Conditions $\mathsf{DB}\leftrightarrow\mathsf{DAQ}$ configuration DB

Mikhail Remnev

2019.08.29

Motivation

One of main motivations for ECL to keep electronics configuration consistent with Conditions DB is ShaperDSP logic check procedure in DQM.



This check will fail on correct data if:

- DSP coefficients in FPGA != DSP coefficients in DQM.
- Thresholds in FPGA != Thresholds in DQM.

Electronics configuration is also relevant for simulation and reconstruction.

Updating HLT payloads after changes to the hardware



- This means that DQM will use invalid payloads for some time period. Logic test will fail, HLT software trigger will use outdated calibrations.
- Note: it is not possible to get actual version of DSP data in DQM.
- From my experience, online integration convener usually updates the database very quickly, so it is not a major problem.
- Still, it might be useful to consider backup options, in case both conveners are unavailable.
 - Authorize run coordinator to approve staging requests?

ECL configuration in DAQ DB

- To reduce complexity in configuration management, each configuration type has single configuration object for all COPPERs (instead of 52 objects, as it has been initially).
- The following naming schema has been introduced for setting sector-specific configuration:

all-thr1	:	30 # threshold_1 = 30 for all channels.
barr-thr1	:	31 # threshold_1 = 31 for barrel channels.
fwd-thr1	:	$32 \text{ # threshold}_1 = 32 \text{ for forward endcap channels}.$
bwd-thr1	:	33 # threshold_1 = 33 for backward endcap channels.

• Runtype-independent information is stored in separate DBObjects.



• DBObjects in *cursive* need to be synchronized back to ConditionsDB.

Management of ECL energy thresholds



- There is one manually defined conversion from ConditionsDB to DAQ DB and one conversion from DAQ DB to ConditionsDB.
- Same procedures are prepared for attenuator coefficients.
- Similar procedures are planned for DBObjects with run-specific configuration.

$\mathsf{DBObjects} \leftrightarrow \mathsf{Conditions}\mathsf{DB} \text{ payloads}$

 \underline{lf} other people also do something similar, it might be worthwhile to have a standardized procedure for data transfer between two databases.

 $\mathsf{DBObjects}\mapsto\mathsf{ConditionsDB}\ \mathsf{payloads}$

- Payload with key \rightarrow value container.
- Then conversion will work even after major changes in DBObject structure. (ECL configuration DBObject have changed a lot since the start of Phase 3, in response to firmware updates and WF saving problems)

 $\mathsf{Conditions}\mathsf{DB} \text{ payloads} \mapsto \mathsf{DBObjects}$

- TBufferJSON::ConvertToJSON(obj) can probably be used...
- Are there any other possible solutions?
- Both of these conversion procedures might be useful, but not truly necessary for ECL.
- Will this conversion scheme be useful for anyone else?

backup

Changing DAQ DB configuration based on run type

• To reduce human error, some parts of ECL configuration are adjusted automatically based on run type.

```
// On RC_LOAD
if (run_type_from_RUNCONTROL == "debug") {
   save_waveform_data = 0;
} else {
   save_waveform_data = 1;
}
```

- This feature is useful but such parameters are currently hardcoded in daq_slc/copper/ecl/src/ECLFEE.c as runtype-dependent.
- Is something like this used by other subdetectors?
- One alternative is to increase complexity of configuration structure by adding extended interpretable statements: waveform_saving : "runtype == 'debug' ? 0 : 1"