

Development of HLTPrefilter

TRG–DAQ Workshop

Sourabh Chutia^{*}, Rahul Tiwary[†],
Vidya Sagar Vobbilisetti[‡], Karim Trabelsi[†]



^{*}IISER, Mohali

[†] TYL, KEK

[‡] INFN, Pisa

[†] rahul.tiwary@tyl.in2p3.fr

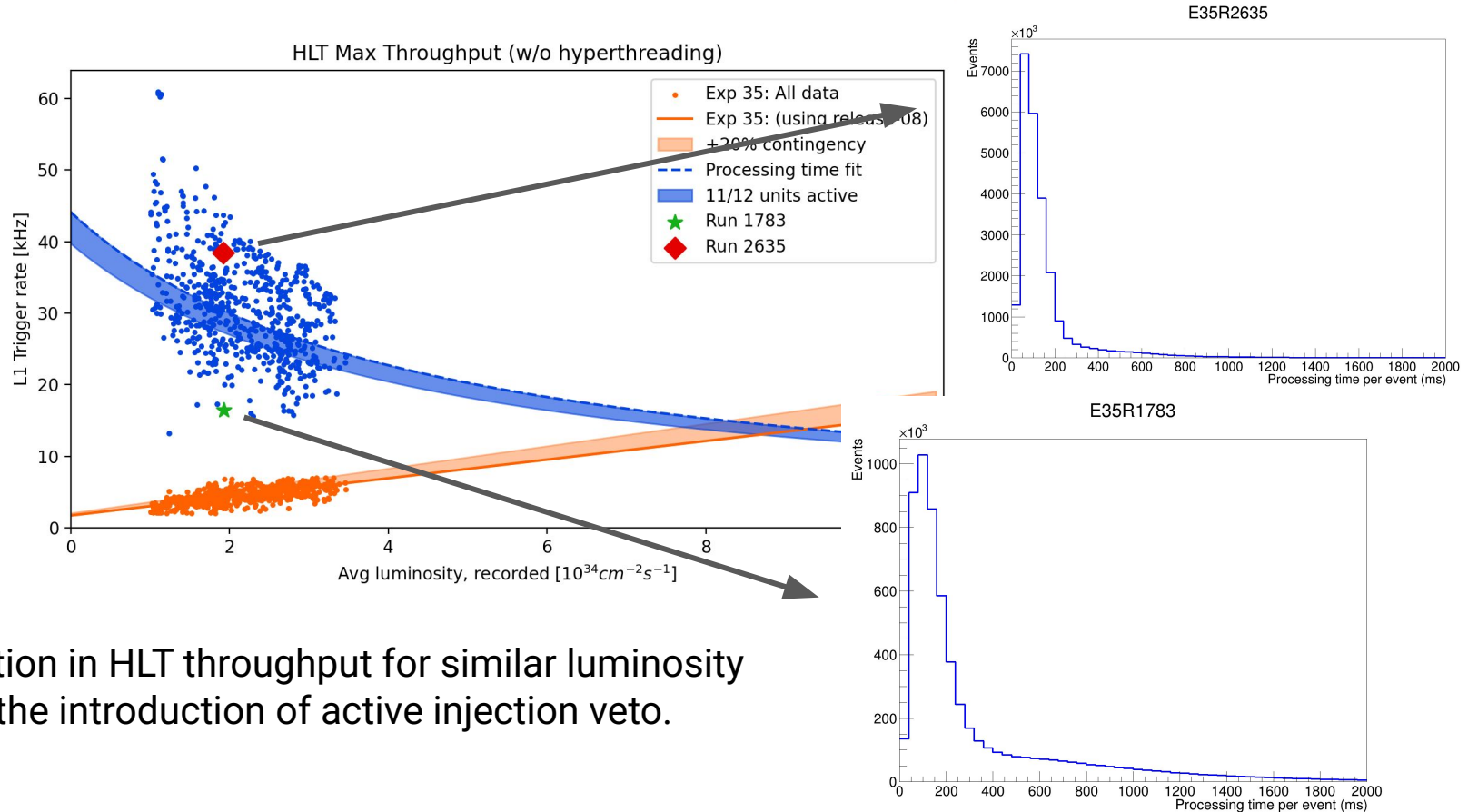
22nd Oct 2025



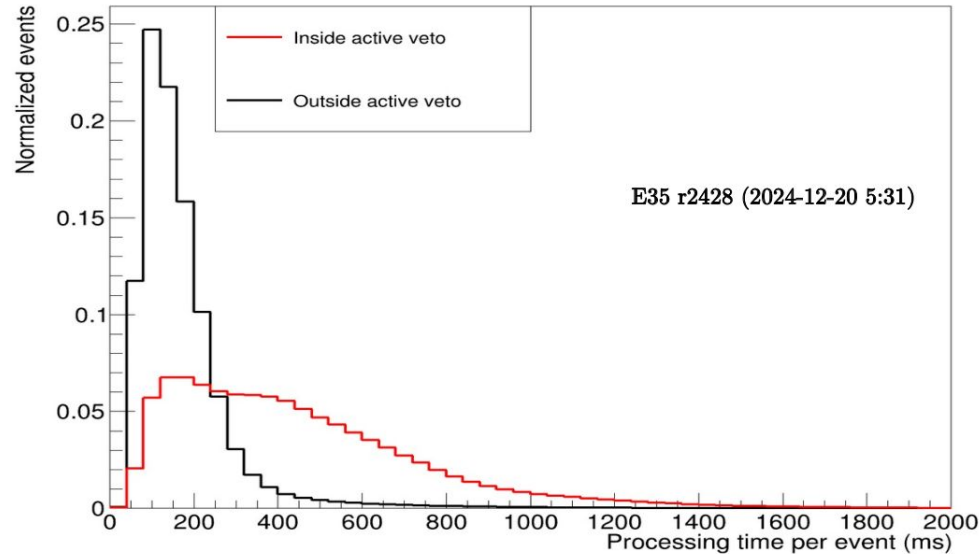
Outline

- Motivation
- Offline studies
- **HLTPrefilter** module
- **HLTPrefilter** monitoring
- Plans for **release-11**
- Summary

HLT performance (Exp 35)



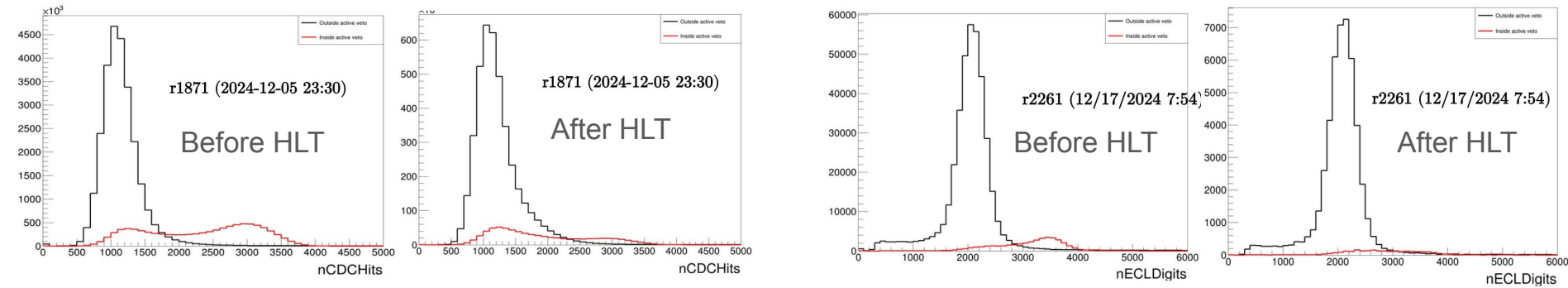
Processing time and Prefilter



*The active veto condition for 2025c will be the same as that of 2024c.

