

Level 3 Trigger

Seokhee Park

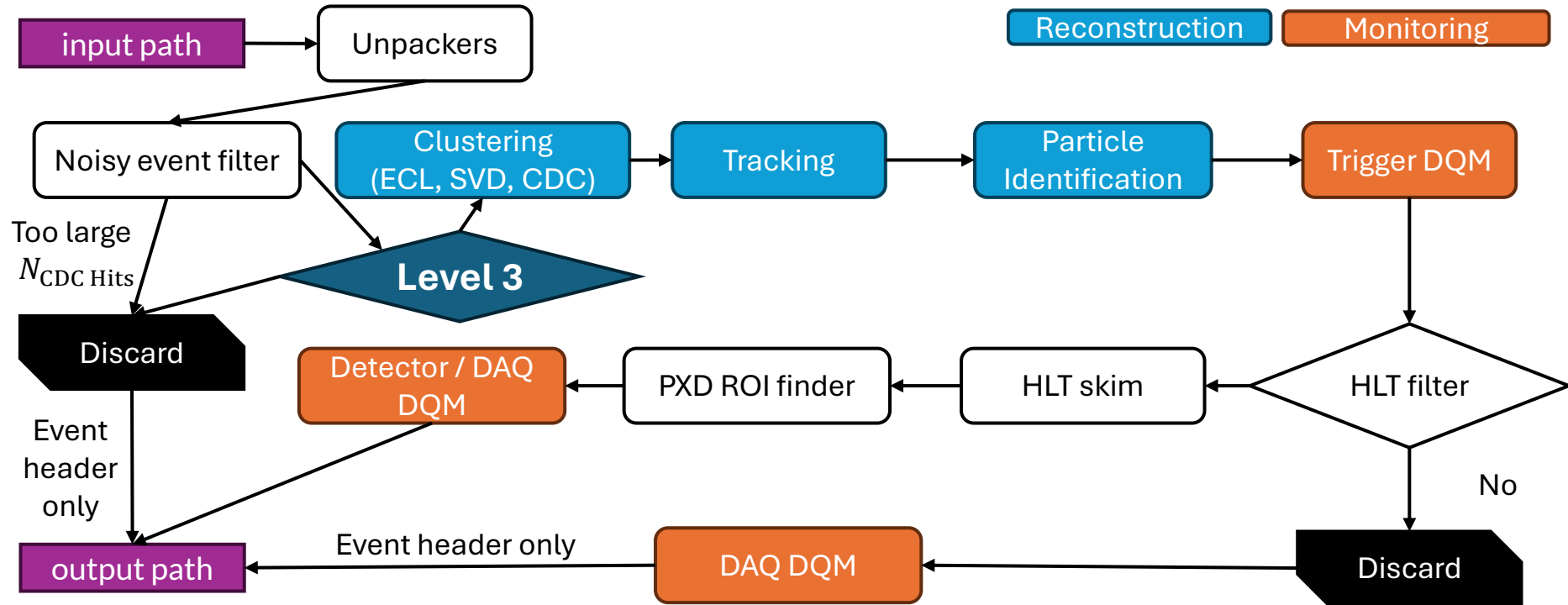
KEK, IPNS

on behalf of the Belle II DAQ group



Introduction

- L3 intended as “**more configurable pre-filter including fast (but not precise) reconstruction of CDC tracks and ECL clusters**”



Introduction

- Currently developed version is inherited by Kakuno-san.
 - The module is known as “Fzisan”. Probably continued from Belle.
 - Archive: <https://gitlab.desy.de/belle2/daq/fzisan-archive>
- (My L3) project is basically for **CPU-only HLT**; **different to HLT acceleration**
 - Mid-term project compared to the heterogeneous HLT.
- The original code is not working on the latest BASF2
 - Some updates are needed.
- The coding style is not close to the current tracking or clustering module.
 - **Rewriting the code** for easily asking to the tracking and clustering expert later
 - This is more important than what I think before. Give the highest priority.
- The first short term goal: find **the most conservative selection** on any of event rejection rate

