Development, Validation and Integration of the ATLAS Trigger System Software in Run 2

The ATLAS Trigger System

Goal: Record 1 collision of interest out of every 40,000 delivered by the LHC every millisecond

Challenges:

- Enormous data volume
  - 1 collision every 25 ns with ~50 interactions
  - 100,000's of readout channels
  - ~1.7MB per physics event
- Diversity of the ATLAS physics program
  - Allocate bandwidth dynamically to accommodate physics priorities
- The LHC operates on its own schedule
  - Provides a variety of operating conditions and collision types
    - High luminosity, low luminosity, proton-proton collisions, heavy-ion collisions
- Planned and unplanned sub-system performance changes
  - New calibrations, interventions, upgrades, failures, etc…

ATLAS must always be ready to record LHC collisions!

Collisions at 40MHz

Level 1 Trigger (Hardware/Firmware)

- Custom fast electronics and programmable hardware (FPGA)
  - Muon system, calorimeter system, topological trigger system, Central Trigger Processor
- Programmable trigger thresholds
- Determination by read-out electronics whether to send information downstream

100 kHz with < 2.5 μs latency

High Level Trigger (Software)

- Fast software based decision using the same or streamlined versions of algorithms used for offline reconstruction
- Software developed by each signature group
  - 100's of developers
- Processed online on a computing farm with ~40000 CPUs

1 kHz, with ~0.2 μs latency, written to disk

Trigger Menu

- Catalogue of L1 and HLT algorithms and parameters
  - ***Physics menu*** for data
  - ***MC menu*** for simulations
- Steers the trigger operation
  - Must be synchronized with a given release
- Coordinated from each signature (developers) with physics groups (users)
- Developers must locally validate their software and menu updates:
  - Compilation test
  - Simulation processing test
  - Data processing test, online HLT emulation

Software Release Structure

20.11.X "Base" Release

Semi-annual validation and deployment when patch release is too large

20.11.X.Y "Patch" Release For online

Weekly Validation and deployment

20.11.X.Y.Z "Development" Release For software validation

Locally validated signature software updates, built and tested daily

Release Types

"20.11" →HLT Online
"20.7"→Reconstruction
"20.3"→Simulation

Enhanced Bias Data

Most of the data processed by the HLT is rejected. Testing the HLT offline requires input data similar to that seen by the HLT online.

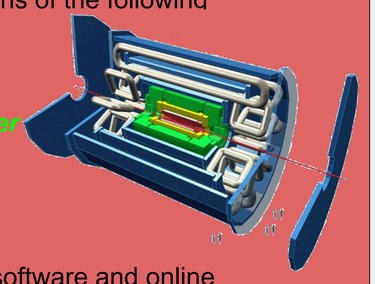
Enhanced Bias (EB) datasets:

- Are made using special triggers that accept the main L1 physics items
  - Giving more weight to interesting/rare events
- Provide input data for offline HLT reprocessing
  - Yields comparable conditions to online data taking
- Benchmark EB datasets for each flavor of LHC run are taken

Trigger Signatures

The vast spectrum of desired physics signals is organized into ***trigger signature*** working groups relying on combinations of the following subsystems:

Muon system
Hadronic Calorimeter
EM Calorimeter
Inner Detector



Each group:

- Develops trigger software and online calibrations
- Validates online performance and data quality
- Develops offline reconstruction algorithms and calibrations for data analysis
- Validates offline quality of reconstructed data

HLT Integration

The ***Online Integration Coordinator:***

- Deploys the trigger software, ensuring compatibility with:
  - Data flow
  - HLT farm nodes
  - Data reconstruction
- Oversees the HLT performance with the help of **Debug Stream & Data Quality (DQ) experts**
  - Analysis of Debug Stream data
- Ensures harmonization of future hardware/software upgrades

Trigger Software Validation Cycle

Team: Trigger Operation Coordinator | Trigger Signature Experts | Menu Expert | Release Expert | Software Validation Expert | Debug Stream & DQ Experts

This cycle is performed weekly for Physics and MC menu processing

Additional iterations when specific needs arise:

- New menu items
- New LHC run conditions
- Hardware changes
- HLT farm CPU estimates
- Debugging online errors
- Calibration changes

Build and upload both menus
- 30 minutes

Release deployed to the grid
- 5 hours

New software tags collected, release build and run nightly test
- 12 hours

Reprocessing of EB datasets
- 24-48 hours
- ~6800 hours of CPU per cycle
- Up to 4 GB memory usage per job

Production of reprocessing performance metrics:
- Average memory usage
- Individual algorithm performance analysis
- Estimated algorithm rates
- Physics performance metrics
  - Reconstructed observable distributions compared to reference
  - Trigger signature efficiencies
- ~1 hour

Investigate and resolve bugs, create new tags

Shared Oversight

Collect, create and assign bugs

START

Jet Expert: Examine and validate performance

Electron Expert: Examine and validate performance

Muon Expert: Examine and validate performance

… Expert: Examine and validate performance

Solicit feedback on performance from signature groups

Reject or Sign Off

Deployed for online running

The status of the cycle is summarized in the weekly trigger meeting

Tools

The following tools are web based

JIRA

- Commercial bug/issue tracking and development software
- Tasks are organized, assigned, monitored, discussed and archived

Twiki

- Detailed instructions for each role
- Maintenance overseen by coordinators with user input

Whiteboard (on the Twiki)

- One for every role
- Dynamic log of current issues and new information
- Useful for transitioning roles

Shiftwork

- Roles are filled using a weekly shift scheduler

Prodsys2

- Framework for organizing ATLAS software campaigns on the grid:
  - Initial data processing
  - Production of samples for physics/performance studies
  - Trigger reprocessing

AMI

- Database of:
  - ATLAS datasets
    - Data
    - Simulations
  - ATLAS software configurations
- Integrated with Prodsys2

Typical Issues

- New menu items for new or updated physics analyses
- Optimizations of existing algorithms to collect more luminosity
- Corrections and adaptations accounting for detector issues
- Hunting rare bugs that are observed during data taking

Outlook

Upcoming improvements include:

- Multi-core grid processing for faster reprocessing jobs
- Streamlining the overall ATLAS data reconstruction process

Authors: Robert Keyes, Tamara Vazquez Schröder, Simon George, on behalf of the ATLAS Collaboration

Acknowledgments: Thanks to the Natural Sciences and Engineering Research Council, Canada, for funding our efforts