LHConCRAY

Acceptance Tests 2017 – Run4 System Report

Miguel Gila, CSCS

August 03, 2017

Table of Contents

cscs

ETHzürich

Changes since Run2/3

Problems with DVS

- ## DVS load
  - At times load on DVS would shift from being almost balanced to target a single node
  - Cray identified this to be a GPFS problem (being unresponsive).

- ## DVS hung and XFS filesystems on files crashing
  - When a DVS server is unavailable, CNs use another one after a non-configurable timeout
  - However, when an XFS filesystem is mounted off the failed DVS server, caching and open descriptors often corrupt the xfs filesystem
  - This is, at times can go unnoticed and cvmfs continue serving files with incorrect checksums

- ## These two problems have affected us in the evaluated period at least 2 times
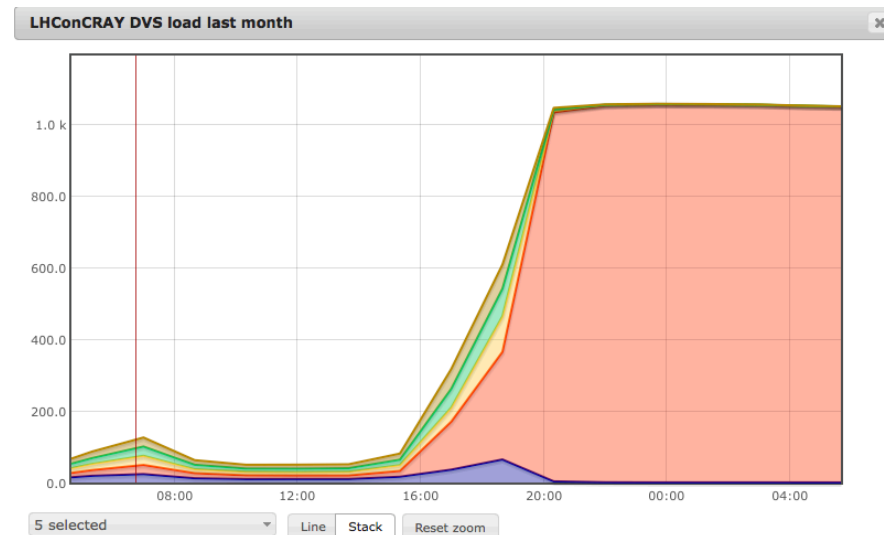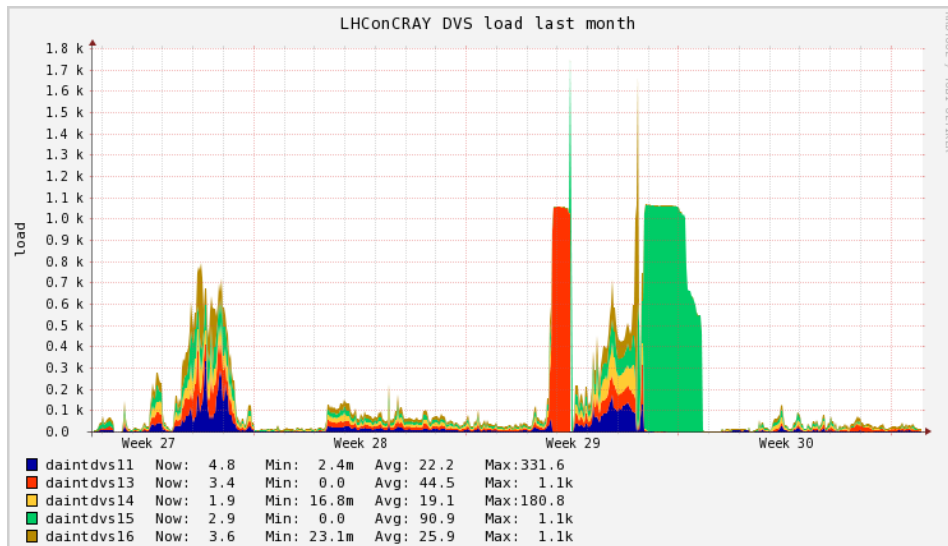
Changes due to DVS issues 1/2 (CVMFS)

- CVMFS was originally configured to use a XFS filesystem on top of DVS as local cache

- However, CVMFS developers created two new features that allow us to store data directly onto DVS projected filesystems (no more XFS!):

    - Workspace: since DVS does not support `flock()`, with the workspace setting it is now possible to set all locks relative to the cache local to the node (on ramdisk)

    - Tiered cache with in-ram storage: with enough RAM, it's now possible to instruct cvmfs to store its cache on memory, without the need for local storage. This can dramatically increase performance

    - In order to reduce repetitive queries outside, all relevant cvmfs repos have been preloaded on GPFS thanks to a new, fast service provided by CERN for HPC sites. This gets updated several times a day. Note that cvmfs is smart enough to query outside if the file is not on the preloaded cache

