Naf User Committee
Jan. 2011.

SGE and AFS playing together

Andreas Haupt

January 12, 2011

once again: explanation of the afs issues

- AFS provides a strong guarantee for clients to get the up to date (not cached) information of files and directories
- clients activate the callback functionality via file / directory accesses at files servers holding the data - after that the information is cached on those clients and file servers won't be asked again.
- if files / directories change, the file server sends out callbacks to all registered clients, so that they know, they aren't holding the correct information any more
- problem arises when hundreds of clients registered callbacks for files / directories that change permanently and re-register those callbacks again via reading the file / directory shortly after that ⇒ callback storm
- this typically happens e.g. when SGE job's stdout/stderr is written to $HOME and the job accesses files / directories in $HOME or one of its sub directories ($HOME must be read when changing into sub directories) ⇒ if job number is high enough, file server is more or less fully booked with just sending / renewing callbacks - other accesses just hang and finally time out.

afs issue - actions taken so far…

…to reduce the probability of those problems showing up:

- limit the number of concurrently running jobs of single users
- sleep randomly before job start to stretch the period of jobs writing / reading single directories
- we now have a so called "job submission verifier" (JSV) in place
- qsub parameter: **-jsv ∼finnern/public/jsv.sh** (will finally move to another place)
- → should be tested now for framework compliance
- tells the prolog / epilog scripts to transparently create STDOUT / STDERR files locally first and copies them back later to the originally requested place.
- would not have stopped the AFS hangs last Monday ...

what next?

- The last long lasting AFS unavailability was caused by a user who took care of our suggestions on the first sight . . .
- We cannot investigate all job scripts for such mistakes
- It is likely to happen again and again - we can just minimise the consequenses.
- The only real "solution" would be to stop the AFS token provisioning for all jobs (no constantly changing directories → no callback storms).

future of the NAF batch system

- Oracle finally stated they won't support the open source version of Gridengine any more
- Future development just goes into the commercial version
- Licenses are typically sold per cpu core (site-wide licenses should be available as well)
- rough estimate: price per system increases by 50% over the batch node lifetime
- who would pay for this?
- On the other hand: our current installation works - but new developments (e.g. GPU support) are unlikely to get integrated

SLURM - a replacement candidate for SGE?

- SLURM - Simple Linux Utility for Resource Management
- Active development at LLNL for environments comparable to ours
- https://computing.llnl.gov/linux/slurm/
- Support for Kerberos5 ticket / AFS token provisioning
- Supports many MPI implementations
- Completely different commands / switches compared to SGE → Changes in all frameworks / user jobs would be needed :-(
- We need to carefully watch at the alternatives ...