Software Meeting: Discussion Points

Murrough Landon – 6 July 2005

`http://www.hep.ph.qmul.ac.uk/~landon/talks`

Overview

- Issues with Multistep Runs
- Databases
- OS/TDAQ Updates
- Production Testing SW
- Regbuild
- Documentation!

Issues with Multistep Runs

Start and end of run

- Synchronisation of actions at run start
- What happens at end of run?

Headers

- What do we want, what is compatible with RCD?

DataGenRecipes

- Extension to determine what to readout

Databases (1)

Status of COOL

- COOL conditions DB software available since April (5 releases!)
- COOL and required LCG packages now “included” in TDAQ release tdaq-01-02-00 (actually need to download separately and configure)
- NB COOL/oracle needs separate download of instantclient software from CERN, after signing a site licence and getting a special account to download it (I now have one)
- Rapid development of new COOL related tools and ATHENA support
- Eg web browser for conditions DB, COOL_IO (conversions between COOL and ROOT or text files) for analysis of conditions data and bulk insertion from other SW, also XSLT transforms of COOL to/from XML
- TDAQ databases may also use COOL for storage (and history) instead of OKS files – with the same DAL

Databases (2)

Status of L1Calo developments

- Not so rapid development unfortunately (at least not by me)
- Started work on “coolL1Calo” package
- Tested COOL/oracle access via simple class from lxplus and QMUL (latter needs modification of oracle instantclient download from CERN – add devdb10 to tnsnames.ora)
- Not tried local COOL to SQL database
- Various thoughts, mostly still on paper, about how to organise calibration (and other configuration) data, possible designs of classes, interaction with our existing DB etc.
- Some work for CMM calibration done by RichardB?
- Becoming a critical item...

Databases (3)

COOL basics

- Tables live in folders: need to design a folder structure. Top level reserved for detectors, eg /tdaq/l1calo/etc
- Should have some levels of subfolders, eg /calibration, /configuration(?), /runtype(?), productiontest(?), installationtest(?) etc. And maybe further subfolders for physics, cosmics or test calibrations?
- Requested not to have too many tables: one table for a given object for all instances in the whole system. Our existing calibration data has objects for channels (eg PprAsic, JemInput) and for whole modules (CpmReadout, TtcrxSettings). Expect one table for each.
- Rows identified by channel ID. At present just an integer, though string IDs have been requested by users. At least for the moment this will need a scheme to encode/decode channel IDs into crate, module, (submodule?) and channel within (sub)module for each kind of table.

Databases (4)

COOL basics (continued)

- Bulk insertion (in one DB access) of multiple channels with different IDs is not yet supported - though strongly requested. So for the moment need to write each value separately though this should be changed in future for better efficiency.
- Our access SW should be able to (lazily) get and cache data for just one crate at a time (for crate controllers) - ie need lists of channel IDs for requests. Though calibration programs will want to make bulk insertions for the whole system.

OS/TDAQ Updates

- Bruce proposes OS/TDAQ migration next week
- Freeze existing versions, tag all packages
- Several incompatibilities, especially in OKS databases, so not practical to maintain old and new versions – eg different nightly build failures at QMUL (RH73/online-00-22-00) and CERN (SLC3/tdaq-01-02-00)
- All sites would have to move at (roughly) the same time
- Nice to have disk from RAL with updated OS, network booting and all that, plus TDAQ, LCG etc software
- After migration, can start with GNAM, new RCD facilities, COOL and other LCG software, etc

Production Testing SW

Controlling sequences of tests?

- Hoped to be able to drive run control and set IS variables from a python script
- Tried several times to get TDAQ-python interface software from TGC-J group
- They claim to have lost this software and cannot find it on any computer...
- It would probably take a week or so to reinvent something similar but fairly minimal...
- ...if we really need it
- Otherwise better documented sets of procedures for non-expert testers
- NB Boost-python now comes with LCG packages needed for COOL so (hopefully) no/few extra external packages to install

Regbuild

- Recently added some read functions to “in memory” regbuild methods
- When/if to remove old API(s)??

Documentation

- There are still a lot of expert areas - you just have to know
- Use of DataGenRecipes, whats in a RunType, etc
- At least we can try to add more general documenation to Doxygen main page for each package
- What else? What priority can we give to documentation given the other pressures?