- Active veto was turned on from Exp 35 run 1145 (2024-11-08 18:33).
- HLT has to process extra events with higher processing time due to active veto.
- Vetoing events from the **active veto region** with lower physics usability can help us in smooth operation of HLT in higher luminosity.

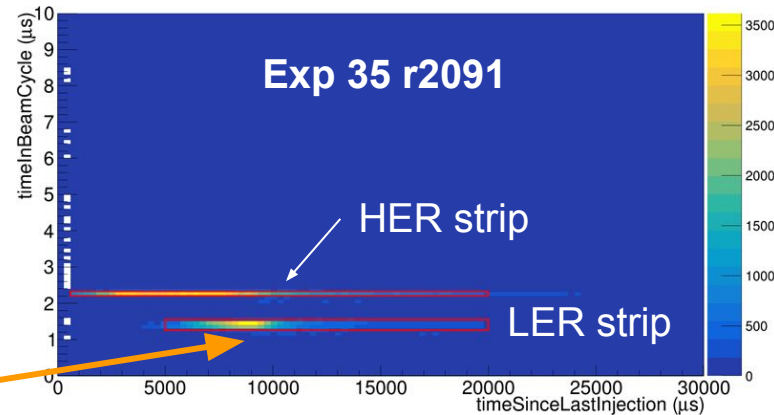
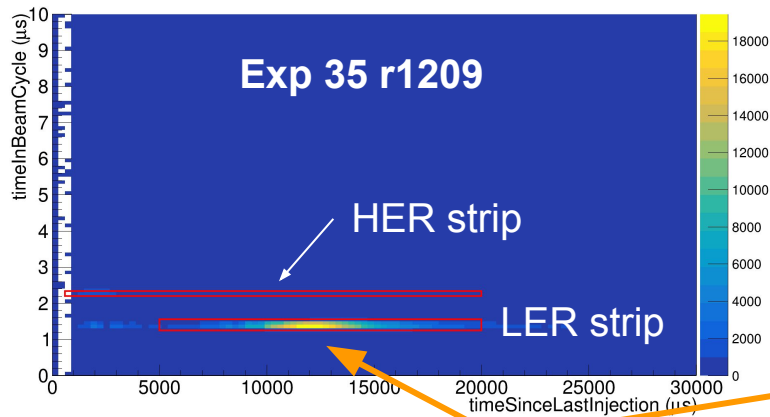
Prefilter based on nCDCHits and ECLDigits



- The events inside active veto (**red histogram**) have a significant portion of events with high nCDCHits and nECLDigits (a second peak) but the HLT filters most of them.
- **Prefilter**: Cut on nCDCHits and nECLDigits for events in **active veto region**.
- **Selection**: $nCDCHits < 2500$; $nECLDigits < 3000$
- **Impact**: On an average, see a drop in the HLT processing of around 15%.
- A similar veto is also applied at the L1 level on CDCTRG hits ($nCDCTRG > 1200$) which is different from the nominal nCDCHits (see backup).
- **Caveats**: Can have some impact on high-multiplicity physics events.

Prefilter based on timing cuts

- **Prefilter:** Remove the LER and HER injection strips.
- The resolution of HLT for these timing cuts is better than L1, because the L1 has large latency (details in backup).
- **Pros:** Logical successor of passive veto, cuts injection background.



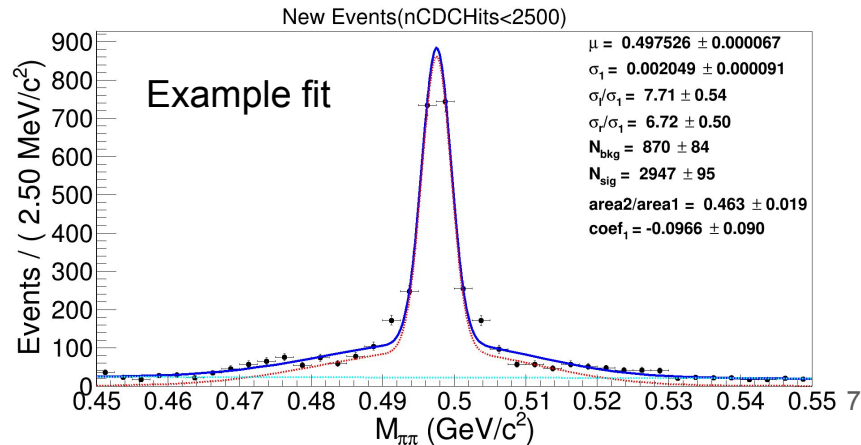
- To cover possible shift of injection strips, a generalized window is selected:
selection=(5000< timeSinceLastInjection<20000 && 1.25 <timeInBeamCycle<1.55)
|| (600 <timeSinceLastInjection<20000 && 2.2 <timeInBeamCycle<2.33)

Prefilter: Impact on physics

- The **Ks** serves as a proxy of high multiplicity physics events, with sufficiently high statistics and a clean signature.
- For this study, we select **Ks** from **KS0:merged** list and append **goodBelleKs** flag.
- A vertex fit is applied to the KS0 candidates using **TreeFit**.
- The following event level selections are applied to filter KS0 events:
HighLevelTrigger==1; HLT skim: software_trigger_cut&skim&accept_kshort.
- **Ks** yield is extracted by fitting $M_{\pi\pi}$ in the range: $0.45 < M_{\pi\pi} < 0.55 \text{ GeV}/c^2$

Exp 35 run 2086-2098

Selection	Ks yield gain (%)
Active veto (no prefilter)	3.7 ± 0.2
Prefilter (nECLHits && nCDCHits)	3.1 ± 0.1
Prefilter (timing cuts)	2.9 ± 0.1

