

Cray Data Management Platform: Cray Lustre File System Monitoring esfsmon

**Jeff Keopp, Cray Inc.
Harold Longley, Cray Inc.**

Safe Harbor Statement

This presentation may contain forward-looking statements that are based on our current expectations. Forward looking statements may include statements about our financial guidance and expected operating results, our opportunities and future potential, our product development and new product introduction plans, our ability to expand and penetrate our addressable markets and other statements that are not historical facts. These statements are only predictions and actual results may materially vary from those projected. Please refer to Cray's documents filed with the SEC from time to time concerning factors that could affect the Company and these forward-looking statements.

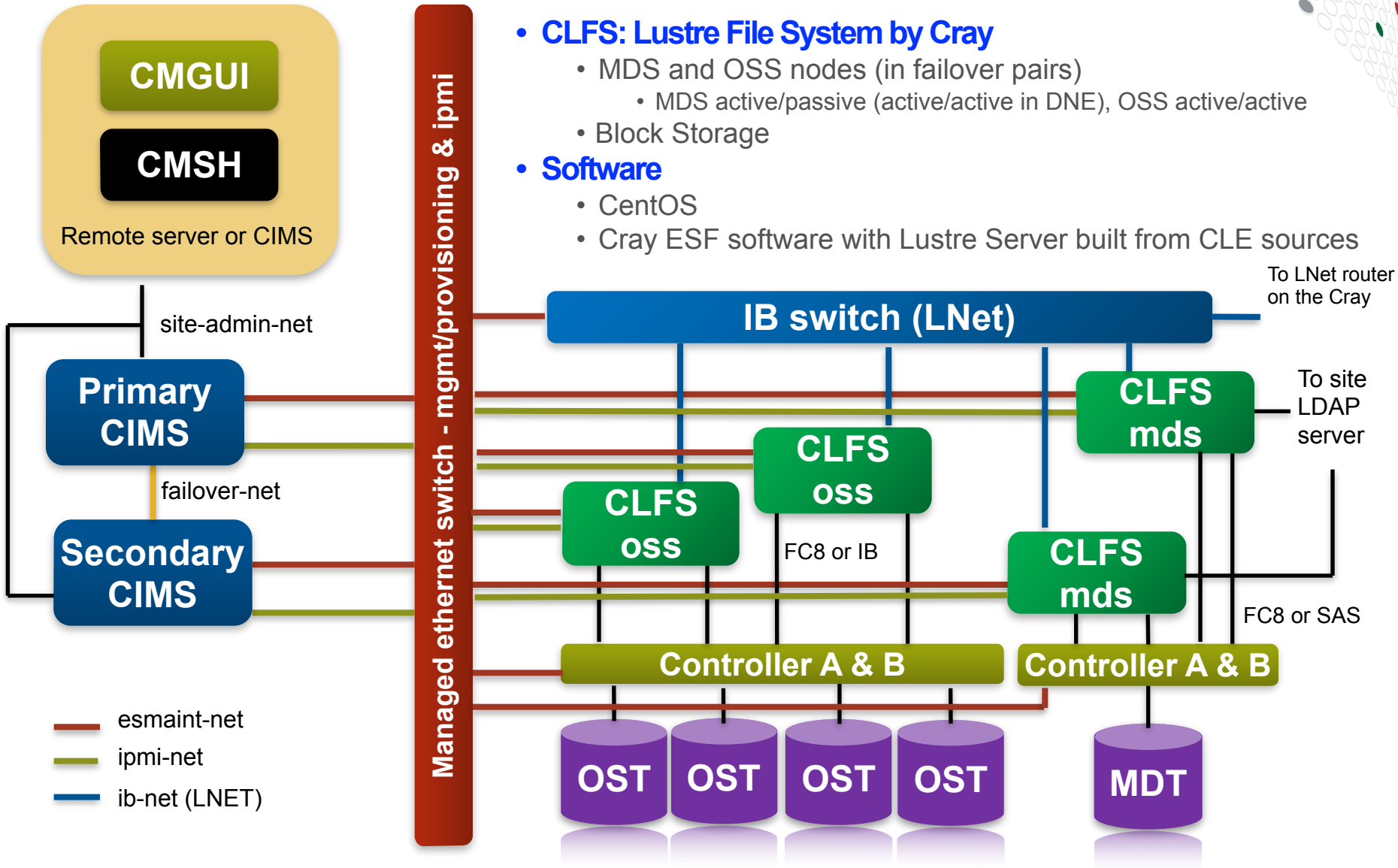


Overview: External Lustre Filesystems

- **Cray External Lustre Filesystems**
 - **Lustre File System by Cray (CLFS)**, previously esFS)
 - **Cray Sonexion®**
- **Management and Monitoring**
 - **CLFS**
 - **Cray Integrated Management System (CIMS)**, previously esMS)
 - **lustre_control** - controls the Lustre filesystem
 - **esfsmon** - monitors and performs automated Lustre failover
 - **Lustre Monitoring Tool** - gathers data from /proc/fs/lustre
 - **Cray Sonexion**
 - **Cray Sonexion System Manager (CSSM)**
 - **Unified System Management firmware (USM)**

This discussion is focused on monitoring and automated failover of CLFS

CLFS Overview



- **CLFS: Lustre File System by Cray**
 - MDS and OSS nodes (in failover pairs)
 - MDS active/passive (active/active in DNE), OSS active/active
 - Block Storage
- **Software**
 - CentOS
 - Cray ESF software with Lustre Server built from CLE sources

— esmaint-net
— ipmi-net
— ib-net (LNET)

COMPUTE | STORE | ANALYZE



Cray Lustre Filesystems – Lustre Control

- **Controlling Lustre - `lustre_control`**
 - **Performs the following filesystem operations**
 - Format/Reformat
