Status of Data Scouting
Impact on HLT Integration

Ricardo Abreu
Noemi Calace
Joerg Stelzer
Werner Wiedenmann

CERN

February 21, 2014
Motivation

- Detector output rate has to be severely reduced
  - Strict filtering and high thresholds
  - Potential undesired data bias
  - Reject potentially interesting events
- The problem could be minimized with knowledge of rejected events
- Allowed event output rate depends on event size
  - Smaller events could be accepted with a higher rate

Data Scouting (DS)

- A feature for saving some online info about events whose frequency otherwise excludes.
  - No detector data
  - Produces reduced HLTRResult
  - Relies on PEB
- Useful as feedback for subsequent trigger setups

Diagram:

Regular Event:
- HLTR
- Detector Data
- Header
- Nav
- C1
- C2
- ...

DS Event:
- HLTR
- C1
- C2
An HLTResult is recorded in the event as a ROB (same for DS events)

A DS event is marked with a DS Stream Tag

- specialization of Calibration Stream Tag → allows specifying ROB IDs for PEB
- A DS Stream Tag specifies only a ROB ID corresponding to the HLTResult

DS Stream Tags are associated with special chains

- The contents of a DS HLTResult are disjoint from those of a regular HLTResult
- An event that is tagged simultaneously with a DS ST and another ST needs multiple HLTResults

Each HLTResult is encoded into a separate ROB

- Each ROB must have a unique ROB ID
- A ROB ID is a concatenation of a Source ID and a Module ID
- The regular HLTResult ROB ID is 0x7c0000 (src id TDAQ_HLT=0x7c; mod id 0)
- To discriminate HLTResult ROBs, each separate DS tag gets a unique module ID - defined in the menu
- We can then have several result ROBs (e.g. ids 0x7c0000, 0x7c0001, 0x7c0002, ...)
Impact on HLT Integration

Main work done by Noemi
https://indico.cern.ch/event/287058/contribution/5/material/slides/0.pdf

Impact on HLT Integration

▷ `hltinterface::HLTResult` updated to include several ROBs — memory for result ROBs still allocated by the HLTPU

▷ both sides have to adapt:
  - On the PSC's side, the HltEventLoopMgr asks for multiple ROB payload serializations
    - It packs them into eformat ROBFragments
    - It ensures the overall space limit is respected → signals eventual truncation and ROB omission
    - Fills EDM size histograms
  - athenaHLT outputs the multiple HLTResult ROBs (as HLTPU)
    - It also now needs to filter and discard multiple input result ROBs

Progress

▷ Only potential adjustments left in the PSC and Noemi’s side
▷ next: work on athenaHLT
Specific issues — settled (in principle)

*Force-accept* and *Force-reject* are flags that can be set on the PSC to accept or reject all events, after all regular HLT processing.

### Effect of force-accept/force-reject

- **force-accept**: before
  - we used to only force-accept events that were not already _accepted_
  - event was considered _accepted_ if it had _any_ stream tag
  - if we didn’t change, pure DS events would not be force-accepted
    - We could screen stream tags — unjustified

- **force-accept**: revised meaning → a force-accept debug stream tag is _always_ added (i.e. independently of whether there were already stream tags)
  - no problem duplicating events into debug stream that already has most of the events in it
  - potential DS ROBs that weren’t triggered aren’t saved, but they can always be derived from the full event

- **force-reject**: unchanged → the HLTResult _always_ returned to the caller as received
Under consideration

- Considering **individual limit** on DS ROB size
  - To avoid DS results becoming bigger than the standard result

- **Who adds the ROB ID** to a DS stream tag?
  - **Possibilities:** PSC, ResultBuilder, Steering, Algorithm
    - All of these have the knowledge to do that (source and module IDs)
  - What happened for calibration tags?
    - In the case of detector ROB IDs → Algorithm (?)
    - In the case of special calibration result ROBs → Algorithm or Steering (?)

- **ideal suggestion:** concentrate this responsibility in the same layer in all cases
- **fall-back suggestion:** at least don’t leave this for anything above Steering

- **EDM size histograms**
  - Used to fill 2 histograms, for events with and without truncation
  - There are now different ”types of truncation”:
    1. exclude part of regular result ROB (as before)
    2. exclude part of DS result ROB
    3. exclude full DS result ROBs
    4. a mix of 2 and 3

- **Do we keep only the same 2 histograms?**
  - If so, when do we write to each of them?
DS implementation in the HLT Integration software depends on new hltinterface
- cannot collect for now — need release on top of new tdaq-common(-01-26-00)
- it will be difficult to test against HLTMPPU before that
- But Noemi’s part is already in

Will test with athenaHLT — use simple fake menu with DS
- integrate with HLTMPPU when possible