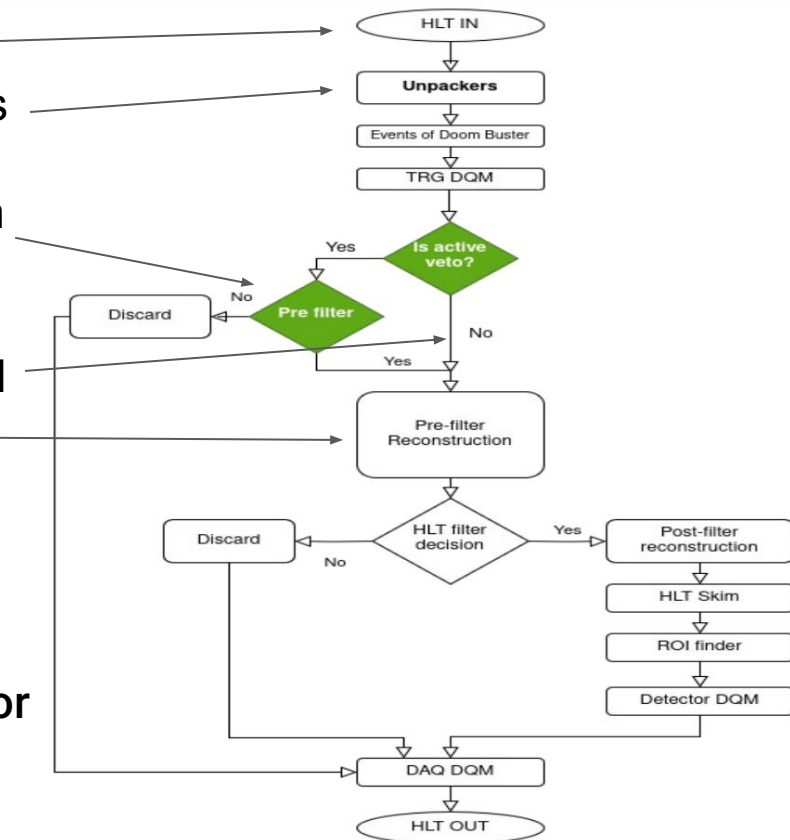


HLTPrefilter (plan)

- Data from L1 is fed into HLT
- Raw detector information of HLT input data is available after unpacking.
- For events inside active veto, some selection is applied (prefilter) to remove background dominated events.
- Events outside the active veto are unaffected
- After prefiltering, all the events go for HLT processing.

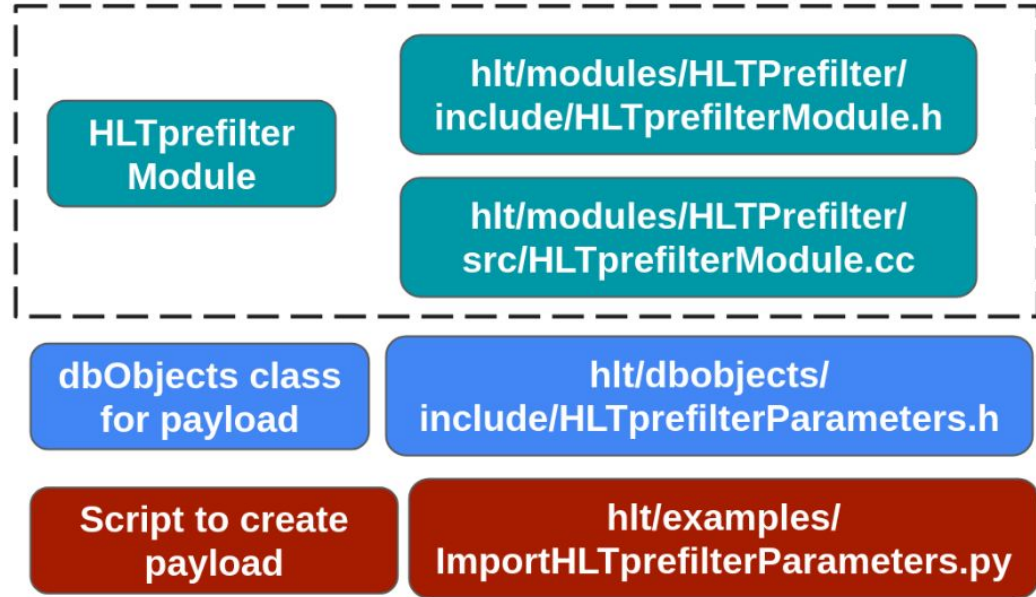
⇒ The plan is to implement a prefilter based on simple cuts on variables that are available before HLT processing.

⇒ The prefilter will operate in monitoring mode for 2025c period, can be switched On when required.



HLTPrefilter Module

- The module is executed in **add_hlt_processing()** function of [processing.py](#) script.
- Dedicated **payloads** to initialize module parameters (script: [ImportHLTprefilterParameters.py](#)).
- Relevant issue: [HLT prefilter module](#); MR: [feature/11159-hlt-prefilter-module](#)
- The MR is merged to “**main**” branch. (cherry-picked to **release-09-00-05**)



Payload for HLT filter lines for DQM (added to [main_2025-09-30](#)): [user_drahul_HLTPrefilter_FilterLines](#)
Names of filter lines: **HLTprefilter_InjectionStrip**, **HLTprefilter_CDCECLthreshold**.

- These filter lines have prescale set to zero so they **don't** interfere with **FinalTriggerDecision**.
- Payload of HLTPrefilter module is used to initialize parameters of these filter lines.

HLTPrefilter Operation

Relevant payloads for the **HLTPrefilter** module (**timing state** payload added to [main_2025-09-30](#)):

- [user_drahul_HLTPrefilter_State_Timing](#) : Initialize parameters of the module in timing cut state.
- [user_drahul_HLTPrefilter_State_CDCECL](#) : Initialize parameters for CDC-ECL cut state

Turn On/Off the **HLTPrefilter**
in **add_hlt_processing**:
(*default: monitor*)