 - Writeconf
 - Start/Stop/Tune
 - Status
 - Manual Failover/Failback
 - **Executed from the CIMS**
 - Components on both the CIMS and CLFS nodes



Cray Lustre Filesystems – Lustre Control

- **Controlling Lustre - `lustre_control`**
 - **Filesystems are defined in `fs_defs` files**
 - “installed” into `lustre_control` - not directly used in operation
 - `# lustre_control install fsname.fs_defs`
 - `/opt/cray/esms/cray-lustre-control-XX/default/etc/example.fs_defs`
 - **Filesystem tuning defined in `fs_tune` files**
 - Used directly in operation
 - `# lustre_control set_tune -f fsname <path_to_fs_tune_file>`
 - `/opt/cray/esms/cray-lustre-control-XX/default/etc/example.fs_tune`



Automated Lustre Failover Monitor - esfsmon

- **Automated Lustre Failover Monitor**
 - **Implemented as a custom health check**
 - `/cm/local/apps/cmd/scripts/healthchecks/esfsmon_healthcheck`
 - **Executed on the CIMS**
 - Monitors filesystem nodes via management daemon on each node
 - **Monitors multiple CLFS filesystems**
 - Supports Lustre DNE (Distributed NamespacE) and non-DNE CLFS
 - **Health check failure triggers failover of Lustre assets**
 - `/cm/local/apps/cmd/scripts/actions/esfsmon_action`
 - Calls *lustre_control* to perform the failover of Lustre assets
 - Supports multiple failovers
 - Previous version went into RUNSAFE mode after first failover



Automated Lustre Failover Monitor - esfsmon

- **Operational Modes**

- **NORMAL** – monitors and will execute automated failover
 - Failures are logged to `/var/log/messages`
 - Failover information logged to `/tmp/esfsmon/fsname.fo_log`
- **RUNSAFE** – monitors but will **not** execute automated failover
 - Failure incidents are logged to `/var/log/messages`
 - Set by existence of `/var/esfsmon/esfsmon_runsafe_fsname`
- **SUSPENDED** – not monitored
 - Set by existence of `/var/esfsmon/esfsmon_suspend_fsname`
- **NOTE:** esfsmon will set suspended mode during failover and failback operations or when *lustre_control* is stopping the filesystem.