- These two changes were applied on all 25 CNs on **July 19**
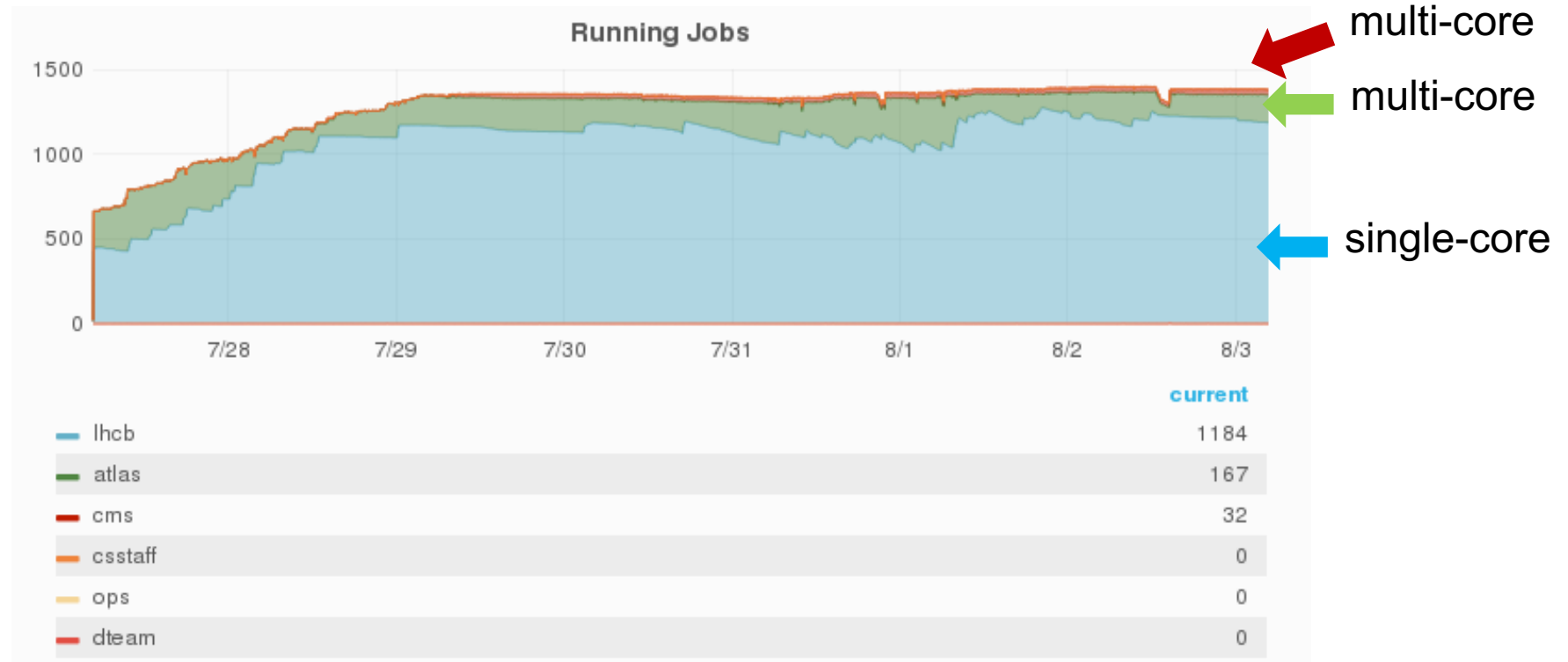
cscs

ETHzürich

Changes due to DVS issues 2/2 (DVS itself)

- As it was configured earlier, DVS client would calculate the hash of a file (or a block) and pick a DVS server of the list of available nodes.

- As reported, at times IO would shift and target just a single DVS node. It could be a group of jobs doing a lot of metadata operations on blocks with the same hash, or on similar paths.

- In order to reduce the chances of this happening, we forced the this hash to be calculated per node, so each CN will target one DVS no matter what file/block it is trying to use.

- As a result, we can see load peaks on CNs, but not on DVS servers.

- This was introduced to all CNs on **July 25.**

cscs

ETH zürich
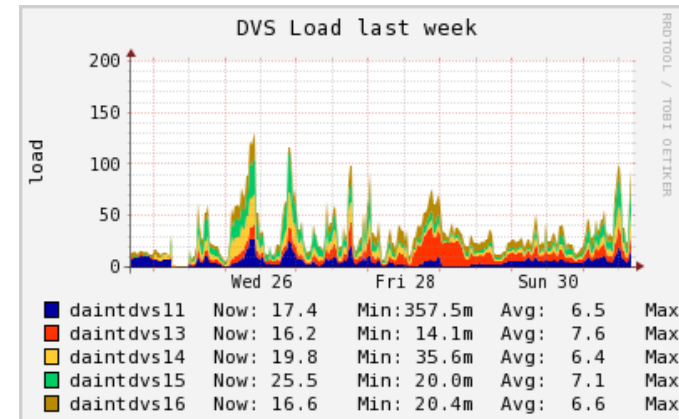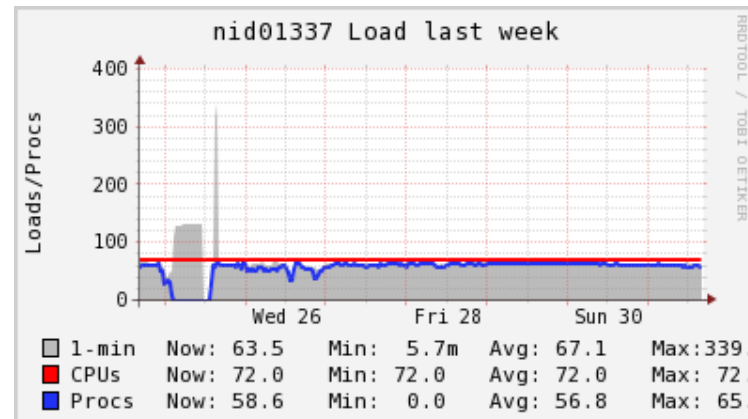
CMS started running production jobs