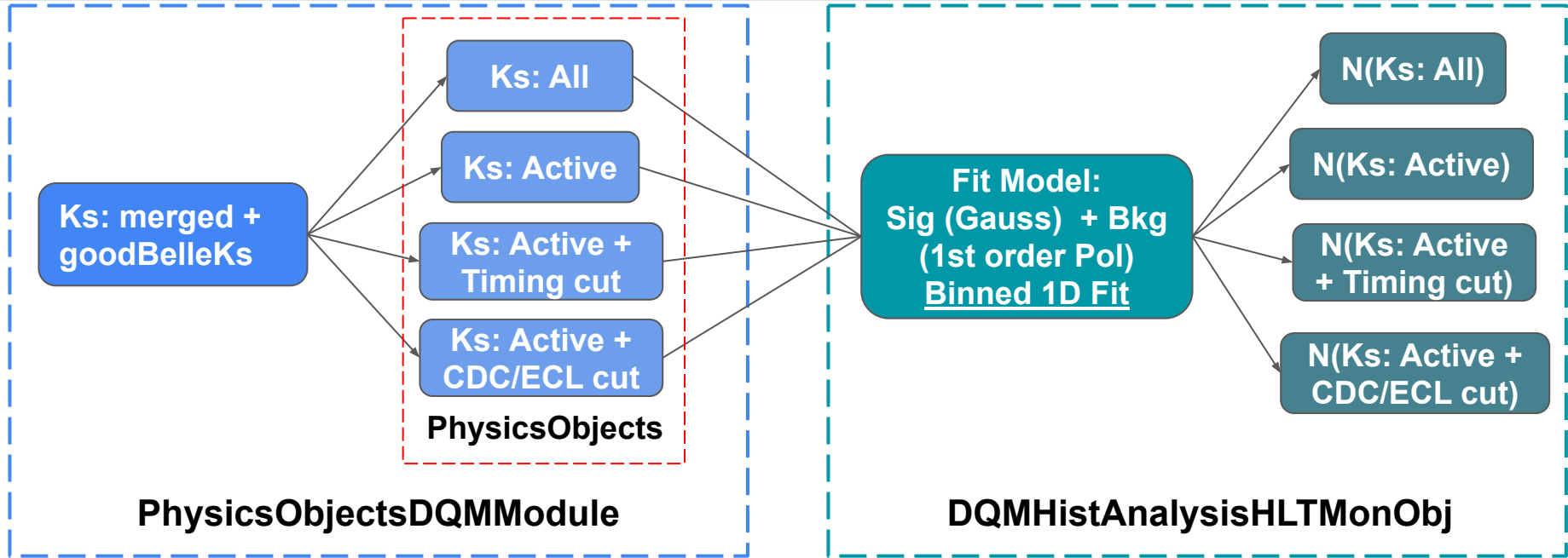


- **Monitor**: The HLTPrefilter will keep the events untouched, it is identical as having no prefilter.
- **Filter**: In this mode of operation, the **HLTPrefilter** will reject events based on the conditions set by the payloads. We use a prescale of 1000 to retain few events for offline cross-checks.

Example scripts for operation:

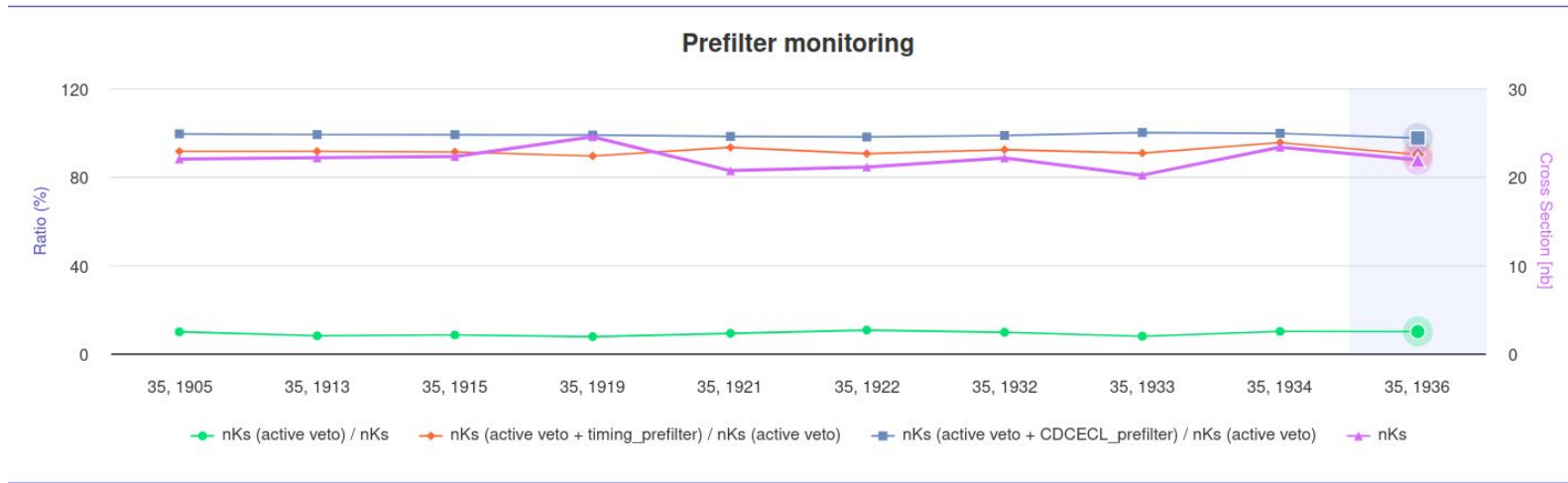
- [hlt/operation/zmq/hlt/beam_reco_filter_prefilter_time.py](#)
- [hlt/operation/zmq/hlt/beam_reco_filter_prefilter_CDCECL.py](#)

HLTPrefilter Monitoring



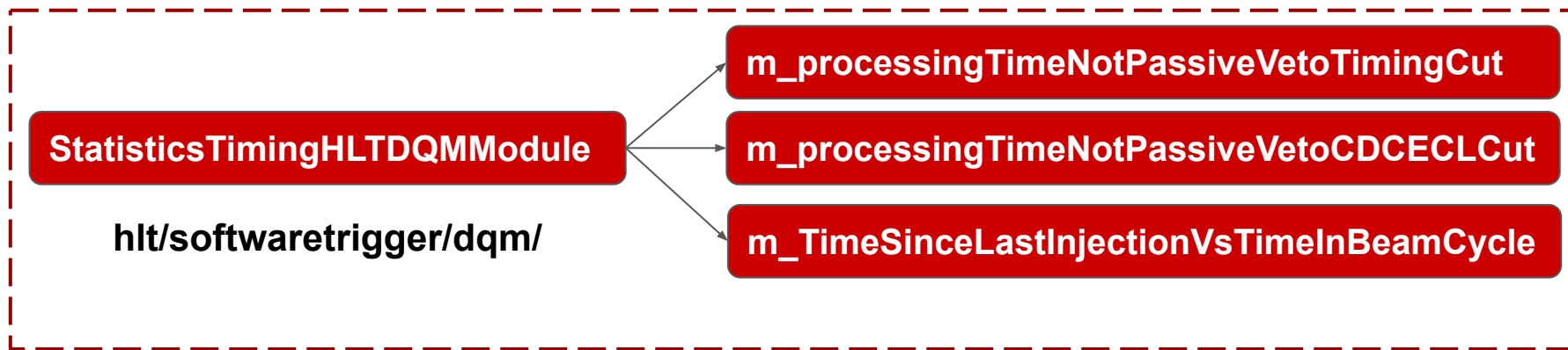
- We save the invariant mass of **Ks** events for different conditions in **PhysicsObjectsDQMModule**.
- A binned likelihood fit to the histograms is performed in **DQMHistAnalysisHLTMonObj** module to extract the number of **Ks** events.

