.

Code review references

gerrit:381