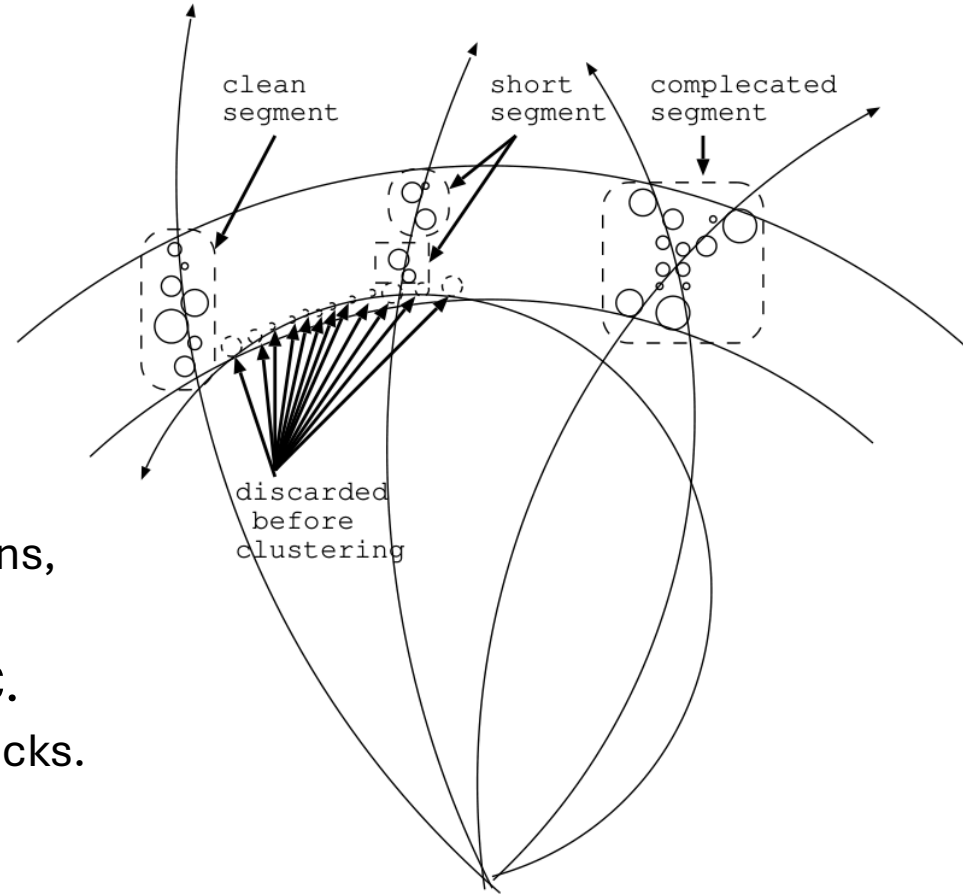
Introduction

- The level 3 trigger introduces **fast clustering** and **fast track finder**.
 - After that, a selection can be done by the L3 tracks and L3 clusters.
- In the original code, there is no complex and multiple trigger conditions like the current HLT filter&skim calculator.
 - For now, a single or a few hand-writing conditions can be done by using the number of clusters/tracks, total energy, momentum, etc.

```
addParam("dr_cut", TrackTrigger->drCut(),
        "Minumum |dr| value of a reconstructed CDC track", 1.0);
addParam("dz_cut", TrackTrigger->dzCut(),
        "Minumum |dz| value of a reconstructed CDC track", 4.0);
addParam("pt_cut", TrackTrigger->ptCut(),
        "Minumum transverse momentum of a reconstructed CDC track", 0.3);
addParam("es_cut", m_EsCut,
        "Minumum energy of the seed for a reconstructed ECL cluster", 0.01);
addParam("ec_cut", m_EcCut,
        "Minumum energy of a reconstructed ECL cluster", 0.02);
addParam("min_n_trks", TrackTrigger->minNGoodTrks(),
        "Minumum number of good CDC tracks", 1);
addParam("min_energy", EnergyTrigger->minEnergy(),
        "Minumum threshold of the energy sum of ECL clusters", 4.0);
addParam("find_evt_vtx", m_findEvtVtx,
        "Flag for event vertex finding", 0);
addParam("save_data", m_saveData,
        "=0:HLTTag only, =1:HLTTag+L3Tag, =2:HLTTag+L3Tag+L3Track+L3Cluster", 2);
```

L3 Track

- The L3 tracking consists with **segment finding in a super layer**, **segments linking**, and finally tracking fitting.
- Especially, the **segment finding is simplified** for speed.
 - Discard low p_t segments
 - Discard complicated segment, hits connected more than two connections, without separation
- The tracking is done only with CDC.
 - HLT filter uses both CDC and SVD tracks.



