

F. Chlebana
Dec 8 2006

Status of the CSL Upgrade

I. Bizjak, F.Chlebana, G. Guglielmo, (J.Lee), K.McFarland,
W. Sakumoto, R.Sarkis, JJ.Schmidt, M.Shimajima,
R.Snider, G.Yu, D.Zhang, DataComm, Computing Division

Progress Since Last Time

- *New CSL in use since Oct 31*
- *Refined Operational Procedures*
- *Monitoring Improvements*
- *Orphaned File Checking*
- *Documentation and Operating Instructions Updated*
- *Tested System Failure Modes*
- *Support Issues*
- *Remaining Tasks*

Started using the new CSL on Oct 31 2006 (Store 5052)

Since then we lost 0.9% (1 pb) of data due to CSL related issues

- 438/1040 nb CSL testing while code developer is at the lab
→ *Requires starting stopping runs*
- 189/1040 nb Logging directory removed
→ *Reverted back to SGI CSL during diagnosis*
- 100/1040 nb COT data corruption lead L3 filters crashing and CSL stopped processing data
→ *Restart procedure failed because monitoring was holding shared memory segments*
- 93/1040 nb CSL logger message queue was full and stopped we handling data
→ *Restart procedure took a long time*

Main source of lost data associated with testing

Rochester code developer was visiting and we wanted to use beam time in order to debug problems while he was present

Another significant source of lost data is slow recovery time

Have a couple of failure modes which require restarting the CSL

- Corrupt data, L3 crashes and Output queue fills up
→ *At that point data taking is halted*
- Logger Message Queue Full
→ *Events not removed from logger queue and run is halted*
- Max Client Connections
→ *Problem shows up when starting run*

Working on fixing the source of the problem...

Developed a more robust and quicker restart procedure

→ *We had an issue with a monitor process that was holding onto shared memory segments which prevented us from completing the startup of the CSL software*

→ *Script should now handle this situation....*

```
b0csl20> restartCSL
```

wait for the banner indicating that the reset is complete as below:

```
#####  
##                ##  
##  CSL Restart Complete  ##  
##                ##  
#####
```

After restarting the CSL, clean up L3 Farm.

Could also do the recovery in parallel to speed it up...

Hardware Performance and Robustness

This was one of my biggest worries...

→ *Have not seen any hardware related issues since we finalized the configuration*

→ *Stressed the system extensively far beyond what we expect during normal running...*

→ *Exceeded our goal of a sustained logging rate of 80 MB/s*

Spare Logger Node Configuration

In order to simplify recovery from a failed logger we connected b0csl21 directly to the dotHill disk array (used in the prototype).

→ *b0csl21 will always be available - no need for system level reconfiguration, simplifies disaster recovery...*

→ *Can use b0csl30 and b0csl21 as our development platform*

Data Sorting Tests

The CSL does not touch the data, it directs the data to the designated output stream and to the consumers based on the *event header* that is passed to the CSL from L3.

Repeated the same data sorting tests as done with the old CSL.

The data sorting test uses a software sender process (SA_sender) that plays a known output pattern through the CSL.

We then verify that the expected number of events end up in the designated stream.

→ We also checked offline data files to make sure that the number of events in the files matched what was expected

We are doing as good as we did in the old system...

Orphaned File Checking

We want to make sure that we do not have files left over on the disk buffer that are not accounted for.

1) Files are being kept track of on the “CSL Fetch” web page

Requires a visual scan...

2) Implemented a more automated “file list based” check.

→ *Keep list of files when first written to disk by the CSL.*

→ *Separate list of files written to tape produced by the stager.*

→ *Compare the two lists to check for any discrepancies.*

Files that are expected to be written to tape:

→ *All files marked as “phys”*

→ *Any run marked by RC to be sent to tape*

Operational Problems

- Have a couple of problems requiring restarting the run

Working on fixing the problems...

→ Developed a robust and fast recovery script
Could be made faster by running in parallel

- Orphaned Files, not closed and left in the .open area

→ *Have not seen this problem after recent code update*

- We sometimes have corrupt db files.

