Migrating a WLCG tier-2 to a Cray XC50 at CSCS-LCG2

Gianfranco Sciacca - University of Bern (speaker)
Miguel Gila - Swiss National Supercomputing Centre

HEPiX Fall 2017 - KEK Tsukuba, 18th October 2017
Piz Daint and Phoenix at CSCS (*)

- **CSCS (Swiss National Supercomputing Centre)** hosts a supercomputer that ranks #3 in the TOP500 as of July 2017
  - *Piz Daint* is a Cray XC40/XC50 providing 19.6 petaflops (*Linpack*)

- CSCS also hosts a WLCG tier-2 site delivering computing and storage services to the **ATLAS**, **CMS** and **LHCb** experiments
  - *Phoenix* is a x86_64 cluster that has been in continuous operation and evolution since 2007
  - Currently provides 6.2k CPU cores (~70k HS06) and 4.8 PB of storage (*dCache*)

(*) see site report by Dario Petrusic (Tuesday session)
The LHConCRAy project at CSCS

Consolidation project to run LHC jobs on Piz Daint
- Partners: CSCS, CHIPP (Swiss Institute of Particle Physics - ATLAS, CMS, LHCb)
- Started ~2 year ago with preliminary studies on a Cray TDS
- Operated in parallel with Phoenix
- The goal is to run ALL VO workloads without changes to the experiments’ workflows

Normal workflow:
- Plugs transparently in to the experiments’ WMSs

Roadmap
- Measure performance in the production environment
- Produce a cost study (until Dec. 2017)
- Decision due: migrate to the Cray or revert to invest on Phoenix
Operational challenges

- **OS environment**
  - Cray Linux Environment (stripped down SUSE)

- **Diskless nodes**
  - scratch areas, job workdirs, ARC cache/sessiondirs
  - /tmp
  - swap

- **Data delivery / access / retrieval**
  - network connectivity

- **Memory management**
  - operate with ≤ 2GB/core

- **Job scheduling**
  - job prioritisation and fair-share in the global environment

- **Software provisioning**
  - CVMFS cache performance in absence of local disk

- **Scalability**
  - depends on all of the above
Current configuration

![Diagram of Piz Daint system configuration]

- **PHOENIX**
  - GPFS
  - CVMFS
  - squid
  - VO Boxes
  - dCache
  - ARGUS BDII

**Internet**

**Lower layer**
- GPFS
- CVMFS
- squid

**Upper layer**
- Internet

**CVMFS preloaded (RO)**
- arc-cache (RO)
- scratch

**Swap**
- iSCSI targets

**DWS**
- scratch
- ISCSI targets

**SLURM**
- arc04
- arc05
- arcds1

**Piz Daint**
- CN
- swap
- DVS

**DWS**
- SSD
- GW
Current configuration - data access, memory, scheduling, OS

- 25 compute nodes: 72 HT cores (Broadwell), 128GB RAM, diskless, 64-68 cores used (12.96 HS06)
  - nodes are dedicated and have IP connectivity with public IP addresses

- 1 production ARC CE + 1 ARC data stager + 1 test ARC CE (internal) - in ARC native mode
  - Perform full data staging I/O (for ATLAS)
  - Can scale up the number of stagers as needed
  - **ARC caching not enabled**: each job has its own copy of all files *(at least for now)*

- **SLURM LRMS**
  - Dedicated WLCG partition *(jobs are not node-exclusive - 1-core or 8-core)*
  - **Memory is not consumable**: Enforce 6GB/core limit for to catch rogue jobs
  - **When scheduling is disrupted due to rogue users, all suffer**

- **OS environment**: *Cray Linux Environment - CLE6.0.UP04* *(based on SUSE 12)*
  - Jobs run in **Docker containers using Shifter**
  - Image is a WLCG full WorkerNode *(CentOS6, EMI3, HEP_OSlibs_SL6, CVMFS)*: 2.6 GB
  - **https://hub.docker.com/r/cscs/wlcg_wn:20170731**
Current configuration - shared file systems

- **Most critical pieces of the puzzle, ongoing work**
  - Dedicated **GPFS file system** shared with the Phoenix T2 cluster
    - used by ARC for input and output data staging, scratch dirs, job work dirs
  - **5 DVS** (**Cray Data Virtualisation Service**) nodes exposing GPFS to the CNs via 40GbE links
    - A few DVS related issues/bugs to deal with
    - Had to turn off ARC caching => issues with symlinks over DVS
    - Issues when a file is accessed by multiple clients, performance degrades very quickly => job timeouts
  - **4 DWS** (**Cray Data Warp Service**), SSD-based ([http://www.cray.com/datawarp](http://www.cray.com/datawarp))
    - Cannot mount on nodes external to the Cray, e.g. the ARC CEs for ARC job sessiondirs
    - **Swap** on DataWarp **enabled**: one iSCSI device per node with 64GB each (**not really used yet**)
    - **Job workdir** (**$RUNTIME_LOCAL_SCRATCH_DIR**) and /tmp: **ongoing work**
      - the key is to distribute metadata operations to more servers
      - this requires creating dynamic allocations per job with a fixed size
  - **Docker images**
    - On the **Cray Sonexion 1600 Lustre FS**
    - so far it has worked very well with no IO penalties because of being on Lustre
Current configuration - CVMFS

- CVMFS running natively on CNs using **workspaces** and **tiered cache**, two new features of CVMFS
- Was previously configured to use a XFS loopback filesystem on top of DVS as local cache
- Tried also Cache on DWS, but this suffered from data corruption

- **CVMFS_WORKSPACE=$PATH** allows us to store data directly on a DVS projected filesystem (no more XFS)
- 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

- **CVMFS_CACHE_hpc_TYPE=tiered** with upper layer in-ram storage: *this can dramatically increase performance*. We have a CVMFS upper layer of 6GB in-RAM per node (shared by all VOs).