L3 Cluster

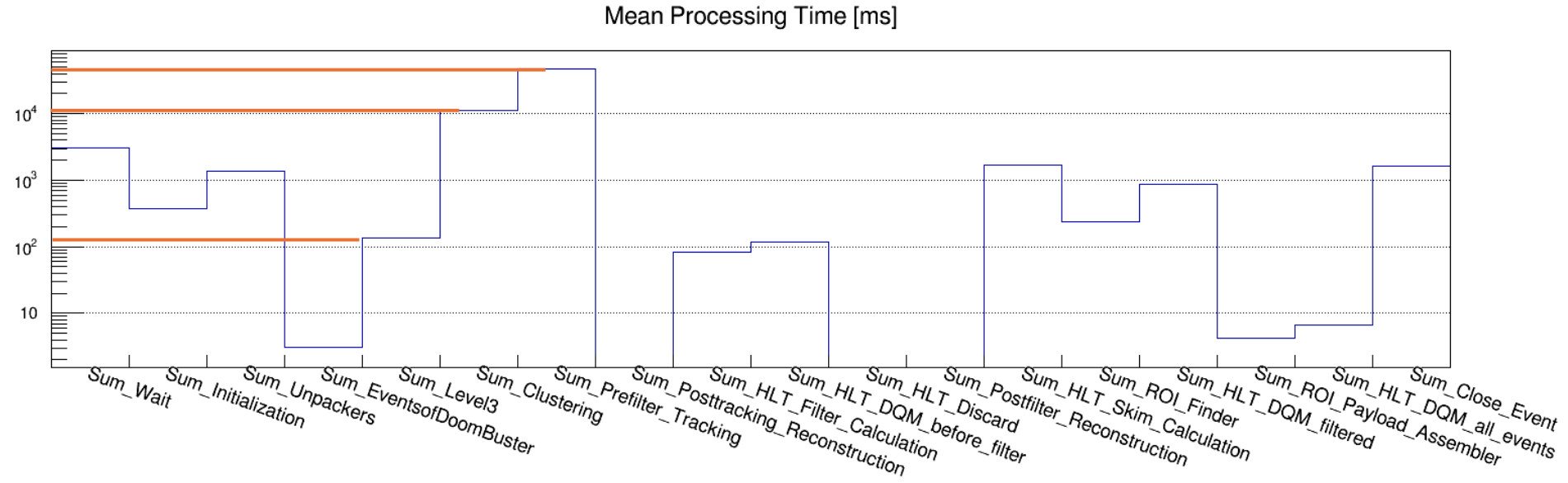
- For the fast clustering in ECL, we use slightly different conditions.
 - The cluster is basically **a just connected hits**.
 - **No limitation** on **the number of local maximum hits** in a cluster
 - **No limitation** on the number of hits in a cluster
 - Neighbor for connected hit is **only 4-direction**: up, down, left, and right
 - The **cluster position** is defined as a **most energetic hit** in the cluster
- We may give more conditions in this stage (just an idea yet)
 - Size of cluster, minimum energy cut of a hit or a local maximum hit, ...

Level 3 monitoring

- A dedicated DQM module for level 3 is ready, but...
 - For now, only can extract the final objects like the number of tracks/clusters, final momentums, dr, dz, etc.
 - No internal items like drift distance, timing, etc.
- While keeping the dedicated DQM module, make the Level 3 module to record DQM histograms directly.
- The performance (CPU mean processing time, memory usage, etc.) monitoring is included in the test setup.
 - There is no issue on memory consumption.
 - The mean processing time is much faster than my expectation.
 - If we have performance issues, I need to separate the module into the tracking, clustering, and trigger calculating for more detailed checking.

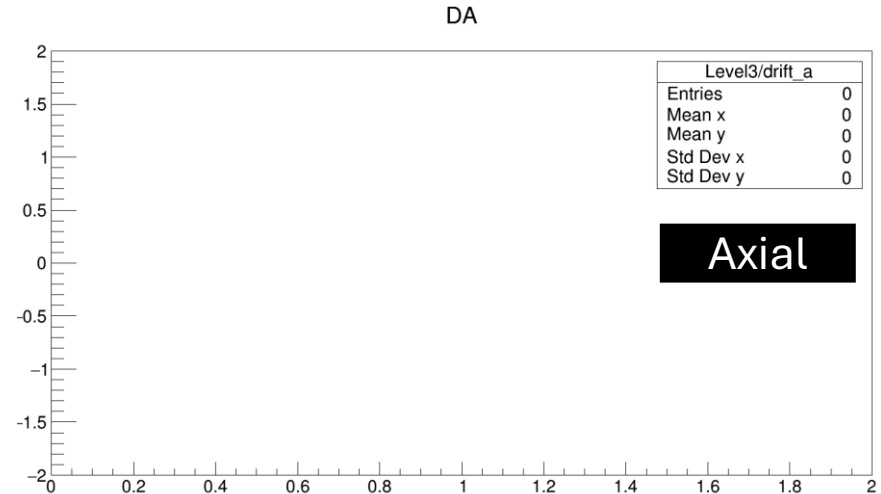
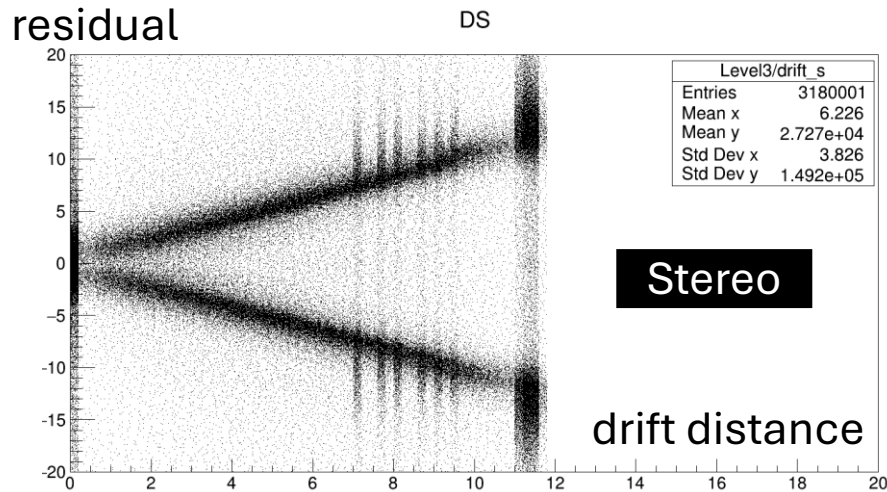
Level 3 computing performance

- The level 3 module's mean processing time is very small.
 - The mean processing time of the Level 3 module is $\leq 0.5\%$ of the sum of clustering and pre-filter tracking.