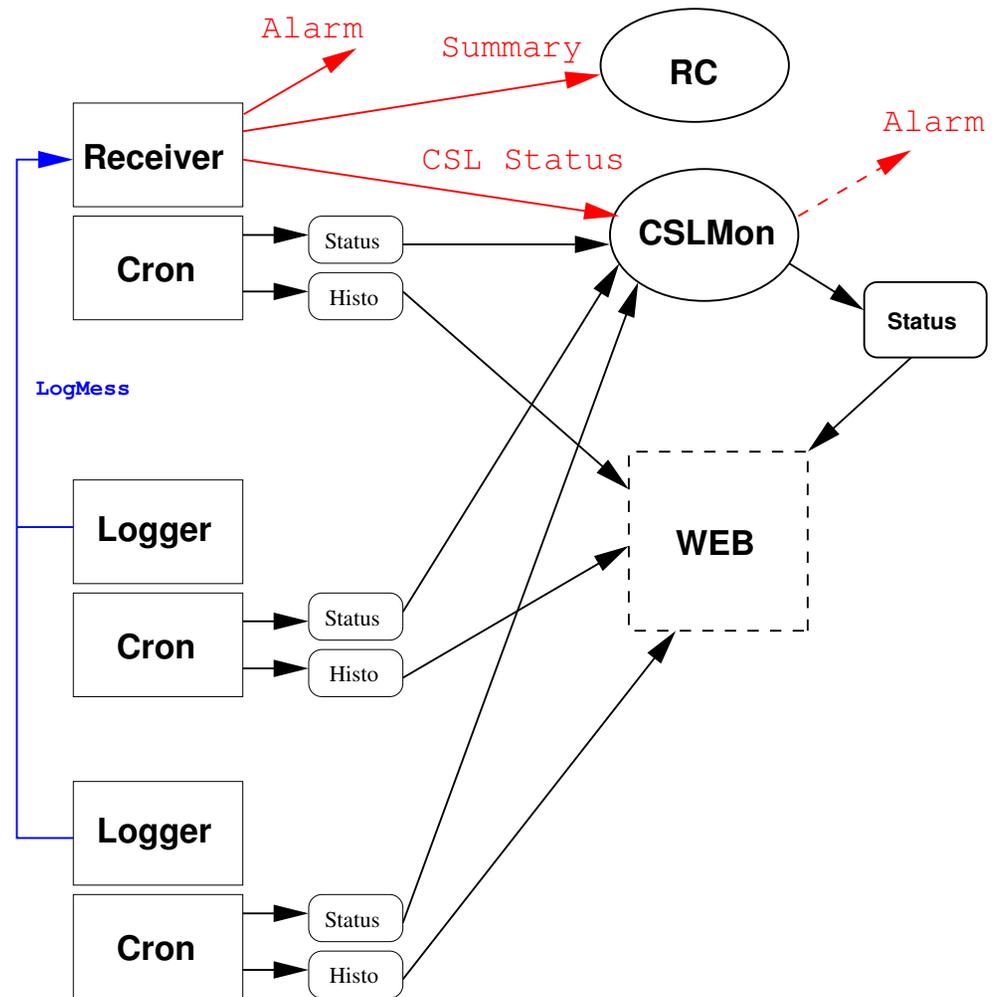
→ Developed script to reconstruct the db file from the data.
Move this by hand to an output area so the stager can pick it up
and copy to Feynman.

Monitoring

Same monitoring information that was available in the old system is available in the new CSLMon.

History monitoring plots available on the web

- Using rrdtools to generate plots
- Using ganglia for system monitoring



We have enough information to be able to monitor the performance of the CSL.

“CSL To Enstore Status” web page provides an excellent overview

The NEW CSL ACE Help Pages - SeaMonkey

Overview Operations Calibration CSL Trouble Shooting Expert Pages Contacts Monitoring

CSL To Enstore Status - Fri Dec 8 08:30:38 2006

SAM DB Server (rawdata_prd) [Alive](#) Lookarea %Full (LastWeek / All) [91 \(365GB / 1058GB\)](#)
 Quota Home / Data / Spool - sam (%) [38 / 15 / 0](#) Pending Store / Still Open [5261 / 102](#)
 Quota Home / Data / Spool - stager (%) [23 / 47 / 0](#) [Expert FAQ](#)