Automated Lustre Failover Monitor - esfsmon

- **Operation**

- Checks CLFS nodes every 2 minutes (configurable)

- **CLFS nodes are grouped by categories per filesystem**

- Checks performed by category - excluding 'esfs-failed-fsname'
 - **esfs-even-fsname** - *Even-numbered nodes in filesystem 'fsname'*
 - **esfs-odd-fsname** - *Odd-numbered nodes in filesystem 'fsname'*
 - **esfs-failed-fsname** - *Failed nodes in filesystem 'fsname'*

- **Checks are performed on a category of nodes in parallel:**

- Power Status
- Node Status
- TCP Ping
- LNet Ping
- Lustre Mounts



Automated Lustre Failover Monitor - esfsmon

- **Failure Determination**
 - **Power Status**
 - Failure triggers a retry to avoid transient failed IPMI response
 - **Node Status**
 - Node DOWN status reported by management daemon
 - Must also fail a TCP ping before being declared “dead”
 - **TCP Ping**
 - Failure triggers a retry to avoid transient PING failure
 - **LNet Ping**
 - ibstat is checked - No active IB interfaces triggers failure
 - The ability to ‘lctl ping’ at least one other node must succeed
 - ‘lctl ping’ will retry for up to 90 seconds
 - **Lustre Mounts**
 - A missing mount triggers failover



Failover Logging – syslog on the CIMS

- **Syslog – Failover example (part 1 of 2)**

- /var/log/messages

```
Apr  1 19:07:24 esms1 logger: esms1 esfsmon: ERROR: esf-oss002
missing lustre mount: Failover initiated.
Apr  1 19:07:24 esms1 logger: esms1 esfsmon: esf-oss002 failed
health check
Apr  1 19:07:24 esms1 logger: esms1 esfsmon: In FAILOVER mode.
Apr  1 19:07:24 esms1 logger: esms1 esfsmon: Setting SUSPEND
mode for failover action (/var/esfsmon/esfsmon_suspend_scratch)
Apr  1 19:07:24 esms1 logger: esms1 esfsmon: Setting RUNSAFE
mode for failover action (/var/esfsmon/esfsmon_runsafe_scratch)
Apr  1 19:07:24 esms1 logger: esms1 esfsmon: OSS node failed:
esf-oss002
Apr  1 19:07:24 esms1 logger: esms1 esfsmon: Setting category
for esf-oss002 to esfs-failed-scratch
Apr  1 19:07:26 esms1 logger: esms1 esfsmon: Successfully set
esf-oss002 category to esfs-failed-scratch
```



Failover Logging – syslog on the CIMS

- **Syslog – Failover example (part 2 of 2)**
 - /var/log/messages

```
Apr  1 19:07:26 esms1 logger: esms1 esfsmon: Powering off esf-oss002 in 6 minutes, allowing kdump to complete
```

```
Apr  1 19:13:35 esms1 logger: esms1 esfsmon: esf-oss002 is powered off
```

```
Apr  1 19:13:35 esms1 logger: esms1 esfsmon: INFO: Failover started: esf-oss002's services failing over to partner
```

```
Apr  1 19:15:12 esms1 logger: esms1 esfsmon: INFO: Failover of targets completed successfully.
```

```
Apr  1 19:15:20 esms1 logger: esms1 esfsmon: INFO: Lustre targets started on failover partner
```

```
Apr  1 19:15:20 esms1 logger: esms1 esfsmon: INFO: I/O tuning of scratch complete.
```

```
Apr  1 19:15:20 esms1 logger: esms1 esfsmon: INFO: Failover completed
```