HLTPrefilter DQM



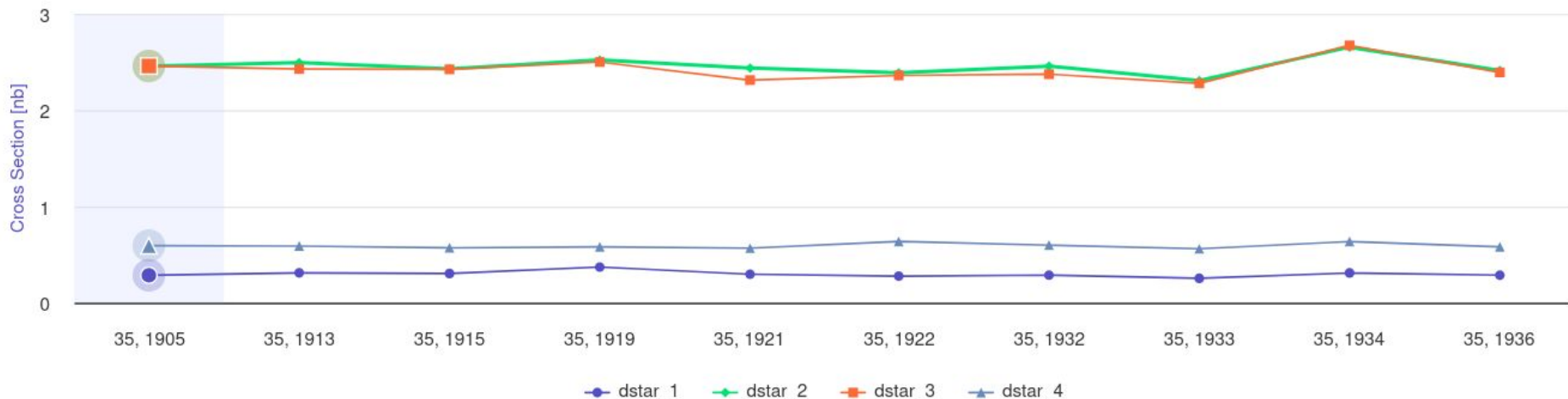
- MR to add **HLTPrefilter** related objects to **MiraBelle**: [mirabelle/-/issues/10](https://github.com/mirabelle/-/issues/10) (merged to **main**)
- Description of the monitoring objects using **Ks** events:
 - **nKs (active veto) / nKs** : Fraction of **Ks** events in active veto.
 - **nKs (active veto + timing_prefilter) / nKs (active veto)** : Retention of timing prefilter.
 - **nKs (active veto + CDCECL_prefilter) / nKs (active veto)** : Retention of CDCECL prefilter.
 - **nKs** : Effective cross-section of **Ks** events normalized to luminosity.

More monitoring...



- More histograms were added in the HLT DQM to understand the impact of HLTPrefilter conditions on the HLT processing time, and the variation of the injection strip.
- Description of the monitoring objects:
 - **m_processingTimeNotPassiveVetoTimingCut** :
HLT processing time after rejecting events failing the timing based prefilter condition.
 - **m_processingTimeNotPassiveVetoCDCECLCut** :
HLT processing time after rejecting events failing the CDC-ECL based prefilter condition.
 - **m_TimeSinceLastInjectionVsTimeInBeamCycle** :
2D plot of timing variables to monitor the position of injection strip.

More monitoring...



- Added a monitor for effective cross-section of existing D^{*+} skims of HLT normalized to luminosity.
- The skims are named depending on how we reconstruct the D^0 : **dstar_1** ($D^0 \rightarrow K^- \pi^+$), **dstar_2** ($D^0 \rightarrow K^- \pi^+ \pi^0$), **dstar_3** ($D^0 \rightarrow K^- \pi^+ \pi^- \pi^+$), and **dstar_4** ($D^0 \rightarrow K_s \pi^+ \pi^-$).
- By monitoring these skims, one can infer about the retention of physics events that are complementary to the **Ks** monitors of **HLTPrefilter** described in previous slides.
- Due to low statistics, for now we rely on simple counting method to calculate the effective cross-section of events passing these skims.

Integration of EventsOfDoomBuster into HLTPrefilter

- **Objective:** Merge the **EventsOfDoomBuster** module with the **HLTPrefilter** framework. (Target: release-11).
- **Architecture Overview:**
 - The merged module will become part of the HLT scripts.
 - It will be executed immediately after the unpacking stage.
 - The setup will follow the structure of the existing FilterCalculator module:
 - Evaluate individual **HLTPrefilter** lines.
 - Run the SoftwareTrigger module to reject events using these lines.
- **Payload Requirements:** We will need two sets of payloads
 - To configure parameters of each **HLTPrefilter** line.
 - To create the **HLTPrefilter** line and define its prescale.
- **Outcome:** A fully functional **HLTPrefilter** menu, analogous to existing HLT menu.
- One can include new (fast) filtering conditions to the **HLTPrefilter** menu to further reduce background dominated events and improve HLT throughput.

Summary

- The setup for the module is complete, all necessary components in place.
- Link to GitLab issues: [HLT prefilter module](#), [mirabelle/-/issues/10](#).
- The **MR** are merged, and the module is cherry-picked to **release-09-00-05**.
- **Future plans:** Merge **EventsOfDoomBuster** module into the **HLTPrefilter**.
 - Establish a configurable **HLTPrefilter menu**, with payloads for parameter setup and prescale definition, enabling inclusion of new (fast) filtering conditions to reduce background and improve HLT throughput.



Backup



Active veto

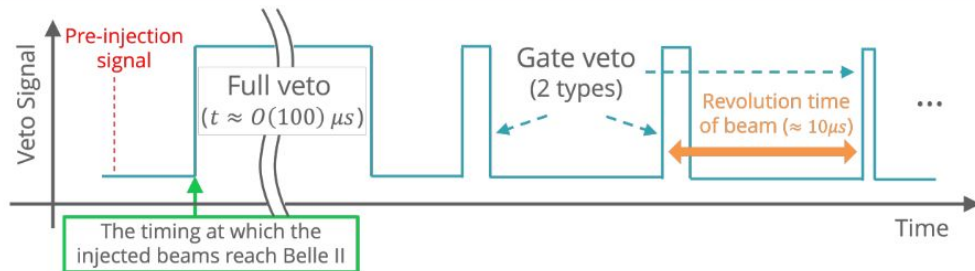
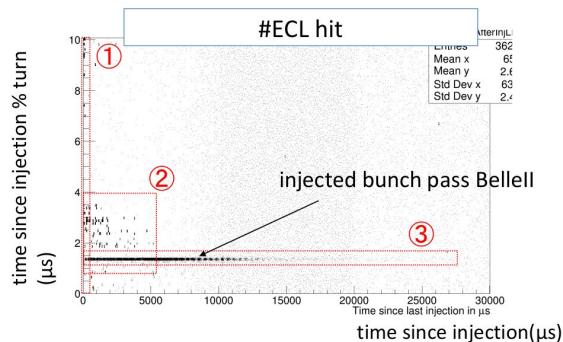


Fig. 1. A schematic explanation for the veto signals from the injection veto

-Injection veto structure

- ①~0.5ms: full veto (same as Belle)
- ②0.5~10ms: gate veto1 (since 2019)
- ③10~30ms: gate veto2 (since 2020)



New scheme

However, the effects of the injection background on the Belle II detector may vary turn by turn of the injected beams due to the betatron oscillation, and the current scheme cannot deal with such variation. This makes redundant deadtime for the Belle II data acquisition system, which reaches 15% at maximum in the beam operation in 2022. To solve this problem, we developed a new scheme for the injection veto system that considers the signals from CDC and ECL for issuing the veto signal because the injection backgrounds make gammas and electrons a few times more than the typical collision events, and the subdetector are sensitive to these particles. We implemented the logic circuit for the new veto scheme on Virtex UltraScale FPGA™ from Xilinx (XCVU160). The new veto signal NV is given by the following logical combination of the full and gate veto signals and criteria for the number of hits from the CDC and ECL as follows:

$$NV = \text{Full} \vee [\text{Gate} \wedge \{ (N_C > T_C) \vee (N_E > T_E) \}] \quad (1)$$

