Rationalizing Message Logging for Lustre

John Hammond
ICES/TACC
The University of Texas at Austin

jhammond@ices.utexas.edu

Collaborators

- James C. Browne, U.T., ICES/CS
- Tommy Minyard, U.T., TACC
- Chris Jordan, U.T., TACC
- Kuo Shyh-Hao, IHPC, A*STAR
- Lee Kee Khoon, IHPC, A*STAR

Overview

- Goals
- Motivation
- Approach
- Requirements on Lustre
- Results and Status
- Future Work
- Solicitation

Goals

Enable effective diagnosis of fault/failure diagnosis and recovery for Lustre

- Make message log both readily parsable and easily human readable

- Enable fast and accurate on-line analysis of the health of a Lustre installation

Ranger - Hardware Summary

- Compute power - 579 Teraflops
  - 3,936 Sun four-socket blades
  - 15,744 AMD “Barcelona” quad-core processors
- Memory - 123 Terabytes
  - 2 GB/core, 32 GB/node
  - ~20 GB/sec memory B/W per node
- Disk subsystem - 1.7 Petabytes
  - 72 Sun x4500 “Thumper” I/O servers, 24TB each
  - 40 GB/sec total aggregate I/O bandwidth
  - 1 PB raw capacity in largest filesystem
- Interconnect - 1 GBps, 1.6–2.85 µsec latency
  - 2 Sun InfiniBand switches, up to 3456 4x ports each
  - Full non-blocking 7-stage Clos fabric

Ranger kernel logging

30K—2M msgs per day, 5K—200K Lustre

Only 5—10% saved to disk

New message type every few days.

Traditional approaches to system logging are not up to the task.

Non-Diagnostic Messages

- LustreError: 7424:0: (quota_master.c:514:mds_quota_adjust()) mds adjust qunit failed! (opc:4 rc:-122)

- LustreError: 9247:0: (ldlm_lib.c:1643:target_send_reply_msg()) @@@ processing error (-107) req@000001018ac30c00 x152525432/t0 o400-><?>@<?>:0/0 lens 128/0 e 0 to 0 dl 1267778275 ref 1 fl Interpret:H/0/0 rc -107/0

Obscure Messages

- Understanding Lustre (kernel, [insert FOSS project name],...) trace requires careful reading of source

```
 Jan 19 16:58:45 i110-301 kernel: [3038184.371463]
 LustreError: 5647:0:
(events.c:66:request_out_callback())
 @@@ type 4, status -5  req@ffff81060142b000
x70401815/t0
 o8->work-OST003c_UUID@129.114.97.37@o2ib:28/4 lens
 304/456 e 0 to 1 dl 1263941647 ref 1 fl
Complete:EXN/0/0
 rc -110/0
```

Incomplete Messages

- Diagnosing causes of OSS hangs requires trace not currently recorded (when generated)

```
Call Trace:
<ffffffffa005097d>{:raid5:get_active_stripe+388}
<ffffffff801346fa>{__wake_up+54}
<ffffffff80134653>{default_wake_function+0}
<ffffffffa005291d>{:raid5:make_request+740}
<ffffffff8015a905>{find_get_page+65}
<ffffffff8017c00f>{__find_get_block_slow+62}
<ffffffff80255876>{generic_make_request+361}
<ffffffff8013602c>{autoremove_wake_function+0}
<ffffffff80255982>{submit_bio+247}
...
```

Redundant Messages
OST refuses reconnect request

```
... work-OST002c: 83ddb83f-ebeb-41f6-6059-84e4386954b0
reconnecting


... work-OST002c: refuse reconnection from 83ddb83f-
ebeb-41f6-6059-84e4386954b0@129.114.105.213@o2ib to
0x000001018f3d0000; still busy with 3 active RPCs


... @@@ processing error (-16)  req@000001012732ec50
x32535133/t0 o8->83ddb83f-ebeb-41f6-6059-
84e4386954b0@NET_0x50000817269d5_UUID:0/0 lens 304/200 e
0 to 0 dl 1257857619 ref 1 fl Interpret:/0/0 rc -16/0
```

Extracting data from messages

```
Source:
 ...
 printk(KERN_ERR "device failed: Error %d: %s\n", errno, errmsg);
 ...
```

```
Message:

 Mar 10 15:09:30 myhost kernel: device failed: Error 150: Not a toaster
```

```
Monitor:

 msg = get_next_syslog_msg();
 /* Extract message date, hostname, program... */

 /* Try to match "device failed: ..." and recover arguments. */
 if (sscanf(msg, "device failed: Error %d: %[^\n]\n", &err, buf) == 2) {
   /* Handle message. */
   /* ... */
 }
```

Logs difficult (impossible) to parse

```
int main ( ... )
{
  char *s1, *s2;
  /* ... */

  printf(“%s%s”, s1, s2);
}
```

Assume the output is "1234", then find s1 and s2.