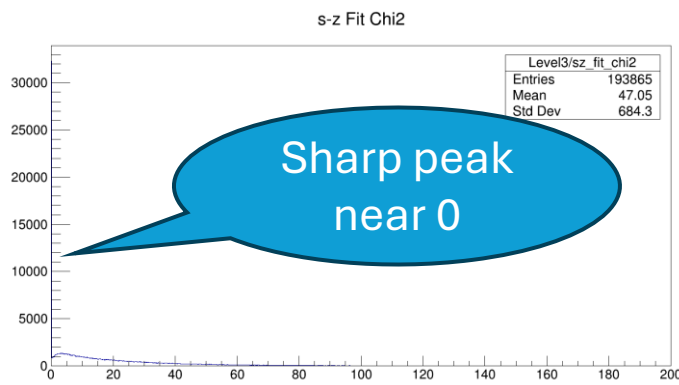
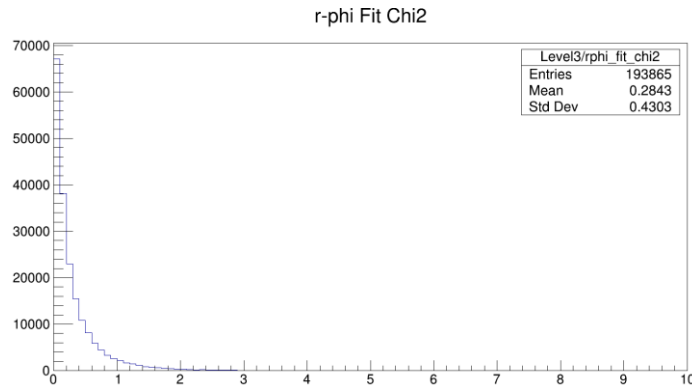
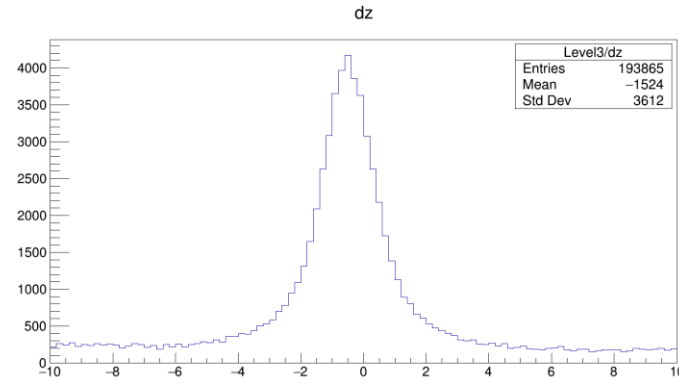
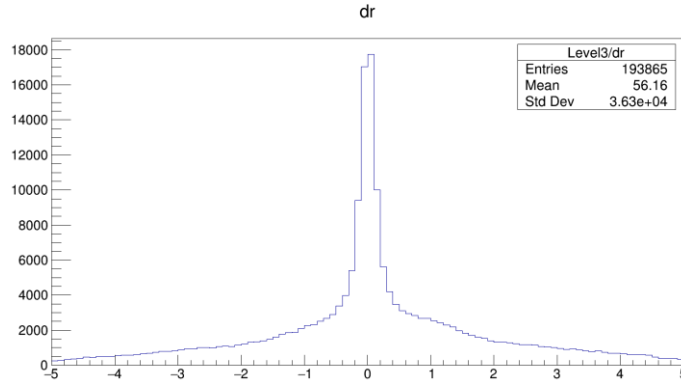
Level 3 reconstruction DQM

- However, not 100% sure that the tracks and clusters are correctly reconstructed.
- Using the pre-defined DQM set by Kakuno-san, some of DQM histograms are just empty even if they should exist.
 - For example, the comparison of **drift distance** (coming from drift time) and **residual** (distance between CDC hit and reconstructed track), the **stereo layer result is shown**, but the **axial result is empty**.