Failover Logging – fo_log on the CIMS

- **esfsmon Failover log (part 1 of 2)**

- /tmp/esfsmon/scratch.fo_log

```
Apr- 1-19:07:24 esfsmon: scratch: esf-oss002 failed health
check
Apr- 1-19:07:24 esfsmon: scratch: In FAILOVER mode.
Apr- 1-19:07:24 esfsmon: scratch: Setting SUSPEND mode for
failover action (/var/esfsmon/esfsmon_suspend_scratch)
Apr- 1-19:07:24 esfsmon: scratch: Setting RUNSAFE mode for
failover action (/var/esfsmon/esfsmon_runsafe_scratch)
Apr- 1-19:07:24 esfsmon: scratch: OSS node failed: esf-oss002
Apr- 1-19:07:24 esfsmon: scratch: Node ordinal=2
Apr- 1-19:07:24 esfsmon: scratch: Node is odd=0
Apr- 1-19:07:24 esfsmon: scratch: Node is EVEN
```



Failover Logging – fo_log on the CIMS

- **esfsmon Failover log (part 2 of 2)**
 - /tmp/esfsmon/scratch.fo_log

```
Apr- 1-19:07:24 esfsmon: scratch: Setting category for esf-oss002
to esfs-failed-scratch
Apr- 1-19:07:26 esfsmon: scratch: Successfully set esf-oss002
category to esfs-failed-scratch
Apr- 1-19:07:26 esfsmon: scratch: Powering off esf-oss002 in 6
minutes, allowing kdump to complete
Apr- 1-19:13:35 esfsmon: scratch: esf-oss002 is powered off
Apr- 1-19:13:35 esfsmon: scratch: INFO: Failover started: esf-
oss002's services failing over to partner
Apr- 1-19:15:12 esfsmon: scratch: INFO: Failover of targets
completed successfully.
Apr- 1-19:15:20 esfsmon: scratch: INFO: Lustre targets started on
failover partner
Apr- 1-19:15:20 esfsmon: scratch: INFO: I/O tuning of scratch
complete.
Apr- 1-19:15:20 esfsmon: scratch: INFO: Failover completed
```



Automated Lustre Failover Monitor – failback

- **Restoring a node back to service - `esfsmon_failback`**
 - Places a node back into service
 - `# esfsmon_failback lustre01-oss001`
 - Performs management housekeeping
 - Calls *lustre_control* to failback the Lustre assets to their primary server
- **NOTE:** Using '*lustre_control failback*' instead of '*esfsmon_failback*' will result in the node remaining in a “failed” category and no longer part of the health check.



Automated Lustre Failover Monitor – failback

- Restoring a node back to service - **esfsmon_failback**

```
Apr  1 18:37:50 esms1 root: esms1 esfsmon_failback: Setting esf-  
mds001 to esfs-odd-scratch category
```

```
Apr  1 18:37:51 esms1 root: esms1 esfsmon_failback: INFO: Failback  
started: failing back Lustre services to esf-mds001
```

```
Apr  1 18:39:08 esms1 root: esms1 esfsmon_failback: INFO: Failback  
completed successfully.
```

```
Apr  1 18:39:13 esms1 root: esms1 esfsmon_failback: INFO: Lustre  
targets started on esf-mds001
```

```
Apr  1 18:39:13 esms1 root: esms1 esfsmon_failback: INFO: I/O  
tuning of esf-mds001 not performed.
```

```
Apr  1 18:39:13 esms1 root: esms1 esfsmon_failback: INFO:  
Reloading Lustre modules on esf-mds002
```

```
Apr  1 18:40:14 esms1 root: esms1 esfsmon_failback: INFO: Failback  
of esf-mds001 completed
```




