Triggering on 7 TeV Collisions with the ATLAS High Level Trigger

Wojciech Fedorko
On behalf of the ATLAS
collaboration

Challenges of trigger at the LHC

- Hadron collider
 - Standard Model electroweak and new physics processes overwhelmed by QCD
 - Busy events (20 interactions/crossing at design luminosity)
- ATLAS trigger challenges:
 - 40MHz bunch crossing rate
 - Limited output bandwidth (300MB/s)
 - Large event size (1.5 MB)
- Primary trigger criteria:
 - High p_T leptons, jets, photons
 - Missing E_T

Trigger design in ATLAS

- Level 1
 - Custom electronics
 - Calorimetry, muons, MinBias
- High Level Trigger
 - Commodity PC farm shared between:
 - Level 2
 - Selection based on dedicated algorithms
 - Reconstruction in Regions-Of-Interest (ROI)
 - Minimize data transfer and CPU usage
 - Event Filter (EF)
 - Selection based on offline algorithms
 - Full event information available
 - Primarily ROI-based reconstruction

- Current System:
 - 27 racks; 30 nodes (XPUs)/rack
 - 35% of the final system
 - Each XPU Dual 4-core @2.5 GHz, 16GB RAM

HLT chain example

- Trigger chain:
 - Sequence of reco/selection algorithms
 - Multiple opportunities to reject
 - Algorithm results cached
- Trigger Menu
 - ~400 chains
 - Designed for a given luminosity scenario
 - Prescale sets
 - Define prescale and passthrough values for each chain
 - Prescale at HLT applied before chain execution

HLT Steering

- Framework for each L2 and EF application
- Executes chains independently of each other and accept/reject decision
- For each chain:
 - Check whether to start the chain based on decision of previous level
 - Apply prescale factor before execution
 - Execute algorithms of each chain
 - Caching of results; reuse results if another chain has already performed a given algorithm with same input and configuration
 - At L2 caching of requested data
- Form a decision – if at least one chain passed all steps event accepted
- Mark 'stream' for the event based on the chains passed e.g. 'Egamma', 'Muon'
- Store trigger results for offline analysis

HLT deployment and menu evolution

- Commissioning strategy:
 - HLT off for first collisions, run offline
 - HLT in flagging mode
 - If performance verified; and L1 rate high turn rejection on
 - Special case: MinBias HLT rejection

First fill in 2010!!

HLT trigger deployment cycle

- Balance between stability and flexibility
- 24h deployment cycle
- 16 releases deployed since August 1st

Feedback during datataking

- Trigger rates and comparison to predictions
- Output data composition, algorithm errors, timeouts and crashes
- Algorithm performance
 - Automatic problem detection
- Dataflow and CPU resource usage

- Needed for detailed analysis of CPU and data transfer resources
 - Allows for identification of algorithms which are heavy resource users
- Input for rate calculation at higher luminosity
- Minimal overhead
- Implementation:
 - For every event store L1 accept time stamp L1&HLT decisions
 - Sample timing data from HLT steering
 - Buffer data from multiple events
 - If event rejected by all chains write a small calibration event

Example of execution time monitoring for a typical L2 reconstruction algorithm. Algorithm time (CPU) and data access

- Run:
 - Period of data-taking with fixed trigger configuration
 - Usually several hours
- Luminosity Block (LB)
 - Short period (2 min.) within a run
 - Conditions and luminosity assumed to be constant
 - Smallest unit of time considered by data quality

Prescale set update during run

- Motivation:
 - Essential for correction of errors in pre-loaded prescale sets
 - Needed for optimization of bandwidth usage during changing luminosity profile
- Challenges:
 - Event processing in L2/EF not ordered by LB
 - Possible restart of HLT application during a run
 - A single HLT application may not process events from a given Lumi Block

Beamspot update during the run

- Beamspot: measurement of position and shape of the luminous region
 - Uses fast L2 tracking followed by a fast vertex fitter
- Important for monitoring luminous region parameters
- Used to augment luminosity measurement
- Can be used to improve performance of the HLT B physics chains

Transverse view of the ATLAS luminous region at $\sqrt{s}=7$ TeV

Implementation of prescale set update

ATLAS Event Format

- Human or automatically generated action
- ConfLB 16 bit word in the CTP data updated to the number of current LB when change prescale command received
- Similar mechanism for update of beamspot – not yet deployed

Conclusions

- ATLAS High Level Trigger provides stable and reliable data-taking
- Flexible framework and fast and safe procedures to deploy new triggers and modify existing trigger configuration
- Feedback and control mechanisms implemented at all stages of trigger deployment and operations