T2/T3 data mangement tool status (lsm, pcache, ccc)

Charles G Waldman
USATLAS/University of Chicago
cgw@hep.uchicago.edu

ATLAS T2/T3 workshop, UTA, Nov 10-12 2009

pcache

- Uses extra scratch space on WNs as a cache for input files
- Fully deployed at UC, IU for over 1 year
- Reduces load on dCache by ~50%

(925168 hits, 945831 misses since 8/1/09)

- Tested at RAC with good results
- Under testing/evaluation at BNL
- http://www.mwt2.org/~cgw/talks/pcache.pdf

Integration of pcache with pilot jobs

- BNL: replace dccp with wrapper script

  pcache.py dccp.bin -flags INFILE OUTFILE
- MWT2: use local site mover (lsm-get)
- US sites have not all migrated to lsm, and European sites probably never will
- pcache needs to be deployed with pilot

  (field has been added to scheddb)

Integration of pcache with Panda

- Panda server has information about what jobs have run on what nodes
- If site runs pcache, then Panda can assume that files are kept on WN, and use this information for assigning jobs to nodes
- pcache needs to update Panda server when files are deleted from cache (http POST with list of GUIDs)

Issues: cache cleanup

- Cache cleanup is slow,because of inventory (basically “find” and sort by mtime)
- New cleanup code developed with help from Pedro Salgado @ BNL
- Maintain a second hierarchy of symlinks, organized by YYYY/MM/DD/HH (mru)

Issues: locking

- pcache is single-threaded, but multiple instances may be running, hence some locking is needed
- Current lock is global (inefficient)
- This led to failures during UAT ('interrupted system call')
- Working on more fine-grained locking mechanism

Issues: cache corruption

- If a checksum or size mismatch occurs, this will be detected by pilot or lsm-get
- Bad file remains in cache! (cache poisoning)
- Currently, we clean this up by hand (infrequent)
- lsm can handle this, but not all sites will use lsm

Issues: ROOT file access

- Files may be opened by ROOT I/O plugin rather than staged by pilot
- e.g. Conditions data
- This defeats caching and job/node matching
- Pcache-aware ROOT plugin?
- Can we usefully cache 'scraps'?

Integrity check: CCC

- http://www.mwt2.org/sys/ccc
- Deployed at MWT2 and AGLT2
- Has found tens of TB of 'dark data' (orphans)
- Overdue: versions for xrootd & posix
- Also need to develop version for sites with non-local LFC (e.g. T3)
- No known bugs, but memory footprint is large (2GB for ~3 million files at UC)