HLT Infrastructure and Application integration

Ricardo Abreu
on behalf of the HLT Integration group
May 18, 2010
HLT Infrastructure

Software layer between dataflow and HLT steering / algorithms

Components:
- Steering controller
- Services that interface between Athena/Gaudi and TDAQ dataflow and monitoring
- Integration tools (e.g. athenaMT/PT, partition maker, runner...)

HLT Infrastructure and Application integration
HLTTTestApps

- A package of the HLT release
  - on top of TDAQ and AtlasOffline releases
- Integrates the offline and online environments
- Provides two tools: athenaMT and athenaPT

Test facility for online and offline components

- Means for testing selection algorithms before they go online
- athenaMT and athenaPT emulate L2 and EF
- Different uses: HLT development/testing/debugging, TDAQ development/testing/debugging, stream reprocessing
**HLT Test Apps**

- Support for many different use-cases. Examples are:
  - default mode - read events from file and process them in L2/EF
  - with online histogramming - creates a localhost partition
  - monitoring mode - connecting to a running partition
  - interactive mode with python prompt

**Examples of recent improvements:**

- Support for *user commands* to L2PU/EFPT (client/server) → test
  HLT’s reaction to configuration updates

- Fault tolerance for partially invalid events → reprocessing

- Better cleanup strategy → reprocessing/testing HLT/testing TDAQ

- Timeout thread → test and tune HLT algorithms

- Execution of python cmds/scripts in interactive mode → TDAQ
  debugging/test HLT’s reaction to conditions update
**THistSvc and Histmon**

- **Histmon** → provides histogram publishing to dataflow software
- **THistSvc** → interface that provides HLT software access to histogram publishing functionality (online/offline)
  - selection software is unaware of online/offline distinction
- Thousands of applications incorporate histmon to periodically publish histograms to IS (configuration defined period)
Histmon – Main Improvements

- Clearly separated interfaces from implementations at file level
- Lumi-block “aware” histograms
- Parallel publishing - one thread per histogram priority group
  - solved bug causing publication periods to be missed
  - improved performance
PartitionMaker

TDAQ package for creating and manipulating ATLAS partitions from a small set of parameters

Main improvements

- Support for the new Coral Proxy – replaces the old DBProxy
  - simplified proxy allocation
  - more support for localhost HLT partitions

- Possibility to set default post-processing plugins – custom parsing of standard PM generated partitions
  - Sysadmins can now enforce constraints that should be applied to any partition in environments like P1 (e.g. logs directory)
  - Defined by the environment variable TDAQ_PM_POSTPROCESSOR – they are applied before any other ones the user supplies
Runner

A tool that can automatically run HLT-enabled partitions from a small set of options and a standard partition configuration XML file

- Nightly tests for HLT/TDAQ releases
- Commissioning new racks with HLT code

Recent/ongoing improvements

- Bug fixes and keeping up with changes on the underlying software (e.g. starting of external windows, releasing of resources, informative output)
- Support for custom runs and results collection scripting
Switch to release 15.6.9

- Resynchronizing online HLT release with offline 15.6.9
  - HLT picks up latest fixes
  - Simplifies future bug fixes (less branching/backporting)
  - MC simulation of trigger closer to online setup
- Deployment on aggressive schedule (<2 weeks of testing)
  - Decouple SW change from menu change
  - Can live with minor bugs at start since HLT not active yet
- Algorithms tested on CAF (athenaMT/PT) and compared to 15.5.6
  - Most differences small and being followed up
  - Continuously rerunning HLT on full express stream - no crashes
- Tested online during two periods without stable beams - Integration into DAQ was very smooth
- Deployed yesterday
Short term evolution

- Consolidate conditions updates during run (e.g. beam position)
- Unify L2 and EF framework code further
- Improve the interfaces for reprocessing data with trigger information in the CAF
- Changes for Gaudi v21 (which offline is planning to use)

Manpower

- Contribution up to now – 4.4 FTE
  - Loosing 1.2 FTE in the near/immediate future
  - Possibly loosing another 1 FTE in October
  - Key components left without maintenance
  - Key knowledge leaving
Conclusion

Substantial improvements of tools, libraries, and services of the HLT infrastructure that have an important impact on diverse other activities
Thank You!