- On July 28 CMS started running almost production jobs (empty payloads)

- On July 31 CMS started running real production jobs
  - About 30 jobs are in the system at all times

Running Jobs

	current
lhcb	1184
atlas	167
cms	32
csstaff	0
ops	0
dteam	0

multi-core

multi-core

single-core

CSCS

ETH *zürich*

Public monitoring

■ (still) Working on a proper dashboard

■ For now a Twiki page with public charts is enabled on
https://wiki.chipp.ch/twiki/bin/view/LCGTier2/LHConCRAYMonitoring

  ▪ Links to VO monitoring pages
  ▪ Selectable time window
  ▪ Notably missing per-VO CPU allocation statistics.

■ Instant view (SLTOP) available on
http://ganglia.lcg.cscs.ch/ganglia/sltop_lhconcray.html

DataWarp

Concepts

- DataWarp is a Cray product that exposes very fast SSD-based storage to CNs

- Three main use-cases
  - Swap space
  - Scratch filesystem
  - ~~Cache (Lustre, not implemented)~~

- DWS activations can be done
  - On-demand per-job (1job/node)
  - Static, set up by an admin

- In our context, static DWS activations can be potentially used for
  - Swap
  - CVMFS
  - Scratch
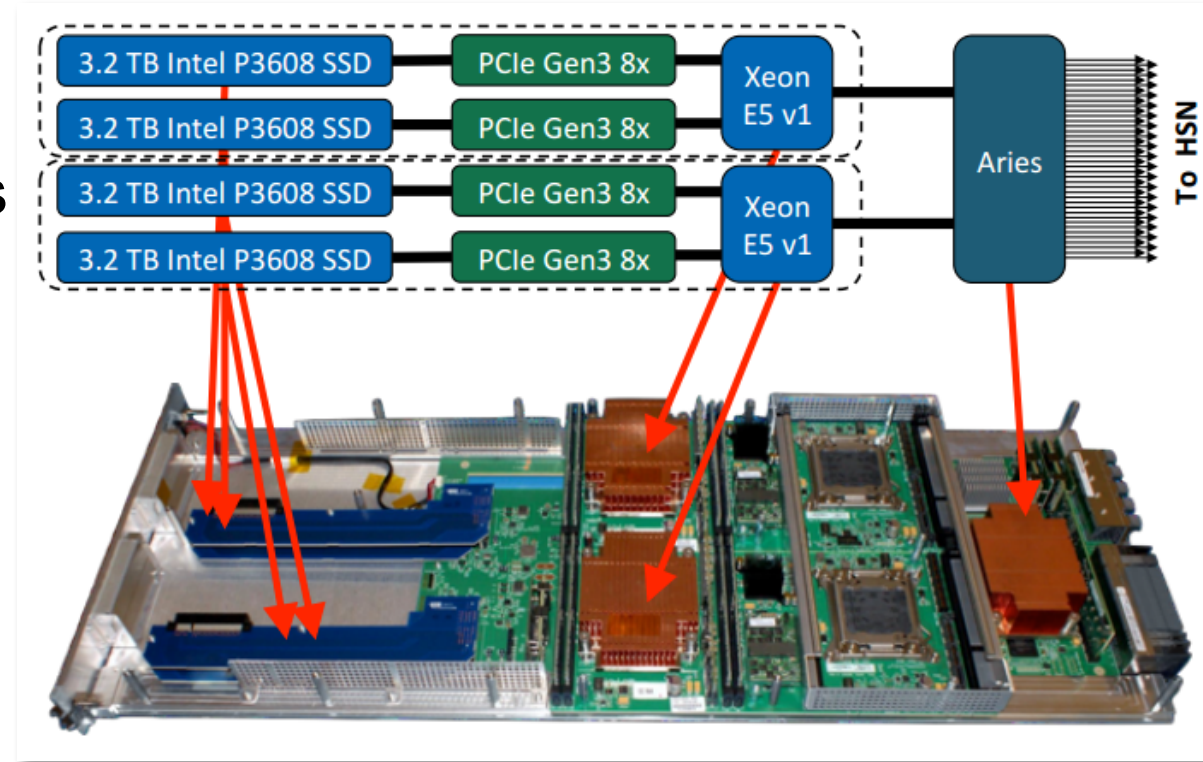
Test bed