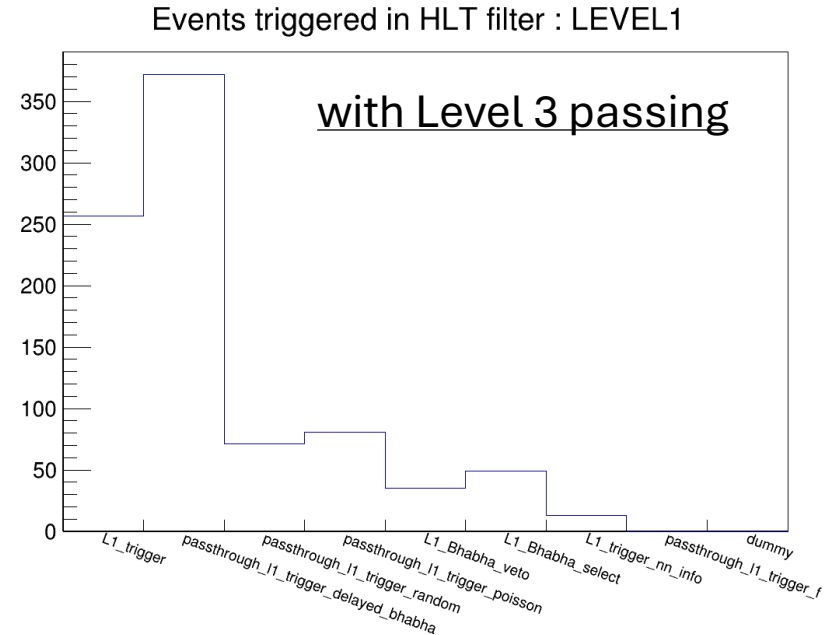
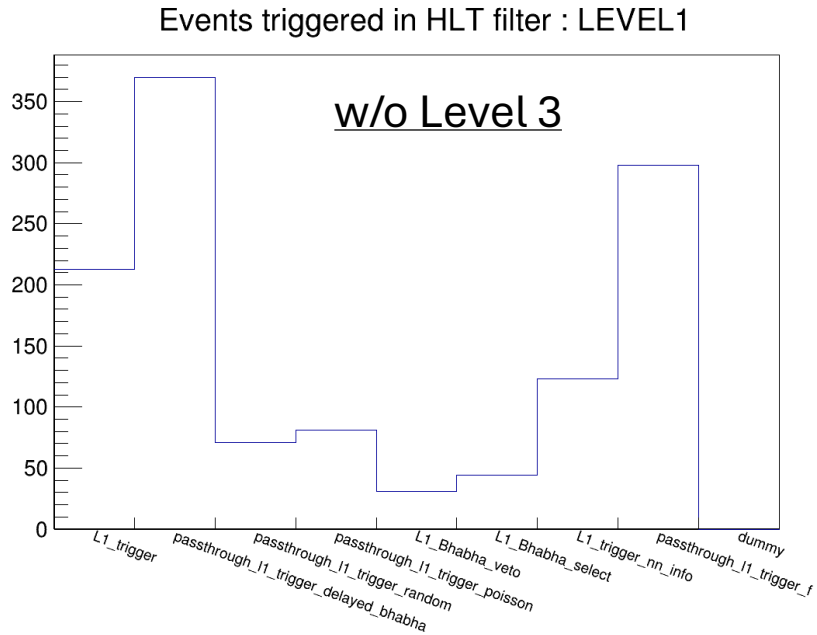
Level 3 reconstruction DQM

- Final results like dr, dz, or track fitting χ^2 are shown correctly.
 - Source code cleaning up is needed for correct understand + asking experts



Level 1 implementation

- The **level 1** information is installed **in the Level 3**.
 - The original Fzisan code don't use the level 1 information for selection.
 - Differences in L1_trigger_nn_info and passthrough_l1_trigger_f



0 track, 0 cluster trial

- With the level 1 information, try to give a condition: discard no L3 tracks and L3 clusters if it doesn't have level 1 info.
 - I expect that certain amount of “filtered” data are kept, but....

```
bool Level3Module::doCalculation(Belle2::SoftwareTrigger::SoftwareTriggerObject &calculationResult)
{
    if (calculationResult["l1_trigger_delayed_bhabha"] == 1) return true;
    if (calculationResult["l1_trigger_poisson"] == 1) return true;
    if (calculationResult["bha3d"] == 1) return true;
    if (calculationResult["bhapur"] == 1) return true;
    if (calculationResult["bhapur_lm11"] == 1) return true;
    if (calculationResult["l1_bit_f"] == 1) return true;
    if (calculationResult["l1_trg_NN_info"] == 1) return true;

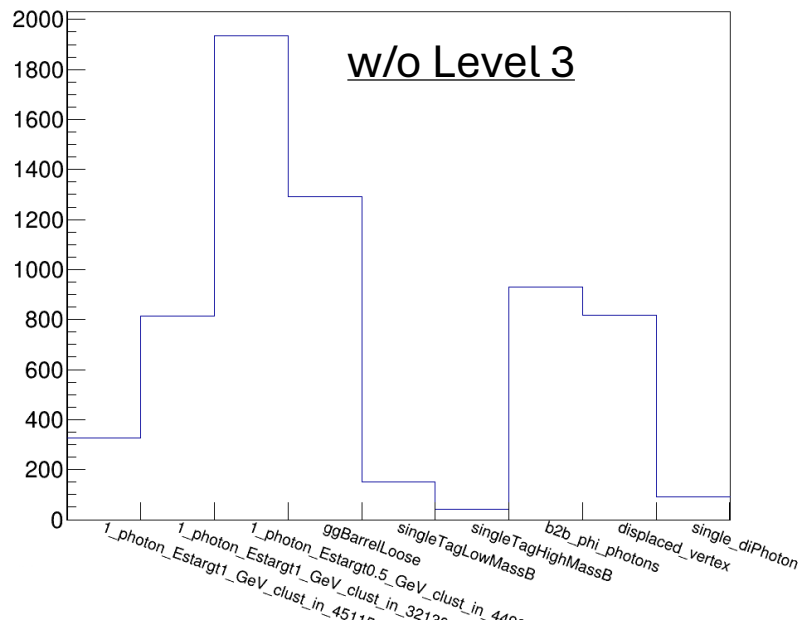
    if (m_l3Tracks.getEntries() == 0 && m_l3Clusters.getEntries() == 0) {
        return false;
    }
    return true;
}
```