Resolution of L1 for timing selections (input from Koga-san)

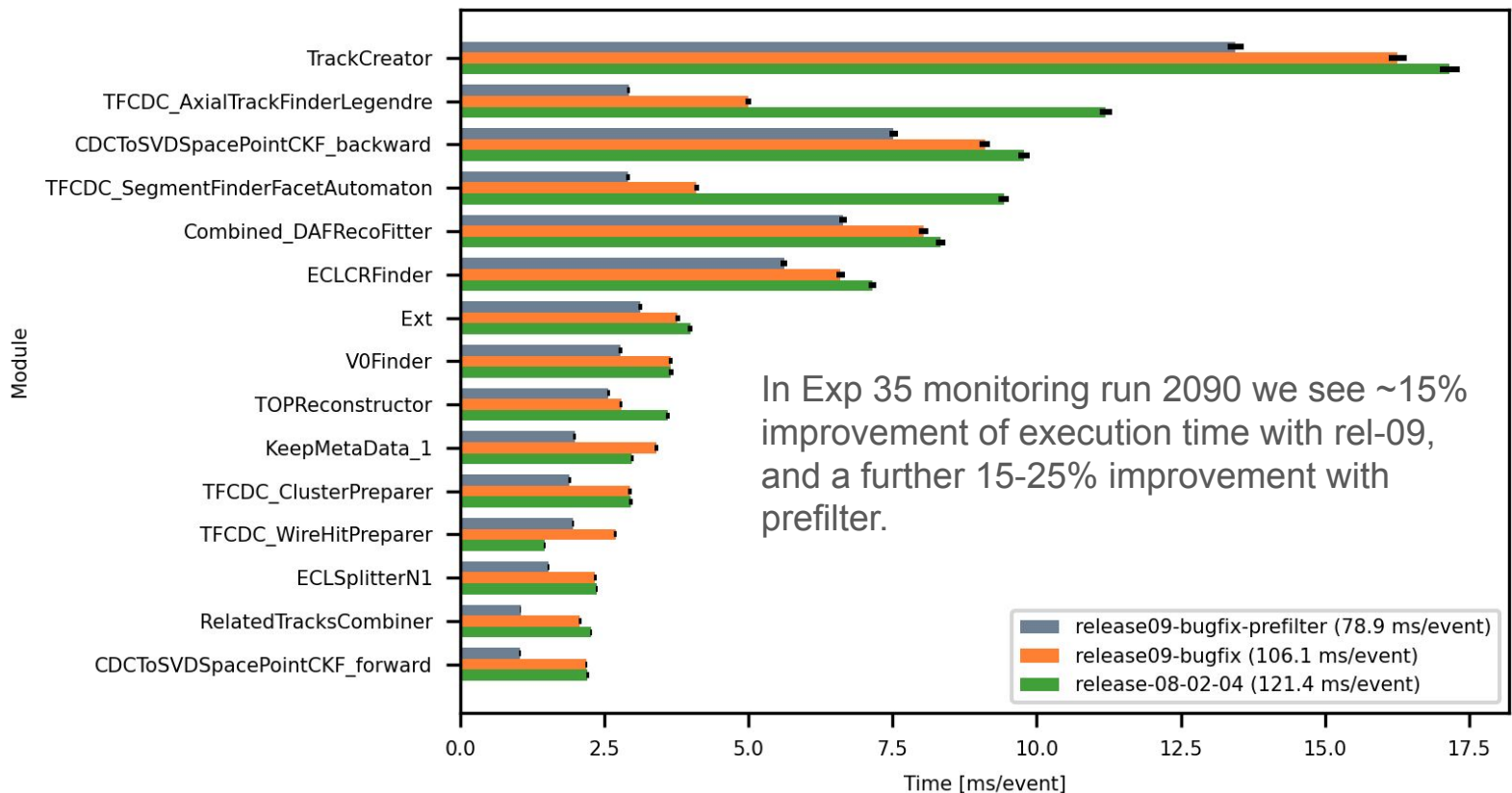
- At the L1 trigger level, one can not apply finer selections on the timing variables (timeSinceLastInjection, timeInBeamCycle_us) for the following reasons:
- Injection veto is applied in Logic calculation (FTDL) (see page 34 in [slide](#)). In this stage, L1 timing is not decided yet and our event timing resolution (jitter from sub-trigger) is $0.5\sim 1\mu\text{s}$. So, the injection veto window should be large enough to cover the jitter.
- The GDL algorithm has been modified, response of CDC and ECL detectors are slow. The CDC has $\sim 800\text{ ns}$ drift time, and ECLTRG has $\sim 500\text{ ns}$ shaping time. If the injection background causes high occupancy on CDC and ECL, it takes time to vanish all hits due to the slow response. In such case, the remaining hits may cause trigger with wrong event timing.

CDC vs. CDCTRG (input from Koga-san)