- 4 DataWarp nodes dedicated for LHConCRAY
  - 5.82TiB x 4 = 23.28TiB

- Static DWS activations were created on June 30
  - SWAP: 64 GB per node, 1.56TiB total
  - CVMFS: /dws/cvmfs (shared across all nodes), 6.25TiB total
  - SCRATCH: /var/opt/cray/dws/lhc_scratch (shared across al nodes), 15TiB total

cscs

ETH *zürich*

Test bed - Swap

- Swap was activated on Jun 19

- One iSCSI device per node with 64 GB each
  - Reduced to 16GiB at the moment, more on this later
- No issues so far!

```
[544547.040890] iscsid[45170]: iscsid: Login failed to authenticate with target iqn.1988-11.com.cray.datawarp:sn80
[544547.076287] scsi 16:0:0:0: Direct-Access     LIO-ORG  FILEIO           4.0  PQ: 0 ANSI: 5
[544547.089719] sd 16:0:0:0: [sdb] 134217728 512-byte logical blocks: (68.7 GB/64.0 GiB)
[544547.102387] sd 16:0:0:0: [sdb] Write Protect is off
[544547.108990] sd 16:0:0:0: [sdb] Mode Sense: 43 00 00 08
[544547.109164] sd 16:0:0:0: [sdb] Write cache: disabled, read cache: enabled, doesn't support DPO or FUA
[544547.124060]  sdb: unknown partition table
[544547.129938] sd 16:0:0:0: [sdb] Attached SCSI disk
```

```
|nid01334:~ # free -h
|              total        used        free      shared     buffers      cached
|Mem:          126G        105G         20G        465M        1.0G         31G
|-/+ buffers/cache:         72G         53G
|Swap:          15G        708K         15G
```

cscs

ETH zürich

Test bed - CVMFS

- Tested on a single node (nid01357)

- Single mountpoint shared across all nodes for maximum performance and reduced data duplication

- Leveraging CVMFS tiered cache structure

- Early results showed good performance

- But after deeper testing we detected transparent **data corruption**
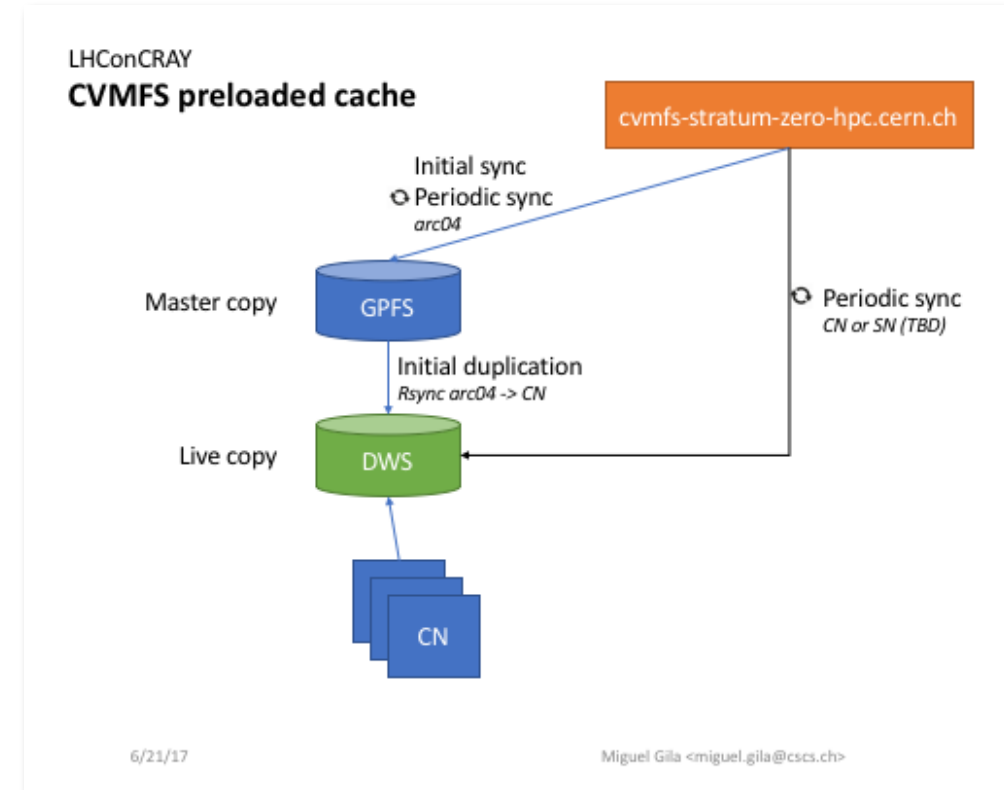
Test bed - Scratch