Taken from
FilterCalculation

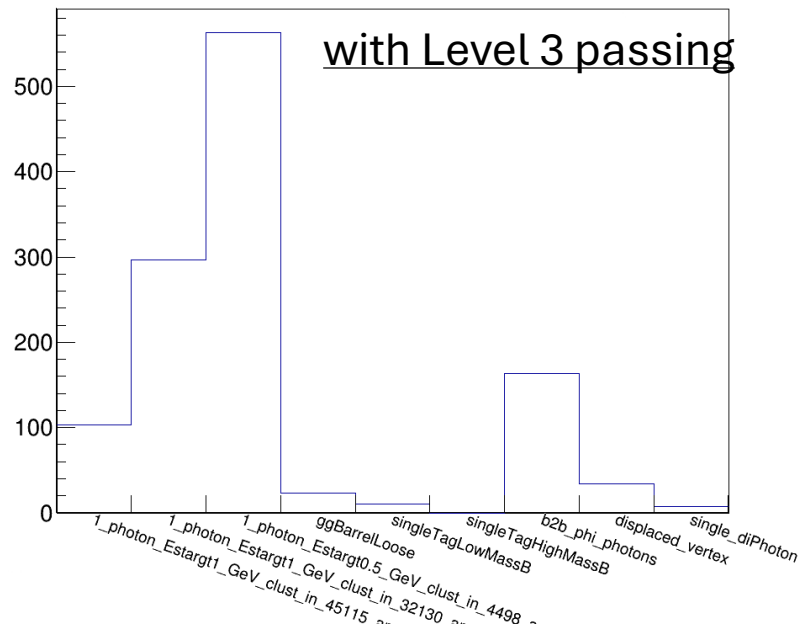
0 track, 0 cluster trial

- By checking the filter menu, **10x difference** is found for all the trigger menus except to L1-related.
- Need to carefully check the L3 track and cluster reconstruction.

Events triggered in HLT filter : PHYSICS



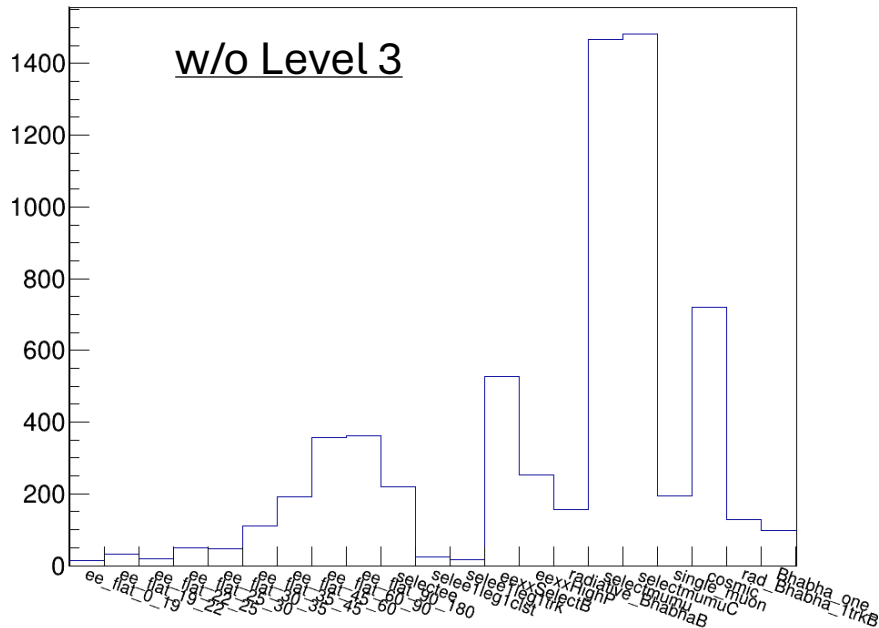
Events triggered in HLT filter : PHYSICS