Node	Updated	FSS	Lun	CSL			Stager			In FSS	Recent Stores	Clam	Clmeta	Lookarea		
				Open	To Store	Error	New	Queued	Submitted		No LastDone	W / E / A HeartBeat	W / E HeartBeat	New	Recent Copies	Error
b0csl21	1208 08:30	Alive	1	94	0	0	0	0	0	0	0	0 / 0 / 0	0 / 0 1208 08:28	0	0	0
			2	0	0	0	0	0	0	0	0	0	0 / 0 1208 08:29	0	0	0
b0csl22	1208 08:30	Alive	1	1	5	4	0 1207 21:59	0 / 0 / 0 1208 08:29	0 / 0 1208 08:29	0	0 1208 08:24	0				
			2	0	0	0	0	0	0	0	0	0 1129 08:11	0 / 0 1208 08:29	0 / 0 1208 08:29	0	0 1129 07:49
b0csl23	1208 08:30	Alive	1	1	14	1	0 1207 20:25	0 / 0 / 0 1208 08:29	0 / 0 1208 08:29	0	1 1208 08:25	0				
			2	0	0	0	0	0	0	0	0	0 0906 20:39	0 / 0 1208 08:29	0 / 0 1208 08:29	0	0

Recovery from Hardware Failure

The system is very redundant, expect to be able to quickly recover from hardware failures, recovery procedures are in place...

Disk Array (failed disk, controller, chassis)

RAID automatically rebuilds

Path to secondary LUN would still be available

Logger

Switch to hot spare (b0csl21)

Receiver

Switch to hot spare (b0csl30)

SANBox

Can withstand single box failure

GigE Switch

Move to hot spare

Tested switching to secondary LUN when primary LUN is full

Code Management

Improving code management

→ *Source code is tagged in cvs*

→ *We want to use the UPS environment to ensure better control over versions and allow us to quickly change between versions.*

First steps have been taken...

Other Issues

Details at:

<http://ncdf76.fnal.gov/~chlebana/daq/cslUpgrade/commissioning/InitialRunning.html>

Run Section Table

DFC script fills data base with min/max run number for each runsection. Need to adapt old script to new environment and fill missing content for runs already taken

Close to being finished

Orphaned File Check

Added more automated checking

CSL logger message queue fills up

Debugging... developed fast recovery procedure

MAX Client connection exceeded

Debugging... increased the max number of connections allowed

db files corrupted

Debugging... can recover db file by running script

Files left on the .open area

Possibly fixed...

Automatic File Cleanup

Added cron job to clean up log files and delete old files in the "to_be_deleted" area

.open directory deleted

Should not normally happen. Added additional checks for this condition

Calibration CSL

Does not listen to RC messages. Requires having Calibration CSL to subscribe to a unique Smart Socket subject name

Cannot handle Bad Events

Debugging... developed fast recovery procedure

Testing fix...

PROCMon check failing

Increased timeout and distinguish between real failure and if PROCMon does not complete the check

Code Management

Setting up UPS environment for code management

Silicon ISL threshold scan

For now focusing on operational issues...

Data File Catalog

When moving to SAM we stopped writing to DFC

The min/max event number per runsection is not saved in SAM

This breaks offline code which relies on this information

Need to resurrect the script to fill in this table

Should be straight forward...