- Tested on a single node (nid01357)
- Single mountpoint shared across all nodes. Easy to setup, limited space distributed
- Using ARC's RUNTIME_LOCAL_SCRATCH_DIR variable that copies sessiondir to DWS
- Early results showed good performance
- **Data corruption** was detected with ATLAS jobs (2 files with same filename=data corruption)
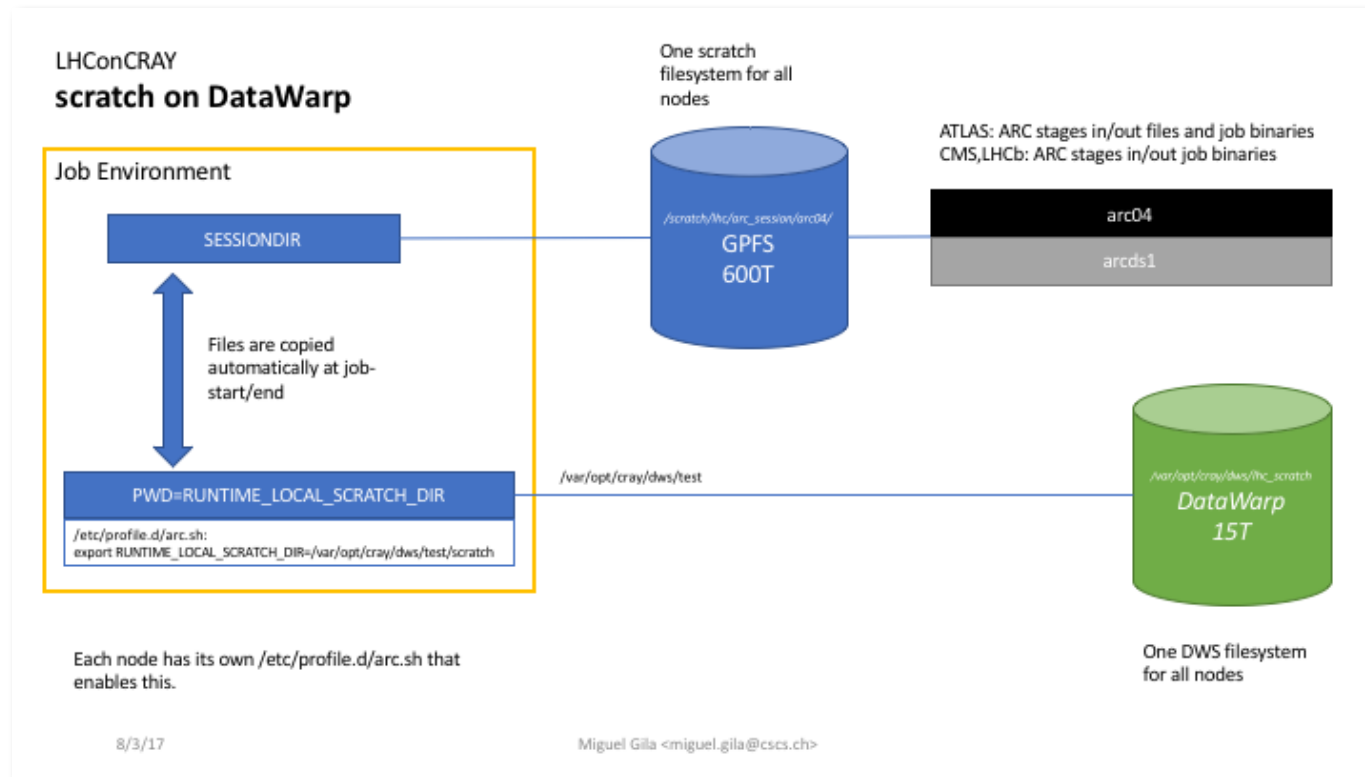
A possible solution

- Cray detected the corruption problem and issued a Field Notice and a Patch:

- FN #6178a - DataWarp potential data corruption in CLE

- Corruption does not affect swap (they're iSCSI devices)

- Patch PS56 released and installed on Piz Daint

- Images rebuilt on 31.07.2017

- DWS server nodes for LHConCRAY rebooted with new image

- Currently **having problems activating swap** using these nodes
  - Added 16GB of SWAP from the rest of the DWS server nodes as a temporary measure

- Testing continues

cscs

ETHzürich

Current configuration

Current configuration

- 25 compute nodes (72core, 128GB RAM, diskless)

- 1 production ARC + 1 data staging ARC + 1 test ARC (internal)

- Dedicated GPFS filesystem shared with Phoenix

- 5 DVS nodes exposing GPFS to CNs

- CVMFS running natively on CNs using tiered cache and workspaces
  - Upper layer: 6 GB in-RAM shared to all experiments
  - Lower layer: preloaded cache on GPFS, mount on CNs RO with caching enabled

- Memory limits NOT really enforced
  - Hard limit of 6000MB/core to catch rogue jobs

- Swap on DataWarp enabled (~~64GB~~ 16GB per node)

- ARC caching not enabled (ATLAS)
  - Each job has a copy of the files, even if they're the same on multiple jobs.