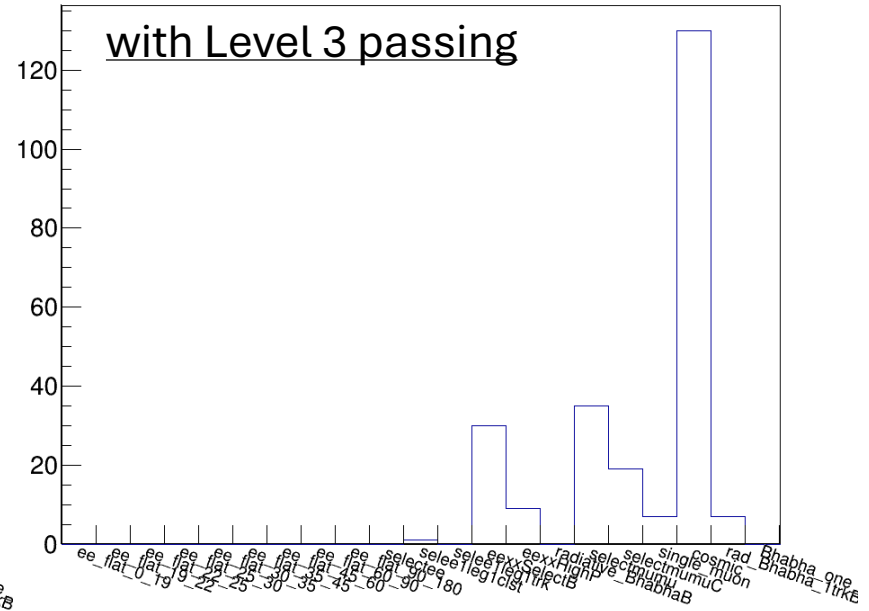
0 track, 0 cluster trial

- By checking the filter menu, **10x difference** is found for all the trigger menus except to L1-related.
- Need to carefully check the L3 track and cluster reconstruction.

Events triggered in HLT filter : QED



Events triggered in HLT filter : QED

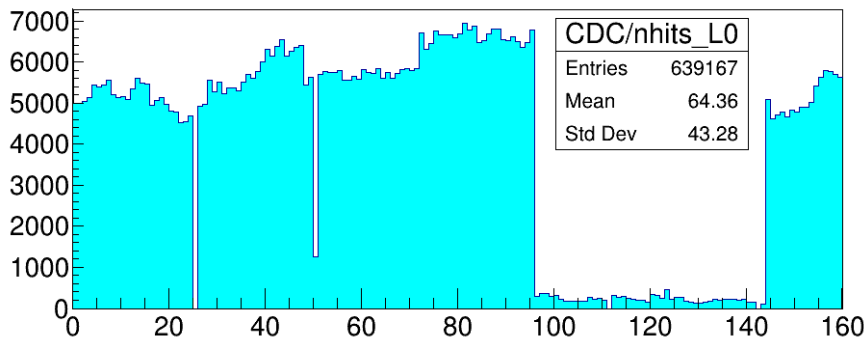


Checking CDC DQM

- While **nhits** of each layer is **similar**, **occupancy** shows strange **difference**.

w/o Level 3

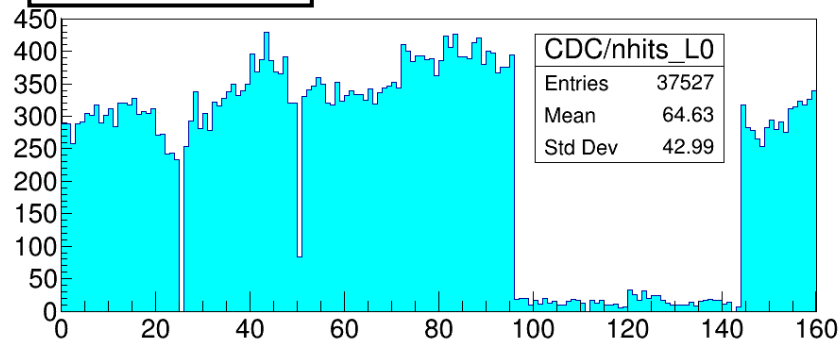
nhits Layer 0



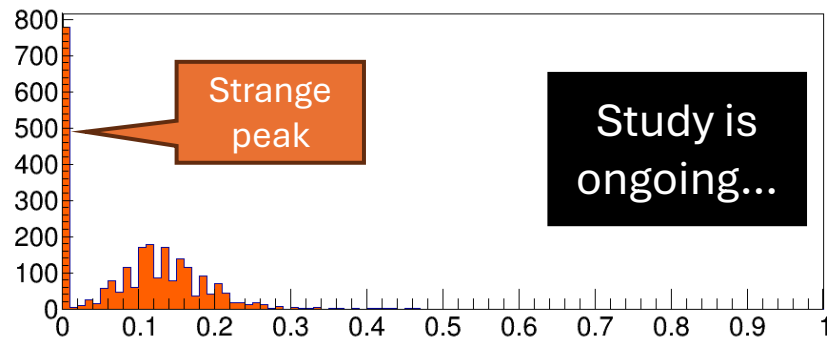
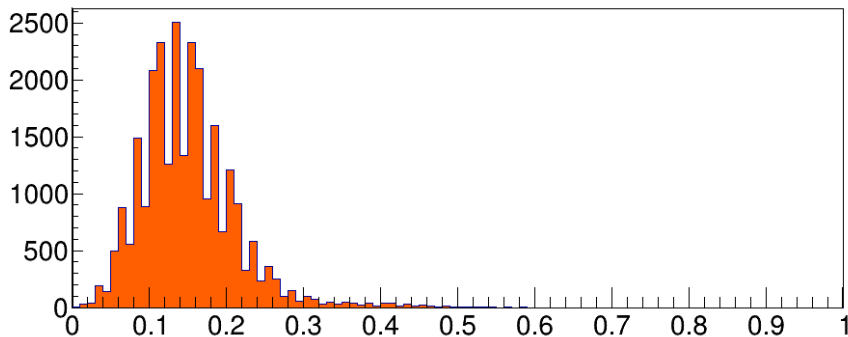
occ. each layer

