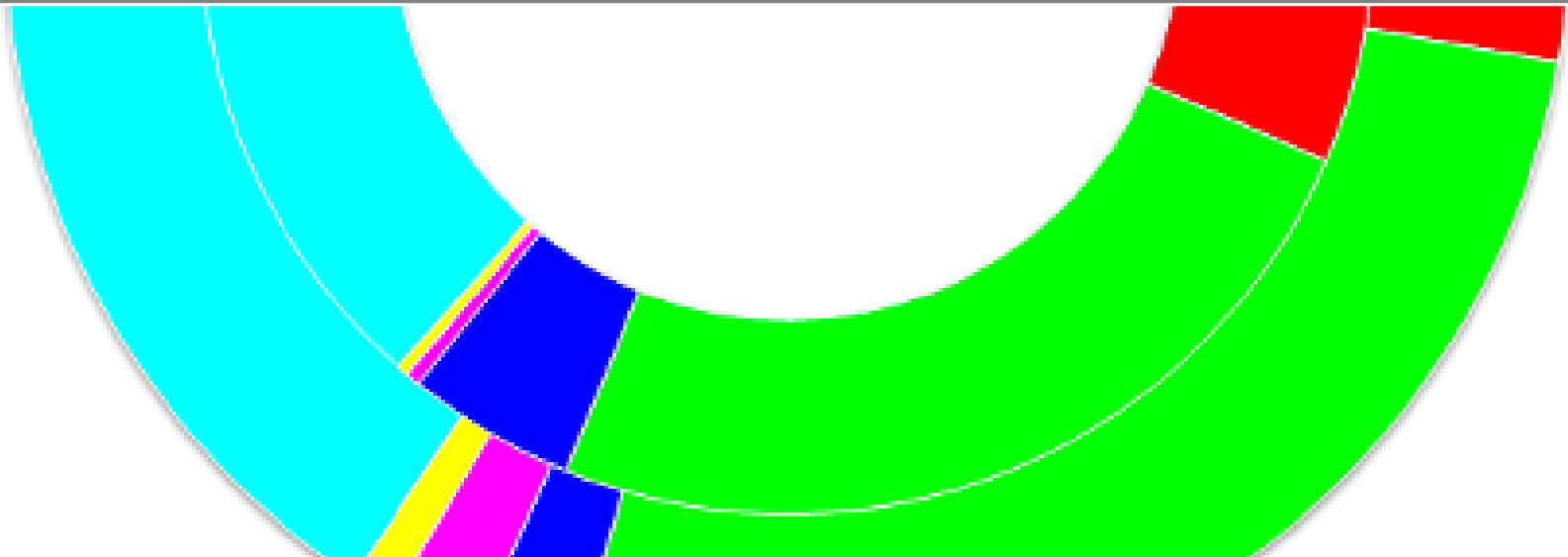


LRMS Migration at GridKa

HEPiX Fall 2012

Manfred Alef

STEINBUCH CENTRE FOR COMPUTING



Compute Fabric at GridKa

- GridKa provides compute services to 9 HEP VOs as well as to many non-HEP VOs
- Jobs submitted by all VOs are mapped to 1 huge cluster, no VO specific clusters
 - Currently 2 (related to LRMS issues) general sub-clusters + 1 test cluster
 - 1000 nodes (630 + 289 + 80)
 - 14600 job slots (6350 + 6340 + 1920)
- LRMS fair-share configurations are based on the pledged shares (i.e. HS06 scores)

LRMS History at GridKa

- 2001 ... 2003:
 - OpenPBS
 - Frequent issues, several causes:
 - "Too many" jobs submitted at the same time
 - Node fall down into strange state
 - More and more issues with every cluster expansion

LRMS History at GridKa

- 2003 ... 2012:
 - PBS-pro / PBS Professional
 - Commands and tools compatible to OpenPBS
 - Improved server/mom protocols
 - More stable than OpenPBS, but ...
 - ... increasing number of problems with every cluster expansion
 - Black hole node issues
(e.g. scheduler starts 100's of jobs on node with local home partition remounted read-only because of disk failure)
 - Jobs in strange state
(e.g. reported as „running“ but no processes on node)
 - Slow progress in solving problems by PBS support staff

LRMS History at GridKa

- 2010:
 - ...

- October 2010:
 - Split up into 2 separate sub-clusters to improve stability
- January 2011:
 - Migration to new release PBS Professional 11.0
 - Status information stored in DB instead of flat files
 - Bug in new license management tool caused downtime of 3 days
- March 2011:
 - PBS crashed immediately after reboot of server
 - Cause: corrupted database