cscs
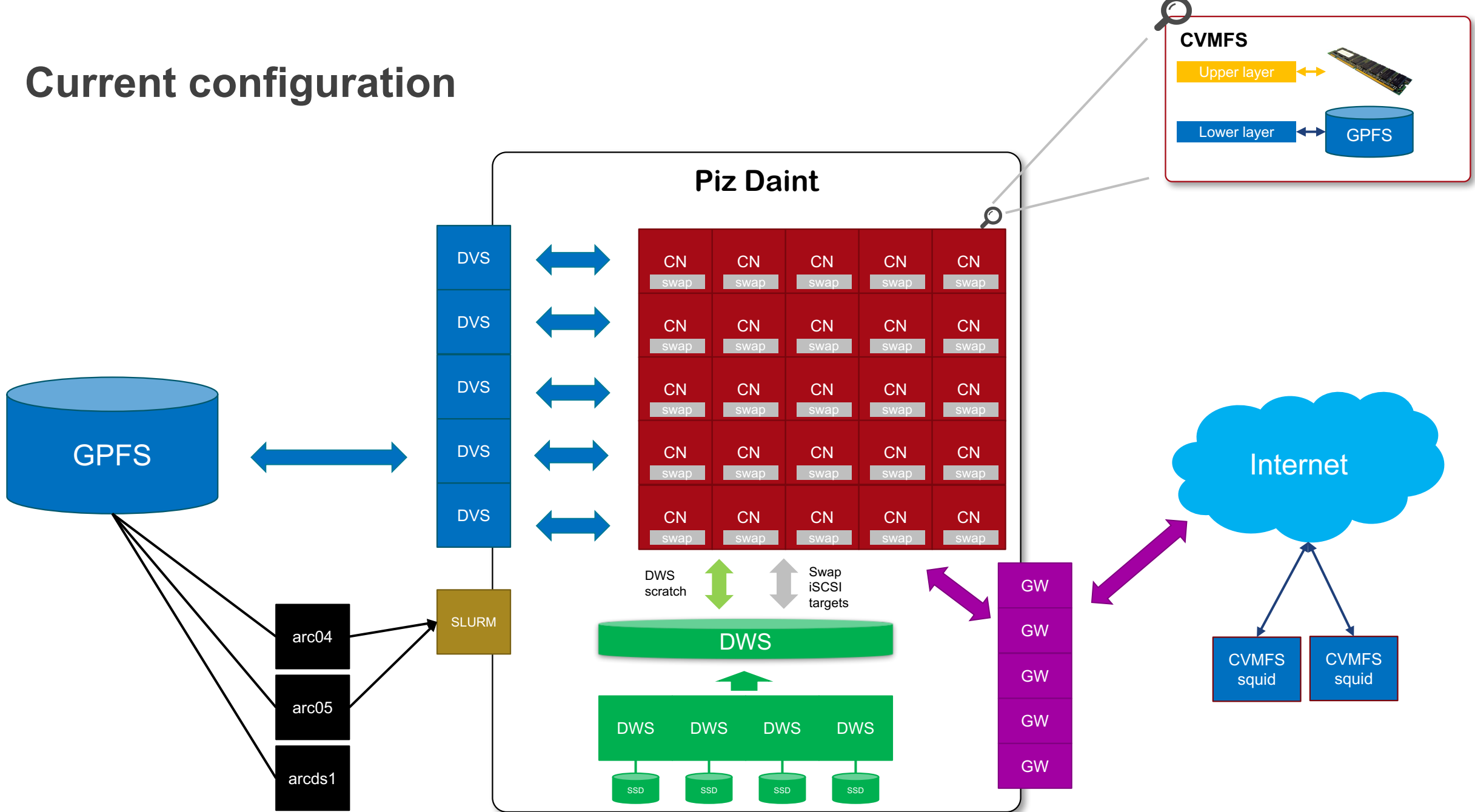
ETH zürich

Current configuration

System statistics

System utilization (since Jul 19)

- Core allocation up to ~90% (1600/1800) with 64core/node (out of 72).

- Queues filled with pending jobs:

```
VO Running Pending
--------------------
atlas   197 115
cms      30   3
lhcb   1170  53
ops       0   0
dteam     0   0
```
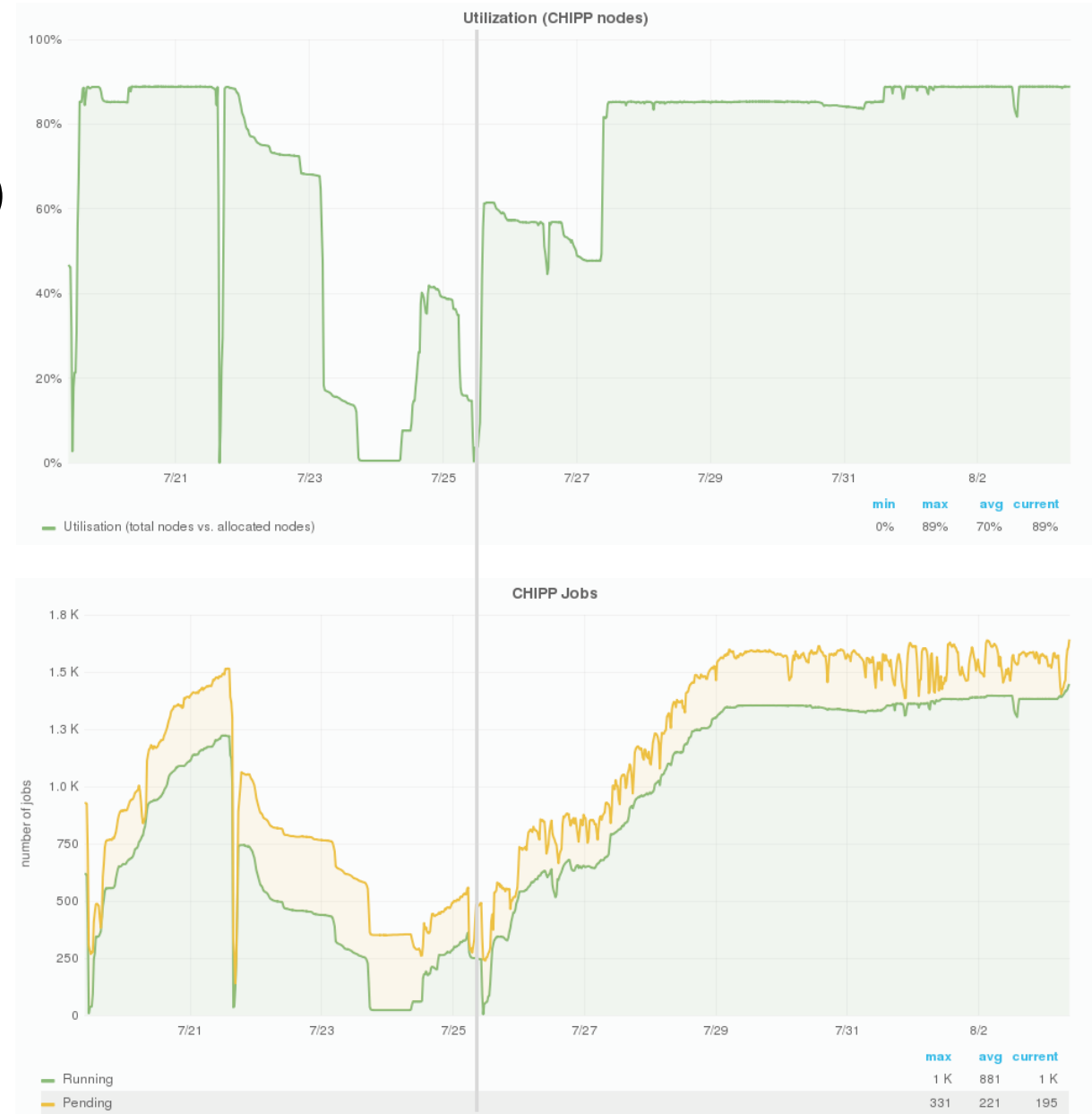
System utilization (since Jul 19)