with Level 3

nhits Layer 0

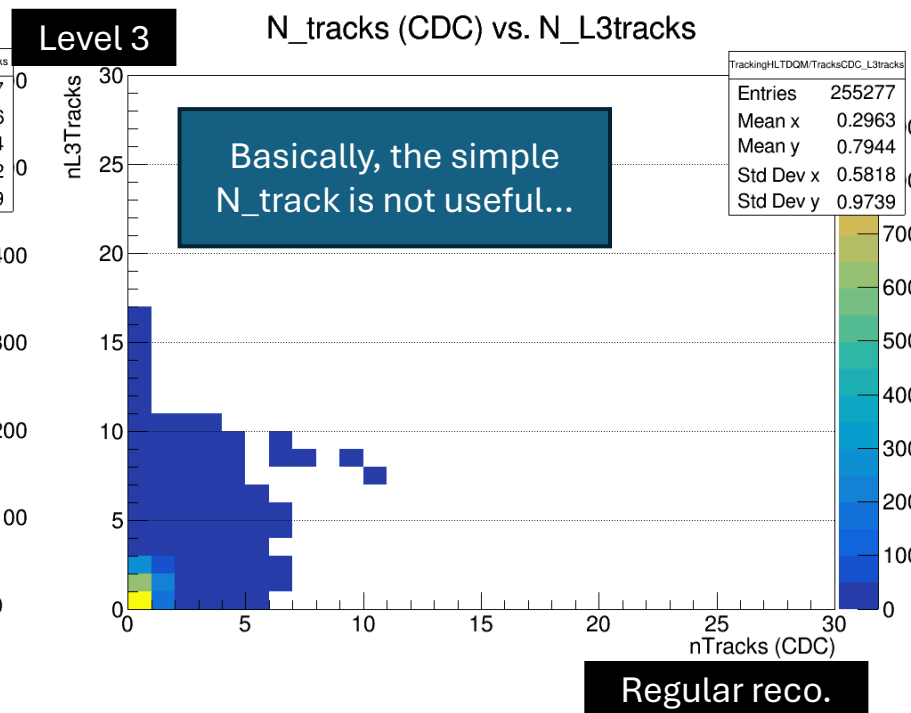
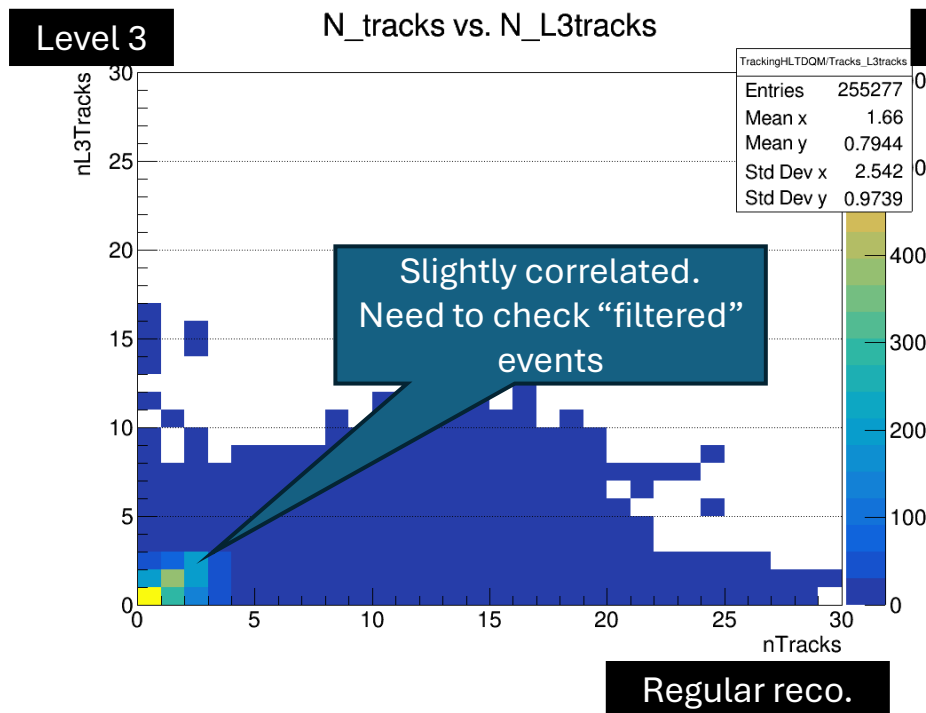


occ. each layer



Comparisons: L3 track vs. regular track

- Compare all tracks / CDC-only tracks with L3 track (beam_reco_monitor)
 - No useful information...



Summary

- The Fzisan code is used for developing level 3 trigger.
 - The original development was suspended in 2015.
- Primitive conversion to the latest BASF2 build is done.
- Performance test is good shape, only 0.5% of CPU usage compared to the pre-tracking and pre-clustering.
- However, by checking pre-built histograms, not all the information is shown. More study is necessary.
- Level 1 menu is successfully installed in the Level 3 trigger.
- Tried to 0-track, 0-cluster condition, but too many events are discarded.
- I'm feeling that current development is very slow. First, focus on the re-writing source code for asking experts.