LRMS History at GridKa

- 2010:
 - Investigation of alternative LRMS' started

Comparative Studies

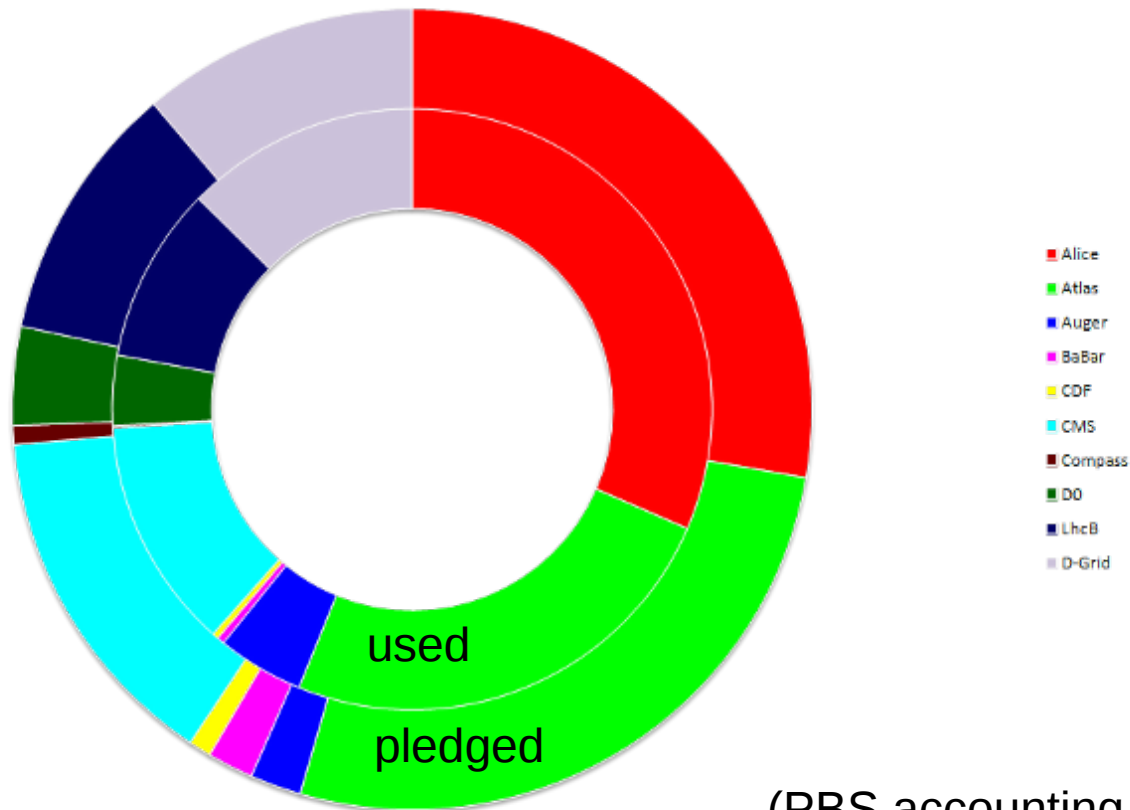
- 2010 ... 2011:
 - Tests of other batch systems
 - Torque (2.5.x) / Maui
 - Advancement of OpenPBS
 - SGE
 - Summer 2011: Univa Grid Engine
 - Test setup
 - Shadow batch system covering the whole GridKa production farm
 - Alternative LRMS' running in parallel using different network ports than PBS Professional
 - "Sleep" jobs, random run time
 - Test results
 - Torque / Maui: several issues
 - Grid Engine: tests succeeded, also with many very short jobs, and also with 100 job slots per node

Migration Plan

- Migration to Univa Grid Engine
 - 1st July 2012:
 - Test cluster: 80 nodes, 1920 job slots
 - 15th Dezember 2012:
 - Migration of the whole compute farm
 - Single cluster (to be used by all supported VOs)

Fair-Share Scheduling

- Long-term accounting statistics demonstrate that every VO gets its pledged share (in units of HS06-hours):



(PBS accounting, June 2012)

Fair-Share Scheduling

- However, many complaints of long maximum waiting time of jobs
- Main reason of long waiting times of (a small portion of) jobs:
 - Fluctuating job submission of VOs:
 - Being inactive for a long period of time, then submitting bulks of jobs within a few hours
 - Taking over many job slots because of good fair-share scores
- Adjustment knob of LRMS': halftime of past job usage

Fair-Share Scheduling

- Grid Engine provides more configuration options than other LRMS':
 - **Fair-share (share-tree) ticket policy:**

The fair-share or share-tree policy allows that the usage shares from a Univa Grid Engine cluster can be defined in a hierarchical tree structure. The main characteristic of this policy is that past job usage is taken into account.
 - **Functional ticket policy:**

The functional policy is derived in each scheduler run from scratch and does not incorporate any historical data.

 - Based on 4 individual types of entities: user / project / department / job
 - **Override ticket policy:**

The override ticket policy is helpful for temporary changes in the overall scheduling behavior. With this policy an administrator can grant extra tickets to individual jobs or classes of jobs. It allows a temporary override of a configured and applied policy like the share tree or functional ticket policy.

Fair-Share Scheduling

- Different policies can be active at the same time
- Fair-share configuration of the Grid Engine test cluster at GridKa:
Share-tree and funktional ticket (based on projects=VOs) policies
equally weighted

Experiences with the Grid Engine Cluster

- Learning curve!
- Very fast and efficient ticket handling by Univa support team
 - Located in Regensburg, Germany
(no time shift – same time zone as Karlsruhe)
- Stable operation, no crashes, no black hole node issues, ... so far
- Very flexible fair-share policies available

Experiences with the Grid Engine Cluster

- Annoying:
 - Different syntax of commands and configuration files for share-tree and functional share policies
 - Share-tree policy:
 - Tree
 - 1 file describing the whole tree
 - Functional share policy:
 - 4 individual, flat groups of entities
 - 1 config file per entity or qsub flag (job)
 - No tree-like dependencies
- Missing feature:
 - Monitoring of job efficiencies (cpu-per-walltime ratio of running jobs)

Questions, Comments?