Automated Lustre Failover Monitor – esfsmon

- **Configuration File - [esfsmon.conf](#)**
 - Used by *esfsmon_healthcheck*, *esfsmon_action* and *esfsmon_failback*
 - `/cm/local/apps/cmd/etc/esfsmon.conf`
- **Includes the following parameters:**
 - State and Data directories
 - State: `/var/esfsmon`
 - Data: `/tmp/esfsmon`
 - Node categories used by each Lustre filesystem
 - LNet networks used by each filesystem
 - Base hostname for each filesystem
 - Paths to lustre_control tuning files
 - Passive MDS (non-DNE)



esfsmon Status - cmsh

- Check current status with “latesthealthdata” command in cmsh

```
esms1:~ # cmsh
[esms1]% device
[esms1->device]% latesthealthdata esms1 -v
```

Health Check	Severity	Value	Age (sec.)	Info Message
esfsmon:scratch	10	UNKNOWN	58	Monitor in SUSPEND mode.
esfsmon:scratch1	0	PASS	58	Monitor in ACTIVE mode.
esfsmon:scratch2	0	PASS	58	Monitor in RUNSAFE mode.
esfsmon:scratch3	0	PASS	58	Monitor in DNE ACTIVE mode.
esfsmon:scratch4	10	FAIL	58	ERROR: lustre04-oss002 failed LNET ping healthcheck: Failover initiated. Monitor in DNE ACTIVE mode.
esfsmon:scratch5	10	FAIL	58	ERROR: lustre05-oss010 failed TCP ping healthcheck: Cray intervention required. Monitor in RUNSAFE mode.
esfsmon:scratch6	0	PASS	58	WARN: Failover MDS lustre06-mds002 failed power healthcheck: Cray intervention required. Monitor in ACTIVE mode.

```
[esms1->device]%
```



esfsmon Status History - cmsh

- Check historical status with “dumphealthdata” command in cmsh

```
[esms1->device[esms1]]% dumphealthdata -7d now esfsmon:scratch -v
# From Wed Feb 26 11:24:09 2014 to Wed Mar 5 11:24:09 2014
Time                               Value                               Info Message
-----
Wed Feb 26 11:24:09 2014           PASS                               Monitor in ACTIVE mode.
Wed Feb 26 14:37:00 2014           PASS                               WARN: Failover MDS lustre01-mds002
failed LNET ping healthcheck: Cray
intervention required.
Monitor in ACTIVE mode.
Wed Feb 26 15:04:00 2014           FAIL                               ERROR: lustre01-mds001 missing
lustre mount: Cray intervention
required.
Monitor in RUNSAFE mode.
Wed Feb 26 15:05:00 2014           FAIL                               ERROR: lustre01-mds001 missing
lustre mount: Failover initiated.
Monitor in ACTIVE mode.
Wed Feb 26 15:06:18 2014           UNKNOWN                           Monitor in SUSPEND mode.
[esms1->device[esms1]]%
```



Lustre Monitoring Tool - LMT

- **Monitors Lustre servers using the Cerebro monitoring system**
 - Maintained by Lawrence Livermore National Laboratory (LLNL)
 - Collects statistics published in `/proc/fs/lustre` every 5 seconds.
- **Enabled/Disabled by starting/stopping the Cerebro daemon (cerebrod) on the CIMS and CLFS nodes.**
- **On the CIMS:**
 - **Imt-server:** Cerebro monitoring system
 - Cerebro uses the MySQL server on the CIMS to store data
 - **ltop:** Live top-like data monitor
 - **Imtsh:** interactive shell for viewing historical LMT data
- **On the CLFS (MDS/OSS) nodes:**
 - **Imt-server-agent:** Cerebro monitor plugin, *ltop* client and other utilities for administering LMT



Lustre Monitoring Tool - LMT OSS/OST Data

- **Data Collected**

- **OSS/OST**

- OSC Status - The MDS's view of the OST
- OSS hostname
- Export Count - number of clients mounting the filesystem
 - Includes one for the MDS and one for the OST itself
- Reconnects per second
- Read and Write rates
- Bulk RPCs per second
- OST resource locks
 - number currently granted, grant and cancellation rate
- CPU and Memory usage
- OST storage space used