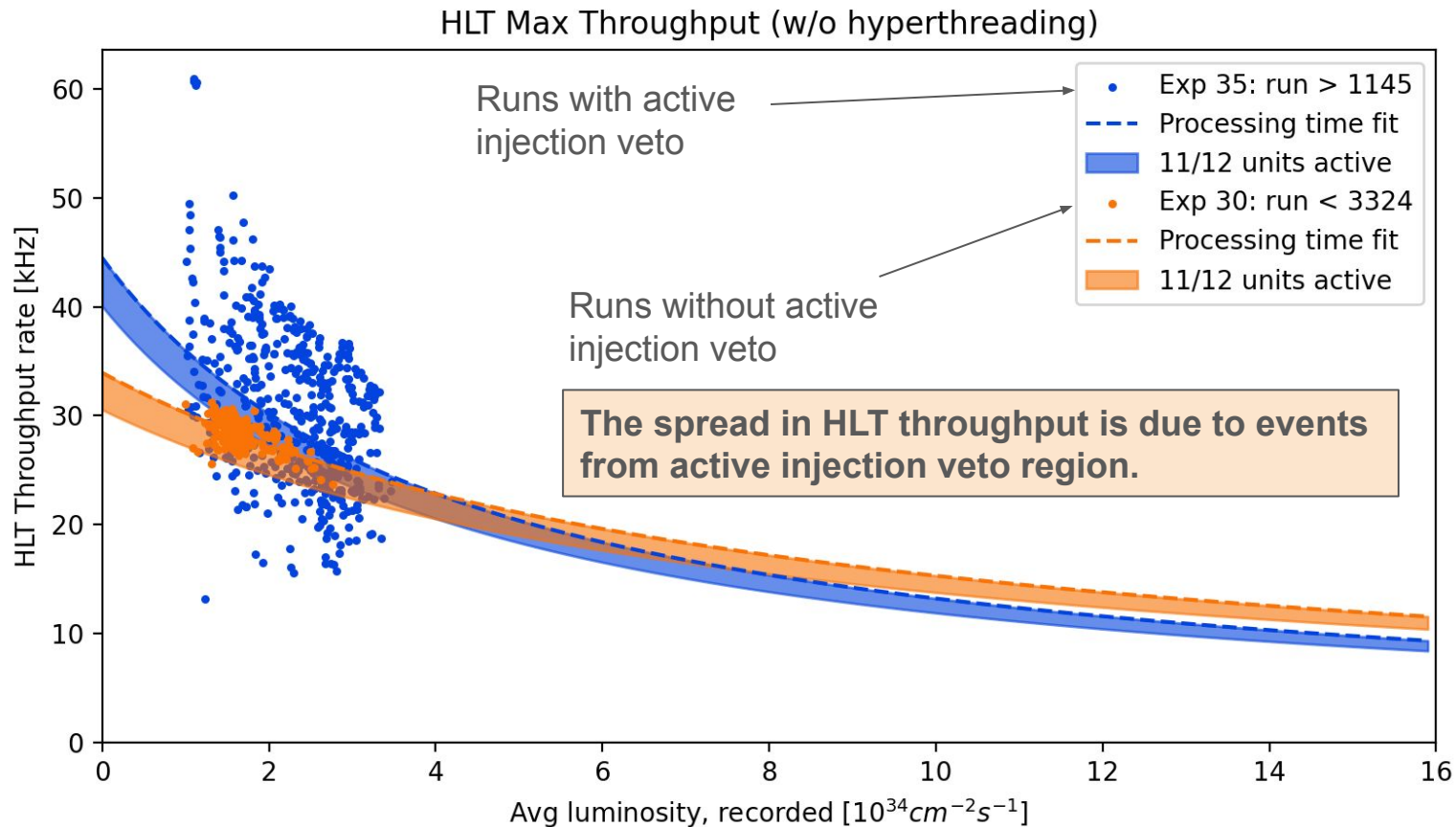
- CDC uses all the super layers, whereas CDCTRG uses following layers:
 - SL0: layer3-8
 - SL1-6: layer 0-4
 - SL7-8: not used
- CDC is not using ADC cut for the nCDCHit. On the other hand, CDCTRG uses $TRGADC > 6$.
- The TRGADC is a bit different from ADC of CDC (the number of sampling points is different).
- CDC is using 800ns timing window for B2L readout. CDCTRG is using 500ns timing window for the nCDCTRG (nTSF) counting. The width and offset are different.

HLT execution time: Exp 35 Run 2090

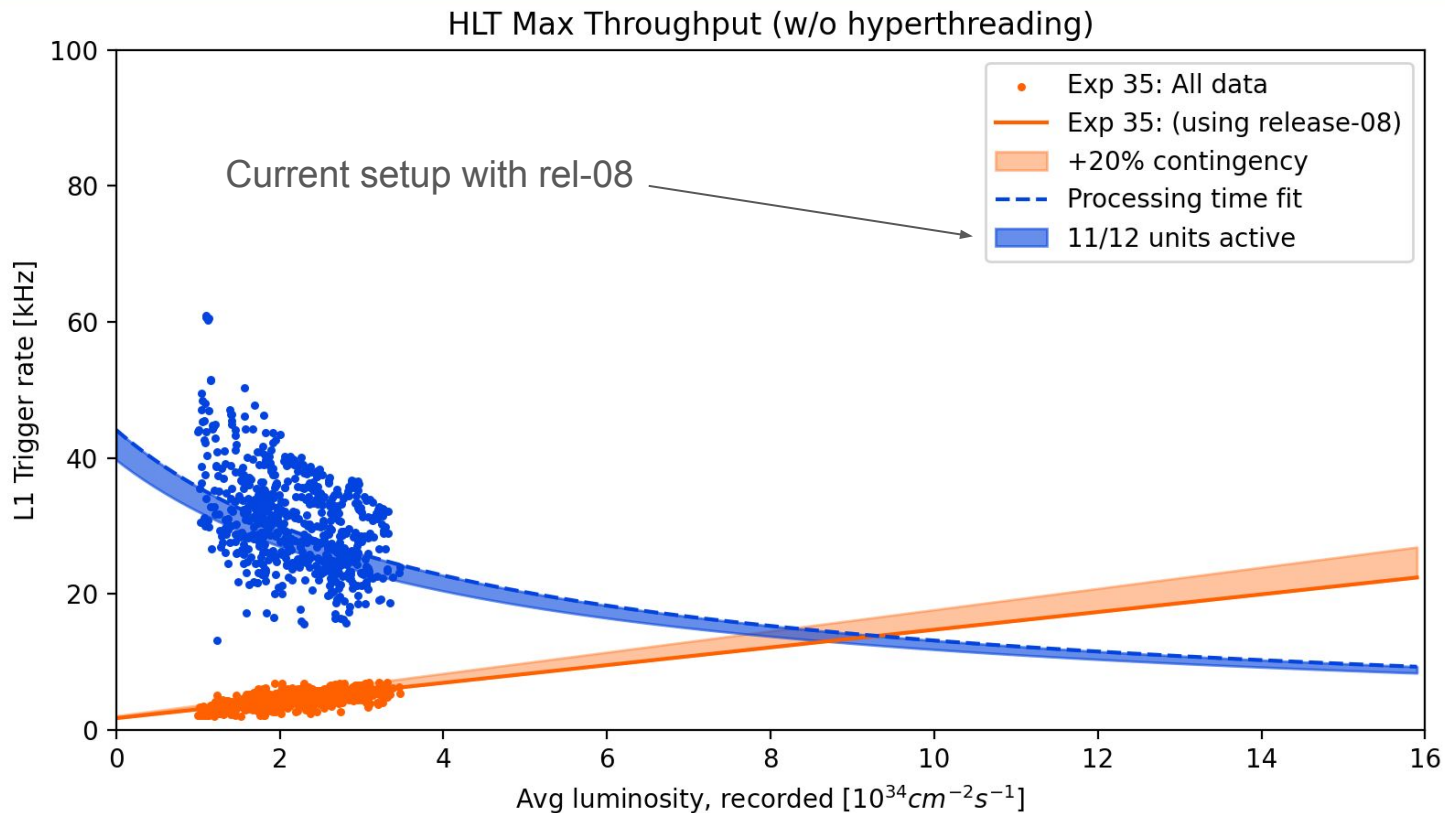
Ex 35 Run 2090 (10000 events)



