Run 2 Online SW Status

Murrough Landon

2 November 2016

- Recent Changes
- CMake
- To Do List
- Git

Recent Changes (1)

- Preprocessor
  - Added 80 MHz playback
- CPM/JEM
  - Tried to fix timing scans (removed some Root6 libraries)
- CMX
  - No changes?
- L1Topo
  - Support for delayed muons
  - Various tweaks to try and improve stability
    - But we still get a few link errors at startup…?
- Misc
  - Improvements/bugfixes to monitoring tools (esp for topo)

Recent Changes (2)

- L1Topo FW Validation
  - Trying to improve tools for validating new topo firmware
    - Currently we check timing is still good
    - Coverage of rates with patterns of hot towers
    - Check simulation in physics runs with topo disabled
  - But now topo is active for physics and expected to be stable
    - Need ways to validate new FW functionality outside stable beams
  - Can try hot tower patterns + (bitwise) simulation in HLT…?
    - Use ATLAS partition during (longer) downtime periods
  - Resuscitate playback/simulation in L1CaloStandalone
    - Extend largely unused feature to generate test vectors from raw events
      - Or can use existing (very simple) topo test vectors
    - Playback through L1Topo and compare CTP output bits with original events
    - Under development… working through various simulation bugs
    - Requires playback memories to be reinstated in next L1Topo firmware!

CMake

- CMake replaces CMT as build tool next TDAQ release
- Already added CMake support to all packages
  - Current L1Calo, Receivers and existing FEX packages
    - Easy to do in parallel without disturbing current CMT operations
  - Links:
    - https://twiki.cern.ch/twiki/bin/view/Atlas/DaqHltCMake
    - https://twiki.cern.ch/twiki/bin/view/Atlas/LevelOneCaloOnlineCMake
    - Talk at recent weekly meeting https://indico.cern.ch/event
- Next steps
  - Nightly builds, release builds etc
    - Probably after moving to gitlab
  - Use CMake dashboards for results, logs etc
    - Might try to reproduce our existing nightly build web page
    - Clearer than more generic CMake ones in some respects

Future Changes (TS/EYETS)

- Preprocessor
  - Any plans to use complex LUT functions? Signal generator?
  - New COOL folder structure (again!) to ease switching between low and high mu configurations
    - Not simple: existing structure is assumed in various places
- CPM/JEM/CMX
  - Need to get timing scans automated again
    - Add CMX scans to calibration panel (before end of Run 2??)
- L1Topo
  - Continue improvement of tools for checking new firmware
    - Hot towers, event playback
- And lets not forget
  - Receiver control until Run 4
  - Simulation/rodmon?

Gitlab (1)

- We need to move from SVN to Gitlab
  - Before CERN stops support for SVN from end 2017(?)
  - Better sooner than later (especially with move to CMake)
- When?
  - Developers need to learn git: different philosophy from SVN
    - I only just started trying to understand it myself…
  - Build/release/patching scripts and strategies need changing
  - For Run2 operations it seems best to wait until the shutdown
    - Although it can be prepared its basically a big bang change on one day
- Optimistic scenario
  - I will try and test migration this month
  - Try and switch in December??? Plan B: early 2017
    - But this will depend on other distractions, eg requests to think about other L1Calo upgrade issues, FEX software, mappings, etc, etc

Gitlab (2)

- Thoughts about Gitlab model
  - In the long term it seems best to slavishly follow closely what TDAQ have decided
    - Gitlab group for online software
      - Separate from existing atlas-l1calo gitlab group
      - May be easily to add custom "continuous integration" support?
    - Separate git repository for each current SVN package
  - Comments welcome
    - NB offline looks likely to go a different way…
- TDAQ and other links
  - TDAQ model: https://twiki.cern.ch/twiki/bin/viewauth/Atlas/DaqHltGit
  - Intro: https://twiki.cern.ch/twiki/pub/Atlas/DaqHltGit/git-intro.pdf
  - Git book: https://git-scm.com/book/en/v2

Gitlab (3)

- Handling large (obsolete) files in SVN history
  - Would like to exclude them from git
  - Possible solutions: git-lfs or git-annex?
  - Need to investigate them…
    - From initial reading my preference is gif-lfs (no special commands)
- Access control
  - Very loose in present SVN (few developers)
  - Keep it that way?
    - Though git is a bit more complex
    - Minimal burden on SW librarian?