- **Lower layer RO on GPFS**: **cvmfs_preload** now a fast and reliable service provided by CERN for HPC sites. This syncs several times a day. If a file is not found on the local caches, the query propagates to the outside.
System utilisation

Utilization (LHConCRAY)

- Utilisation (total CPUs vs. allocated CPUs)
- Monthly average
- 64-core per node relative 100%

<table>
<thead>
<tr>
<th></th>
<th>avg</th>
<th>current</th>
</tr>
</thead>
<tbody>
<tr>
<td>Utilisation</td>
<td>56%</td>
<td>67%</td>
</tr>
<tr>
<td>Monthly average</td>
<td>51%</td>
<td>67%</td>
</tr>
<tr>
<td>64-core per node</td>
<td>89%</td>
<td>89%</td>
</tr>
</tbody>
</table>

Gianfranco Sciacca - University of Bern
System utilisation

CPU statistics (LHConCRAY)

<table>
<thead>
<tr>
<th>Type</th>
<th>min</th>
<th>max</th>
<th>avg</th>
<th>current</th>
</tr>
</thead>
<tbody>
<tr>
<td>Running</td>
<td>0</td>
<td>1,693 K</td>
<td>1,009 K</td>
<td>1,206 K</td>
</tr>
<tr>
<td>Idle</td>
<td>0</td>
<td>1,800 K</td>
<td>650</td>
<td>162</td>
</tr>
<tr>
<td>Unavailable</td>
<td>0</td>
<td>1,800 K</td>
<td>138</td>
<td>432</td>
</tr>
<tr>
<td>64-core per node relative 100%</td>
<td>1,600 K</td>
<td>1,600 K</td>
<td>1,600 K</td>
<td>1,600 K</td>
</tr>
</tbody>
</table>
Observed issues

- **Related to experiments** (*generic system bootstrap*)
  - CMS not running for several months, then low running

- **Related to middleware** (*generic system bootstrap*)
  - ARC delegations, crl's updates, bdii publishing

- **Related to batch** (*mostly specific to the Piz Daint operation*)
  - Fair share tuning in the global Cray environment (ongoing)
  - LHCb submitted ~10k jobs at once because of a problem with the ARC bdii, *adversely affecting the scheduling*
  - Non LHC users hammered Slurm consistently for a while, *adversely affecting the scheduling*

- **Related to Nodes** (*specific to the Piz Daint operation*)
  - Nodes silently becoming black holes (working on tuning blackhole detection)
  - Nodes being drained by the node health check (working on tuning the algorithm)

- **Related to shared FS** (*some specific to the Piz Daint operation*)
  - DVS and node load high at times due to high I/O levels
  - GPFS issues originating on the Phoenix side also affect the operation on the Cray nodes
    - e.g.: several CMS jobs writing up to 200k files each => inode starvation

- **Related to shared components** (*not specific to the Piz Daint operation*)
  - dCache, VO-boxes, network, etc
Performance and efficiencies

- We compare the performance of Piz Daint vs. Phoenix (*):
  - Per VO
  - During fixed time periods of up to one month (trying to keep the system in a frozen state during runs)
  - We evaluate monthly
  - We had 6 such runs so far since April 2017, the 7th and (very likely) last is ongoing

- Performance indicators (**):
  - Availability and reliability
  - Produced vs. available wallclock per core % \((\text{per type of job, where possible})\)
  - Good vs. Failed job wallclock % \((\text{per type of job, where possible})\)
  - CPU / wallclock efficiency % for good jobs \((\text{per type of job, where possible})\)

- The final the system performance would be the product of the following wall-time ratios:
  - % system capacity occupancy
  - % successful jobs
  - % cpu efficiency of successful jobs

(*) the comparison assumes that over long enough periods of time, the job mix in the two systems is comparable
(**) data are harvested from the experiment dashboards
Performance: per VO-efficiency comparison

ATLAS - Good VS Bad %

CMS - Good VS Bad %

LHCb - Good VS Bad %

ATLAS - CPU efficiency %

CMS - CPU efficiency %

LHCb - CPU efficiency %
- Availability and reliability are very similar for both systems
  - dominated by issues with the shared components

- Preliminary conclusion:
  - within up to 20% the performance of the two systems can be judged as equivalent
Summary and plans

- A couple of months ramp-up on Piz Daint, met and addressed plenty of grinding issues
- **Relatively stable operation**, all VOs now capable of running jobs
- Overall CPU utilisation reaching the relative maximum (but not for sustained periods)
- Memory utilisation under control: ~30GB in cache, ~1GB free on average, we have swap
- CVMFS in RAM seems to work quite well, not a single issue since we have enabled it

- **The two systems show comparable performance according to the chosen indicators**
- **Decision on future direction due by the end of the year**

**Ongoing work**
- mainly efforts to improve performance of shared scratch areas
- system tuning in some identified areas (fair-share, node availability, etc)

**What about scalability?**
- This is a concern right now
- We aim at performing a test at the 20k+ core scale in November
Thank you for your attention!

CSCS:
Nicholas Cardo
Dino Conciatore
Pablo Fernandez
Miguel Gila
Stefano Gorini
Dario Petrusic
Gianni Ricciardi

ATLAS:
Gianfranco Sciacca

CMS:
Derek Feichtinger
Thomas Klijnsma

LHCb:
Roland Bernet

Thank you for your attention!