Real examples

```
Jan 19 16:58:45 i110-301 kernel: [3038184.371463]
LustreError: 5647:0:
(events.c:66:request_out_callback()) @@@ type 4,
status -5  req@ffff81060142b000 x70401815/t0 o8-
>work-OST003c_UUID@129.114.97.37@o2ib:28/4 lens
304/456 e 0 to 1 dl 1263941647 ref 1 fl
Complete:EXN/0/0 rc -110/0
```

```
Jan 26 21:26:12 mds3 kernel: LustreError: 0:0:
(ldlm_lockd.c:305:waiting_locks_callback()) ### lock
callback timer expired after 50s: evicting client at
129.114.105.107@o2ib  ns: mds-scratch-MDT0000_UUID
lock: 000001016ace8940/0xf9157c8032284b99 lrc: 3/0,0
mode: CR/CR res: 269333475/1604567023 bits 0x3 rrc: 2
type: IBT flags: 0x4000020 remote: 0x2312e9da3f6a73c3
expref: 55 pid: 13656 timeout: 4479804771
```

System Log Rationalization

- Make message content parsable and human readable
- Rationalize message priorities, delete redundant messages
- Create multiple streams for different purposes
  - Fault/failure diagnosis
  - State machine synthesis
  - Resource/performance management

Approach

- Rationalize message handling by Linux syslog stack
  - Message formatting
  - Partition message stream
- Rationalize Lustre message stream
  - Uniform message structure
  - Revise message priorities to ensure critical messages are delivered

Rational encoding

Modify `printk()` by inserting call to a new function, `rat_printk()`.

Every call to `printk()`,

```
printk(KERN_ERR "device failed: Error %d: %s\n", errno, errmsg);
```

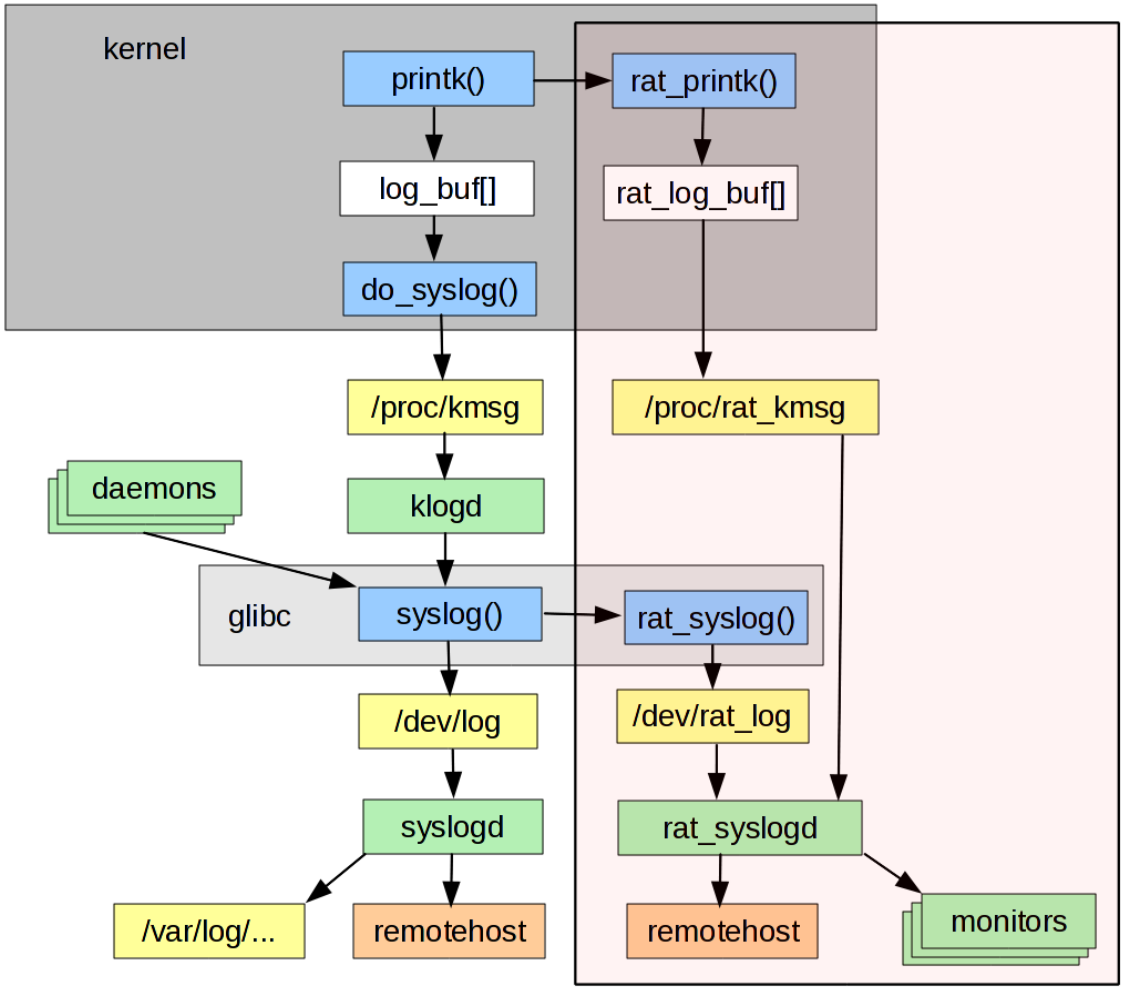
would produce the normal message:

```
Mar 10 15:09:30 myhost kernel: device failed: Error 157: Not a toaster
```