- Load
  - Number of procs in line with load
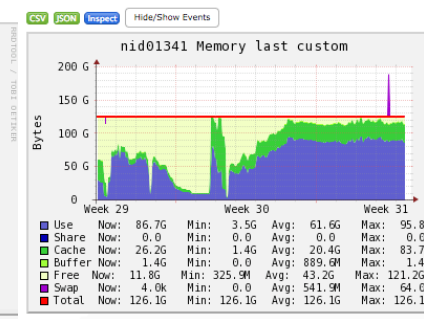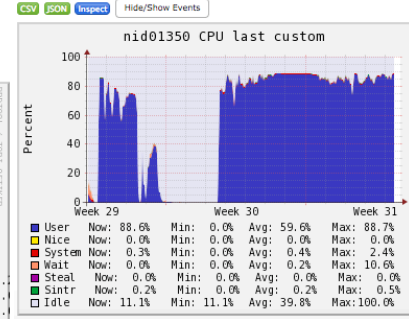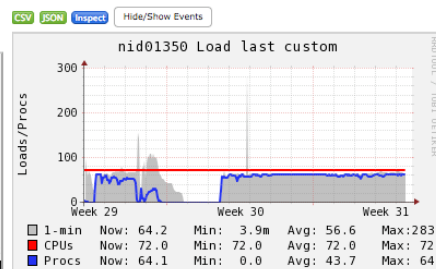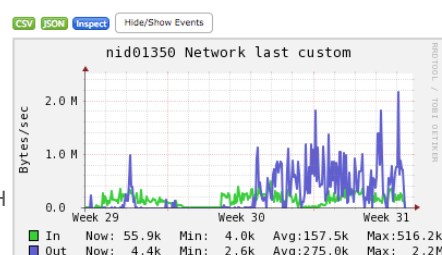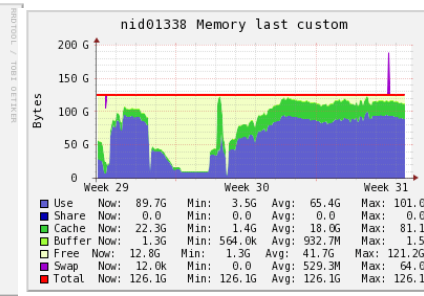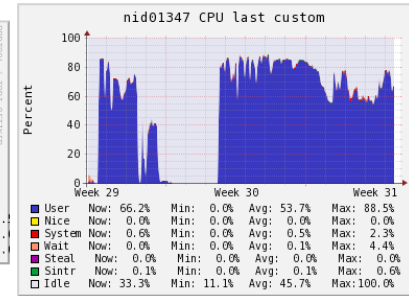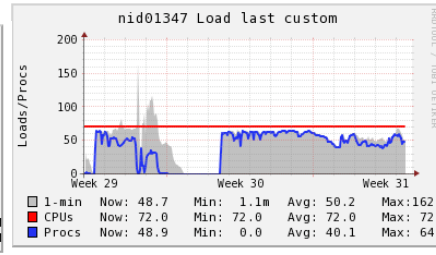 - Some load peaks due to IO
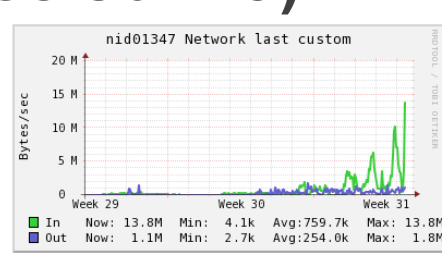- CPU utilization
  - Almost flat on ~85%
  - IO wait negligible
- Memory utilization
  - About 16GB in cache
  - About 8-16GB free on average
- Network
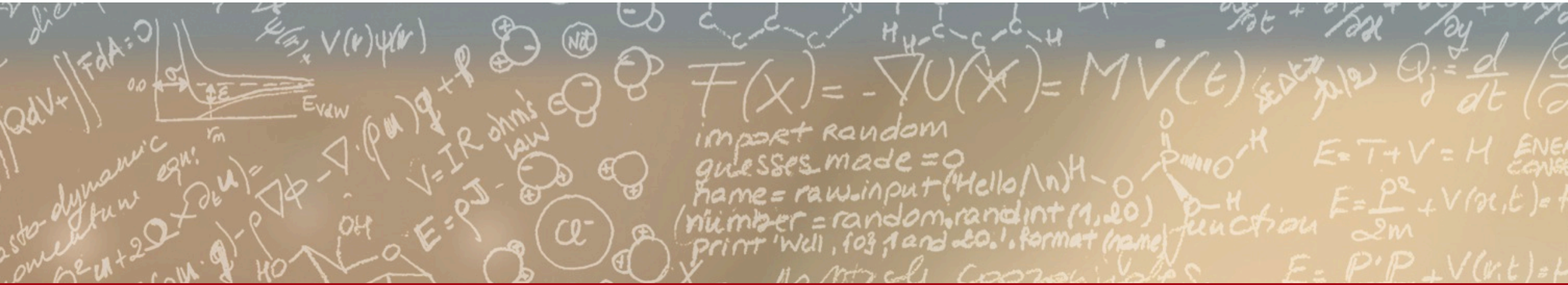 - No significant activity
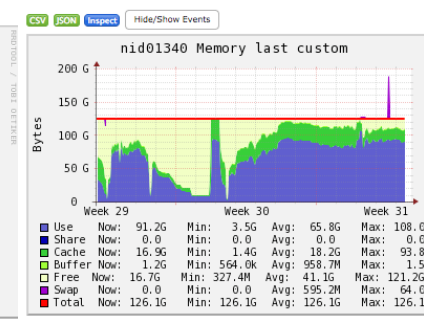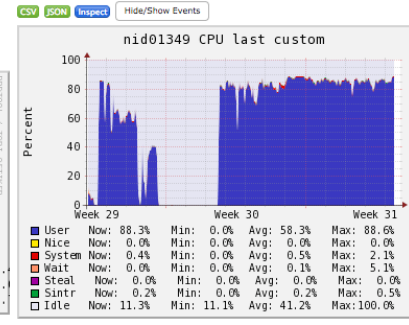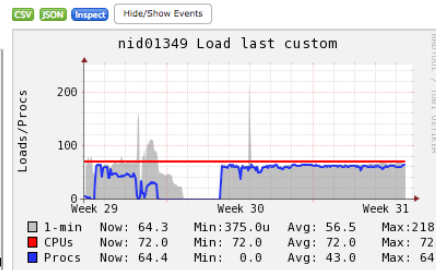