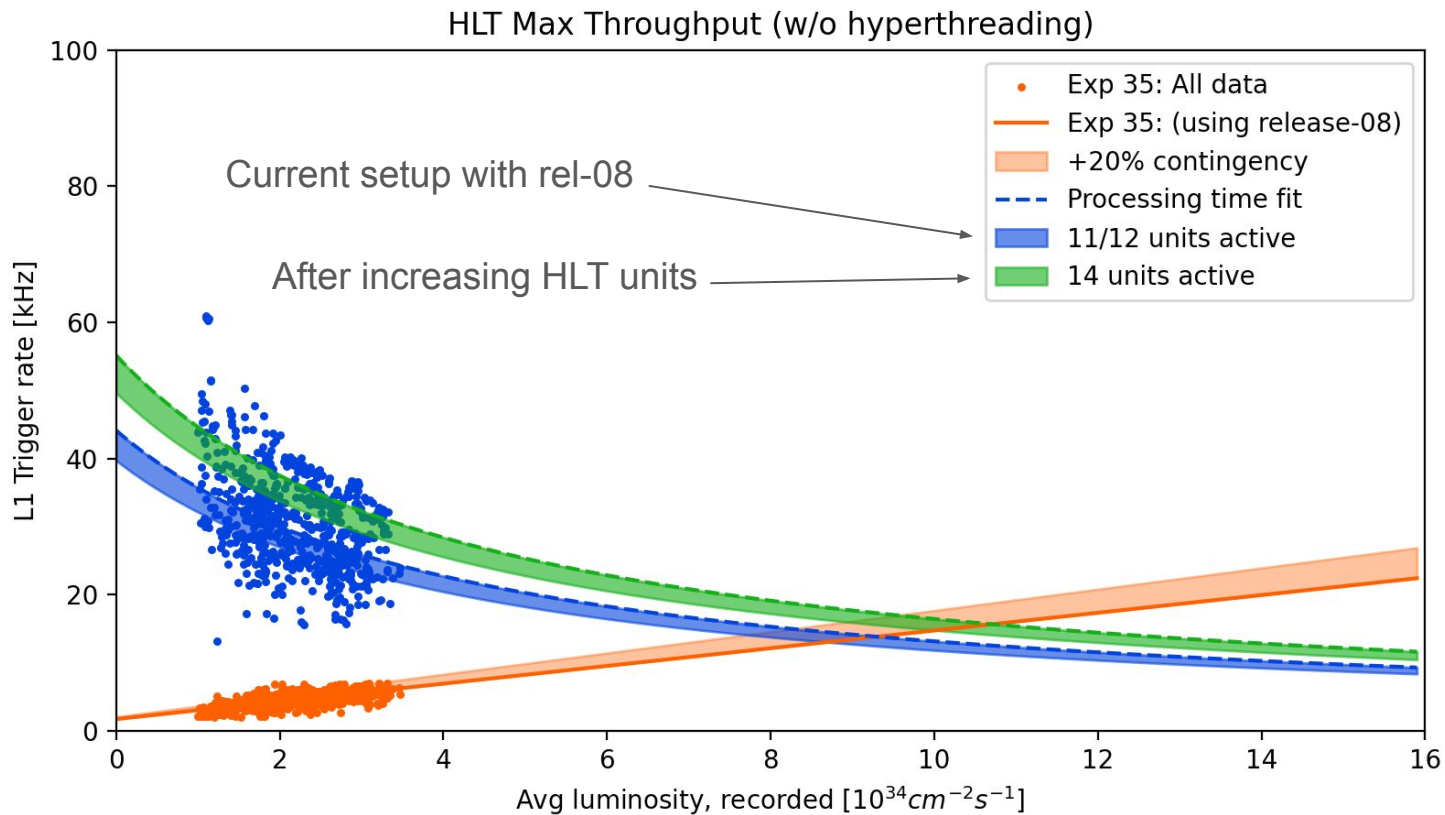
HLT throughput: Impact of active veto



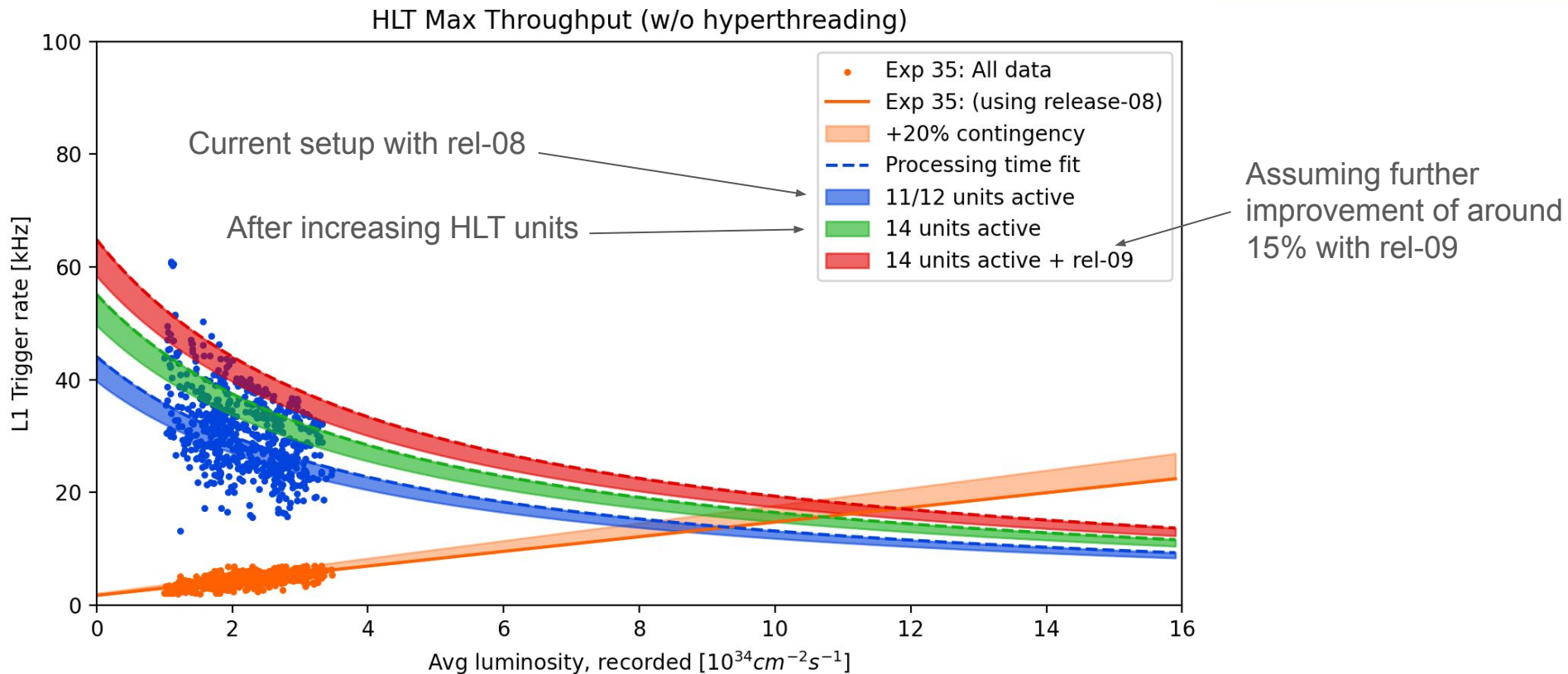
HLT performance in Exp35: Extrapolations



HLT performance in Exp35: Extrapolations



HLT performance in Exp35: Extrapolations



HLT performance in Exp35: Extrapolations