→ *We have all the information in “db” files....*

→ *Need to move the db files to a designated area*

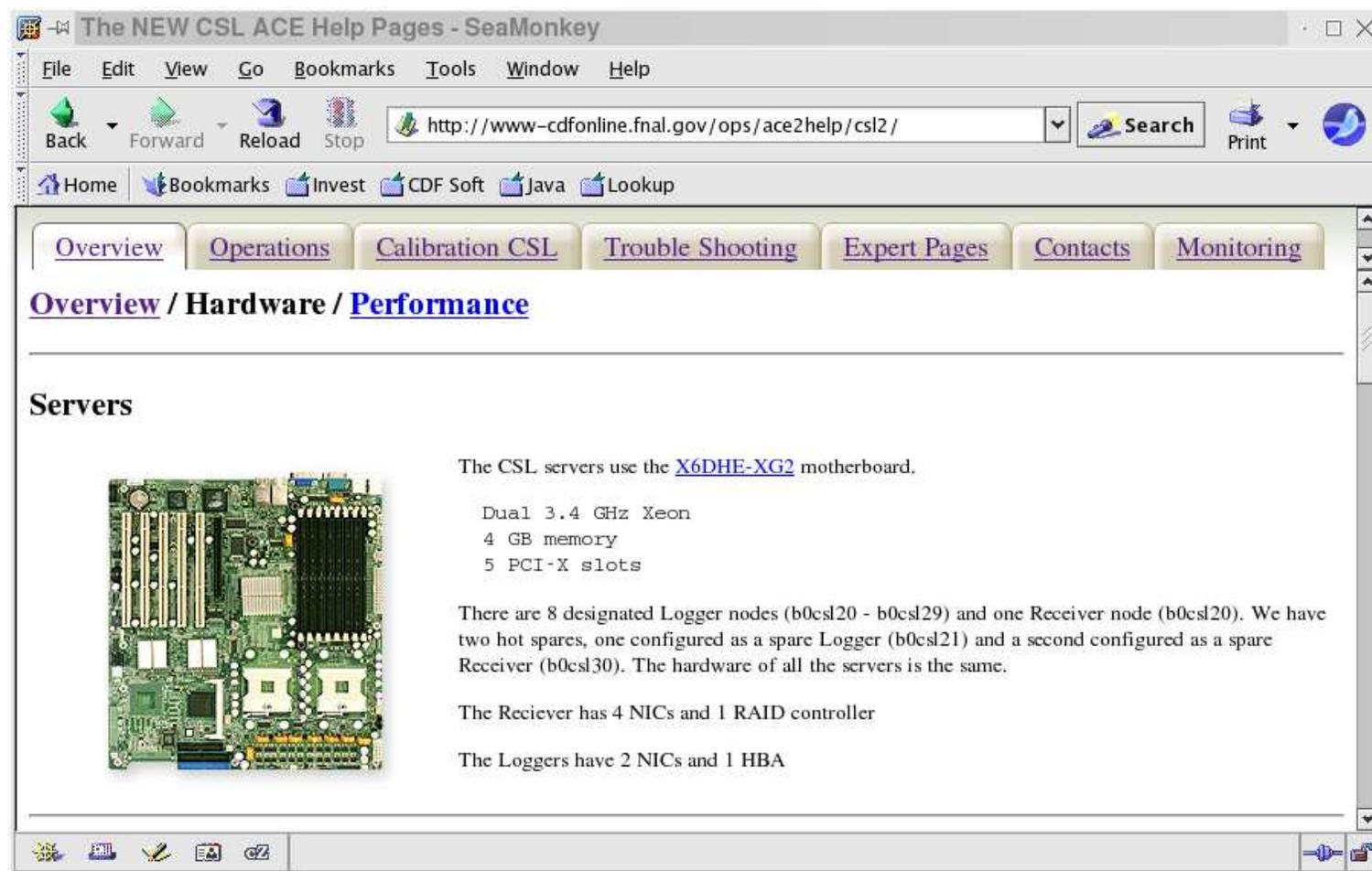
→ *Update the script to run under Linux (nearly done)*

→ *Automate moving of the files to the designated area*

→ *Put back in the checks (in PROCMon) to ensure the script is running*

Documentation on the *interwebs*

We updated operating instructions and “Expert Pages” with shift instructions as well as notes for ourselves.



The NEW CSL ACE Help Pages - SeaMonkey

File Edit View Go Bookmarks Tools Window Help

Back Forward Reload Stop <http://www-cdfonline.fnal.gov/ops/ace2help/csl2/> Search Print

Home Bookmarks Invest CDF Soft Java Lookup

[Overview](#) [Operations](#) [Calibration CSL](#) [Trouble Shooting](#) [Expert Pages](#) [Contacts](#) [Monitoring](#)

[Overview](#) / [Hardware](#) / [Performance](#)

Servers



The CSL servers use the [X6DHE-XG2](#) motherboard.

- Dual 3.4 GHz Xeon
- 4 GB memory
- 5 PCI-X slots

There are 8 designated Logger nodes (b0csl20 - b0csl29) and one Receiver node (b0csl20). We have two hot spares, one configured as a spare Logger (b0csl21) and a second configured as a spare Receiver (b0csl30). The hardware of all the servers is the same.