Lustre Monitoring Tool - LMT MDS/MDT Data

- **Data Collected**
 - **MDS/MDT**
 - CPU usage
 - KB free
 - KB used
 - inodes free
 - inodes used
 - Rates for the following operations:
 - open and close
 - mknod
 - link and unlink
 - mkdir and rmdir
 - rename



Lustre Monitoring Tool - LMT

- Example ltop data:

```
/usr/bin/ltop -f scratch
```

```
Filesystem: scratch
```

```
Indoors:    443.956m total,    49.295m used ( 11%),    394.662m free
Space:      172.188t total,    129.573t used ( 75%),    42.615t free
Bytes/s:    0.000g read,          0.000g write,          0 IOPS
MDops/s:    0 open,              0 close,                0 getattr,            0 setattr
            0 link,              0 unlink,               0 mkdir,              0 rmdir
            0 statfs,            0 rename,               0 getxattr
```

OST	S	OSS	Exp	CR	rMB/s	wMB/s	IOPS	LOCKS	LGR	LCR	%cpu	%mem	%spc
0000	F	oss001	137	0	0	0	0	0	0	0	1	10	77
0001	F	oss002	137	0	0	0	0	0	0	0	0	9	76
0002	F	oss003	137	0	0	0	0	0	0	0	0	9	76
0003	F	oss004	137	0	0	0	0	0	0	0	0	10	76
0004	F	oss001	137	0	0	0	0	0	0	0	1	10	76
0005	F	oss002	137	0	0	0	0	0	0	0	0	9	76
0006	F	oss003	137	0	0	0	0	0	0	0	0	9	76
0007	F	oss004	137	0	0	0	0	0	0	0	0	10	77



Lustre Monitoring Tool - LMTSH

- Example lmtsh data: (part 1 of 2)

```
/usr/bin/lmtsh -f scratch
```

```
scratch> fs
```

```
Available filesystems:
```

```
    scratch
```

```
scratch> ost
```

OST_ID	OST_NAME
1	scratch-OST0000
2	scratch-OST0002
3	scratch-OST0004
4	scratch-OST0006
5	scratch-OST0001
6	scratch-OST0003
7	scratch-OST0005
8	scratch-OST0007

Lustre Monitoring Tool - LMTSH

• Example lmtsh data: (part 2 of 2)

```
/usr/bin/lmtsh -f
scratch
```

```
scratch> t
```

```
Available tables for scratch:
```

Table Name	Row Count
------------	-----------

EVENT_DATA	0
------------	---

EVENT_INFO	0
------------	---

FILESYSTEM_AGGREGATE_DAY	576
--------------------------	-----

FILESYSTEM_AGGREGATE_HOUR	13338
---------------------------	-------

FILESYSTEM_AGGREGATE_MONTH	36
----------------------------	----

FILESYSTEM_AGGREGATE_WEEK	99
---------------------------	----

FILESYSTEM_AGGREGATE_YEAR	18
---------------------------	----

FILESYSTEM_INFO	1
-----------------	---

MDS_AGGREGATE_DAY	640
-------------------	-----

MDS_AGGREGATE_HOUR	14745
--------------------	-------

MDS_AGGREGATE_MONTH	40	OST_AGGREGATE_WEEK	792
MDS_AGGREGATE_WEEK	110	OST_AGGREGATE_YEAR	144
MDS_AGGREGATE_YEAR	20	OST_DATA	8465265
MDS_DATA	2108108	OST_INFO	8
MDS_INFO	2	OST_OPS_DATA	0
MDS_OPS_DATA	44273061	OST_VARIABLE_INFO	11
MDS_VARIABLE_INFO	7	ROUTER_AGGREGATE_DAY	0
OPERATION_INFO	81	ROUTER_AGGREGATE_HOUR	0
OSS_DATA	2101471	ROUTER_AGGREGATE_MONTH	0
OSS_INFO	2	ROUTER_AGGREGATE_WEEK	0
OSS_INTERFACE_DATA	0	ROUTER_AGGREGATE_YEAR	0
OSS_INTERFACE_INFO	0	ROUTER_DATA	0
OSS_VARIABLE_INFO	7	ROUTER_INFO	0
OST_AGGREGATE_DAY	4608	ROUTER_VARIABLE_INFO	4
OST_AGGREGATE_HOUR	106596	TIMESTAMP_INFO	1131417
OST_AGGREGATE_MONTH	288	VERSION	0