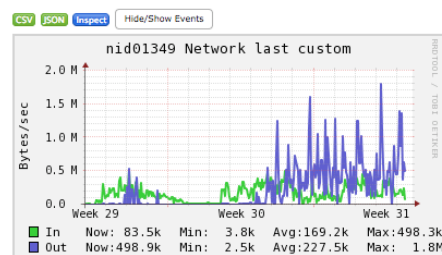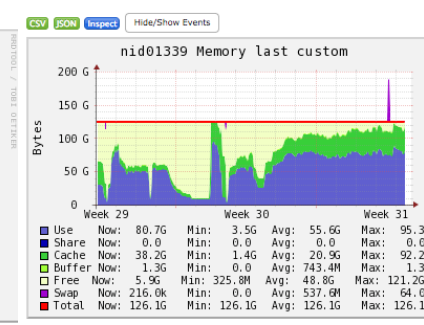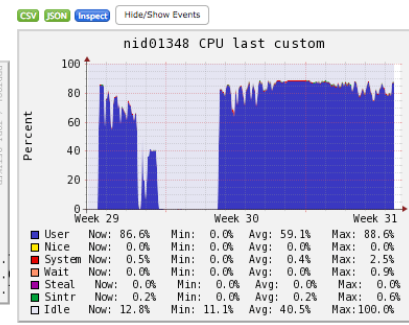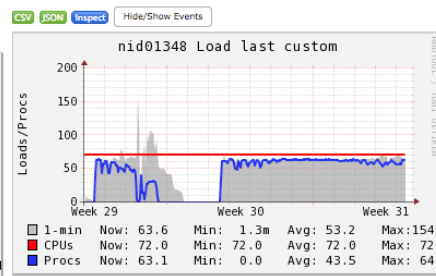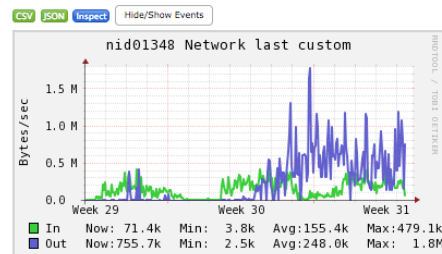
Next steps

Next steps

- Memory utilization suggests that there might be room to squeeze in a few cores and allocate ~66 or 68 cores instead of 64 cores
  - This could be positive (more CPU used!)
  - Or negative if nodes start swapping
  - Any comments?

- DataWarp evaluation ongoing
  - The intention is to test small (1-2 nodes) and evaluate from there
  - Any comments or ideas?

- Extended maintenance around Sep. 27 2017
  - Length to be confirmed
  - Upgrade to CLE 6.0.UP04 being tested

Thank you for your attention.