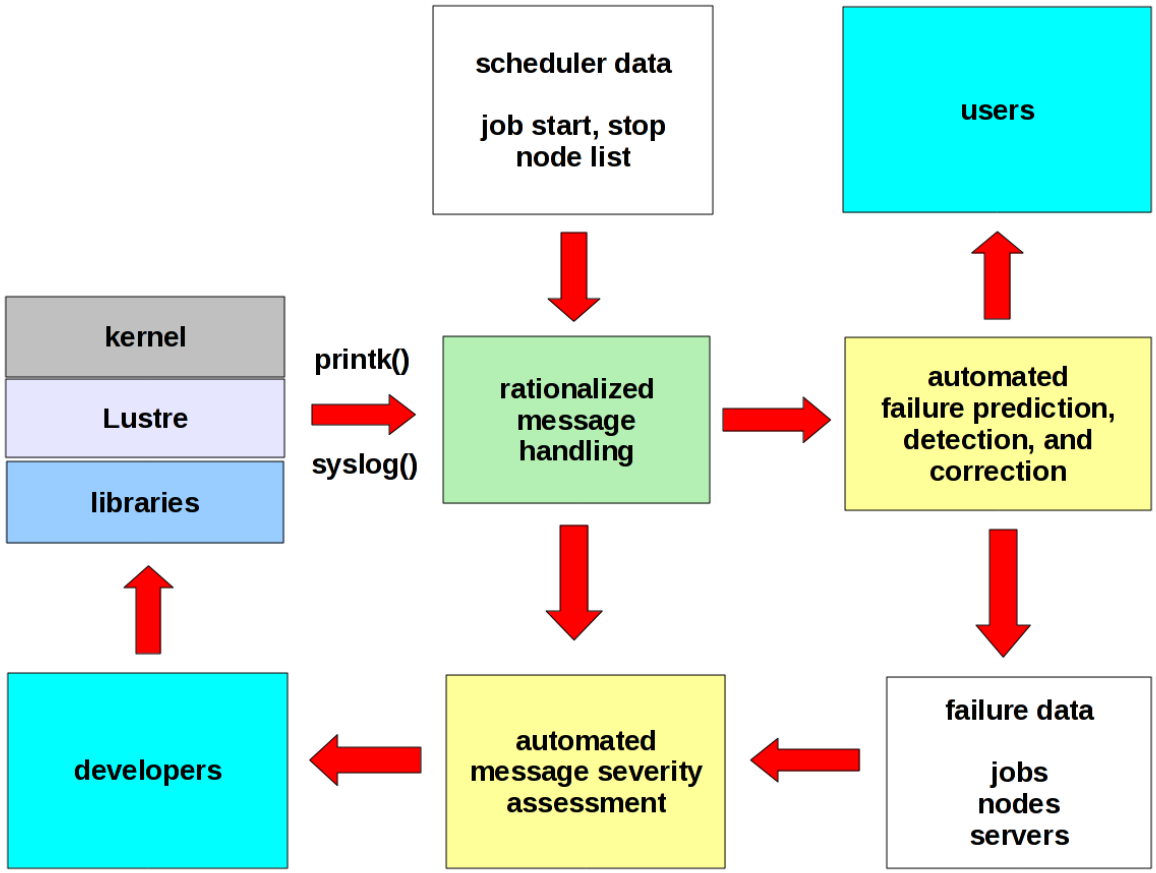
along with a rationalized version of the same:

```
time:1268255370■
host:myhost■
progname:kernel■
0:<3>Print failed: Error %d: %s\n■
1:157■
2:Not a toaster■■
```

Rationalized encoding stack



Rationalization Workflow



Requirements on Lustre

- Modification message formatting macros to take advantage of rationalized printk()

  Won't break interfaces!

- Participation in message rationalization workflow

  Also, won't break interfaces!

Solicitation

- This project will make Lustre a more effective and usable system

- We solicit collaboration with:
  - The Lustre Development group
  - Any Lustre users who would like to participate in this project