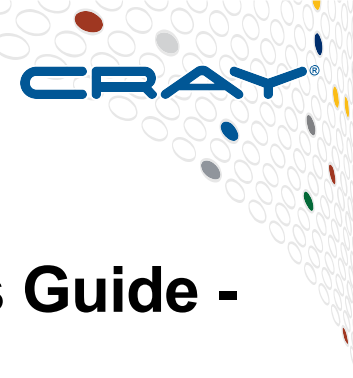
Log Files – located on the CIMS

- **Syslog** – All CLFS nodes forward syslog to the CIMS
 - `/var/log/messages`
- **esfsmon Failover log**
 - `/tmp/esfsmon/fsname.fo_log`
- **CMDaemon** – Management daemon log
 - `/var/log/cmdaemon`
- **Node-installer** – Node Installer log
 - `/var/log/node-installer`
- **Conman** – CLFS Node console logs
 - `/var/log/conman/`
- **Software Installation Logs**
 - `/var/adm/cray/logs`
- **Bright Cluster Manager Event Log**
 - Stored in the Bright Cluster Manager database
 - Accessed via “events” command in cmsh or event viewer in cmgui



Troubleshooting

- **Depending on the problem, check the most likely logs**
 - **Booting issues**
 - node-installer log - `/var/log/node-installer`
 - **Management issues**
 - cmdaemon log - `/var/log/cmdaemon`
 - syslog - `/var/log/messages`
 - event log
 - 'events' command in cmsh
 - 'event viewer' in cmgui
 - Check relevant monitor histories
 - 'dumphealthdata' for health checks
 - **Operational issues**
 - syslog - `/var/log/messages`
 - event log
 - 'events' command in cmsh
 - 'event viewer' in cmgui
 - Check relevant monitor histories
 - 'dumphealthdata' for health checks



Documentation

- **Data Management Platform (DMP) Administrator's Guide - S-2327-C**
 - esfsmon installation and configuration
 - LMT installation and configuration
- **Installing Lustre(R) File System by Cray(R) (CLFS) Software - S-2521-C**
- **Installing Cray(R) Integrated Management Services (CIMS) Software - S-2522-E**
- **LMT - <https://github.com/chaos/lmt/wiki>**



Thank you for your time!

Any questions?

Jeff Keopp
Harold Longley

Legal Disclaimer



Information in this document is provided in connection with Cray Inc. products. No license, express or implied, to any intellectual property rights is granted by this document.

Cray Inc. may make changes to specifications and product descriptions at any time, without notice.

All products, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.

Cray hardware and software products 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.

Cray uses codenames internally to identify products that are in development and not yet publically announced for release. Customers and other third parties are not authorized by Cray Inc. to use codenames in advertising, promotion or marketing and any use of Cray Inc. internal codenames is at the sole risk of the user.

Performance tests and ratings are measured using specific systems and/or components and reflect the approximate performance of Cray Inc. products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance.

The following are trademarks of Cray Inc. and are registered in the United States and other countries: CRAY and design, SONEXION, URIKA, and YARCDATA. The following are trademarks of Cray Inc.: ACE, APPRENTICE2, CHAPEL, CLUSTER CONNECT, CRAYPAT, CRAYPORT, ECOPHLEX, LIBSCI, NODEKARE, THREADSTORM. The following system family marks, and associated model number marks, are trademarks of Cray Inc.: CS, CX, XC, XE, XK, XMT, and XT. The registered trademark LINUX is used pursuant to a sublicense from LMI, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis. Other trademarks used in this document are the property of their respective owners.