The Receiver has 4 NICs and 1 RAID controller

The Loggers have 2 NICs and 1 HBA

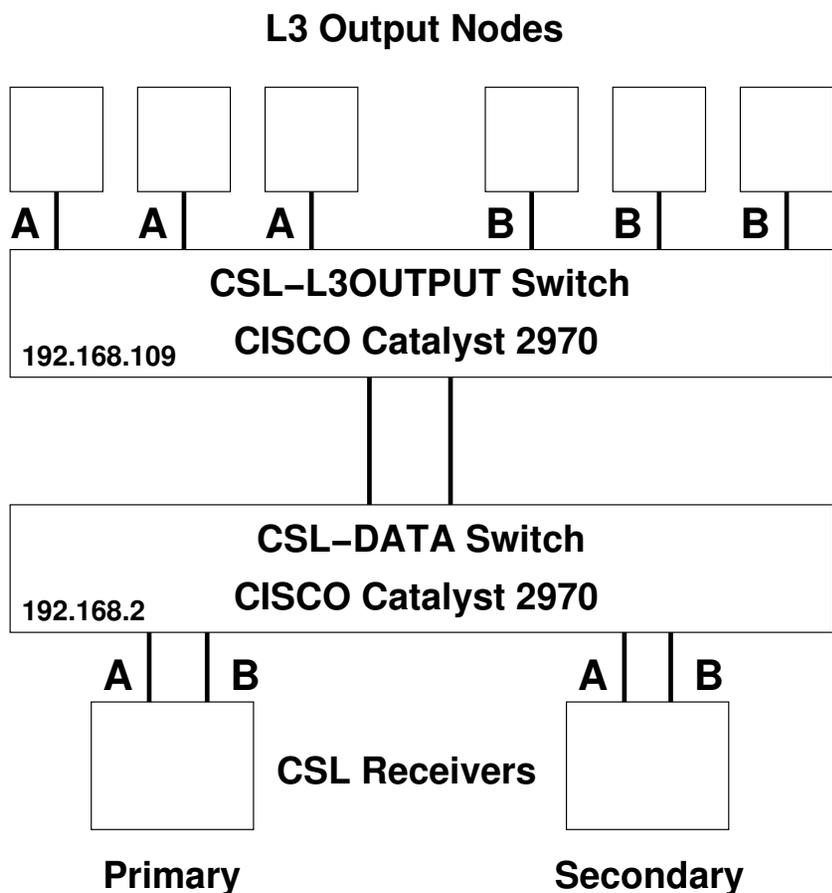
Some Additional Development

There are a couple of **non-critical** outstanding issues:

- Want to get “ether channeling” to work between L3 Output and CSL Data switch, *boosts bandwidth from 120 to 240 MB/s.*
- Install memory on b0csl26, *have 2 GB want 4 GB*
- Eventually write data to the new tape robot *In 2007...*

Tape robots in GCC are being burned in with MC data

EVB Output/CSL Data Switch



We are planning on using ether channeling on two links between the CSL-L3Output switch and the CSL-Data switch. This will distribute the load across the L3 output nodes and gives us an effective bandwidth of 240 MB/s.

Currently only one link is being utilized...

Support Issues

System Redundancy and Hot Spares

Avoid having a single point of failure

- *Multiple fibre paths*
- *RAIDed boot disk*
- *Fully configured hot spares*
- *Available spare parts*

Self repairing, receiver can redirect traffic to free logger/filesystem

System can be quickly reconfigured to bypass failed component

Deep buffers (3 days at 80 MB/s) allow plenty of time to react to down stream problems

Expect most problems can be handled by the CSL pager carrier

Long Term Support Issues

Rochester

- 1 Onsite Graduate Student
- 1 Remote Computer Professional (part time)

Tsukuba

- 1 Onsite Graduate Student

University College London

- 1 Remote PostDoc (part time)

Computing Division

System hardware support CD/Rex

Stager/Logger

Data handling group

Not sufficient to provide ongoing support

What we need:

SPL

Main contact to address operational issues

Monitor performance/elogs for potential problems

Coordinate activities

CSL pager carriers

Should have at least 3

System support

CD/Rex - *I think this should be ok...*

CSL Software support

No resident expert

Will be difficult to get quick turn around to address problems

→ *Must be available in case of urgent problems*

→ *Difficult to implement new requests (ISL scan)*

Monitoring software support

No resident expert

Will be difficult to get quick turn around to address problems

→ *I expect that this will be the place we will continued development as we gain additional operational experience*

Stager/Logger pager coverage

Will need point of contact

Summary

The new CSL is being used in production

Hardware Performing Well and is Robust

- Can operate well above the target of 80 MB/s
- Redundant system, quick recovery from hardware failures

Monitoring framework in place and provide enough information to understand performance issues

<http://www-cdfonline.fnal.gov/ops/ace2help/csl2/>

→ *Monitoring tab*

This is an area that would benefit from ongoing improvements

Documentation and Operating Procedures are in place

Orphaned File Checking

→ *Automating the check*

A Few Remaining Operational Issue

→ *Handle bad events*

→ *Truncated db file*

→ *Data File Catalogue*

Support Issues

→ *No resident software experts*

→ *Need to develop resident expertise for the monitoring software*

→ *Need at least 3 people for pager support*

Remaining Tasks

→ *Couple of non-critical tasks that need to be finished up.*

Architecture Overview

