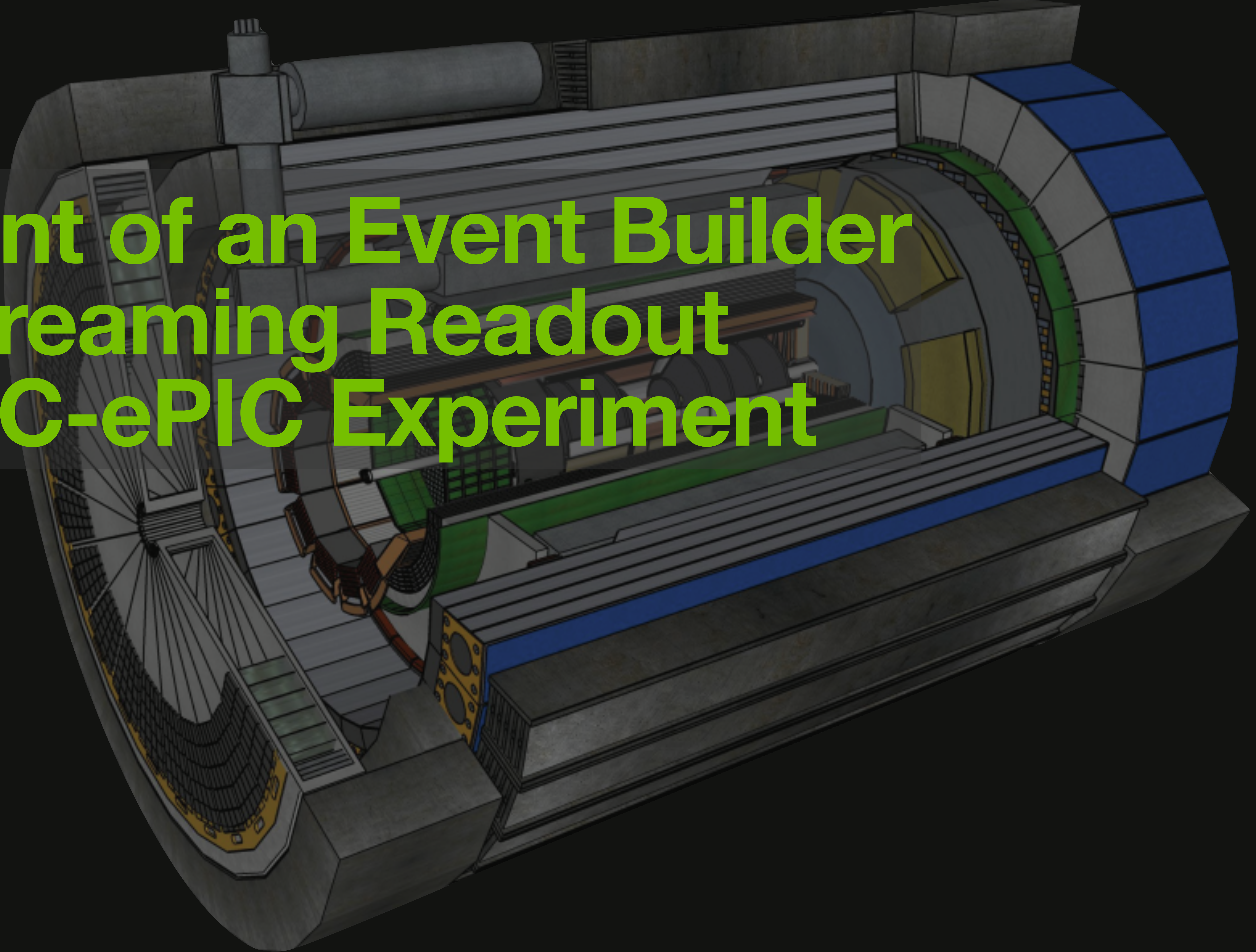
Development of an Event Builder with Streaming Readout for the EIC-ePIC Experiment



QNSI Quark Nuclear Science Institute, UTokyo | ePIC

The University of Tokyo
Takuya Kumaoka

Readout Data Flow

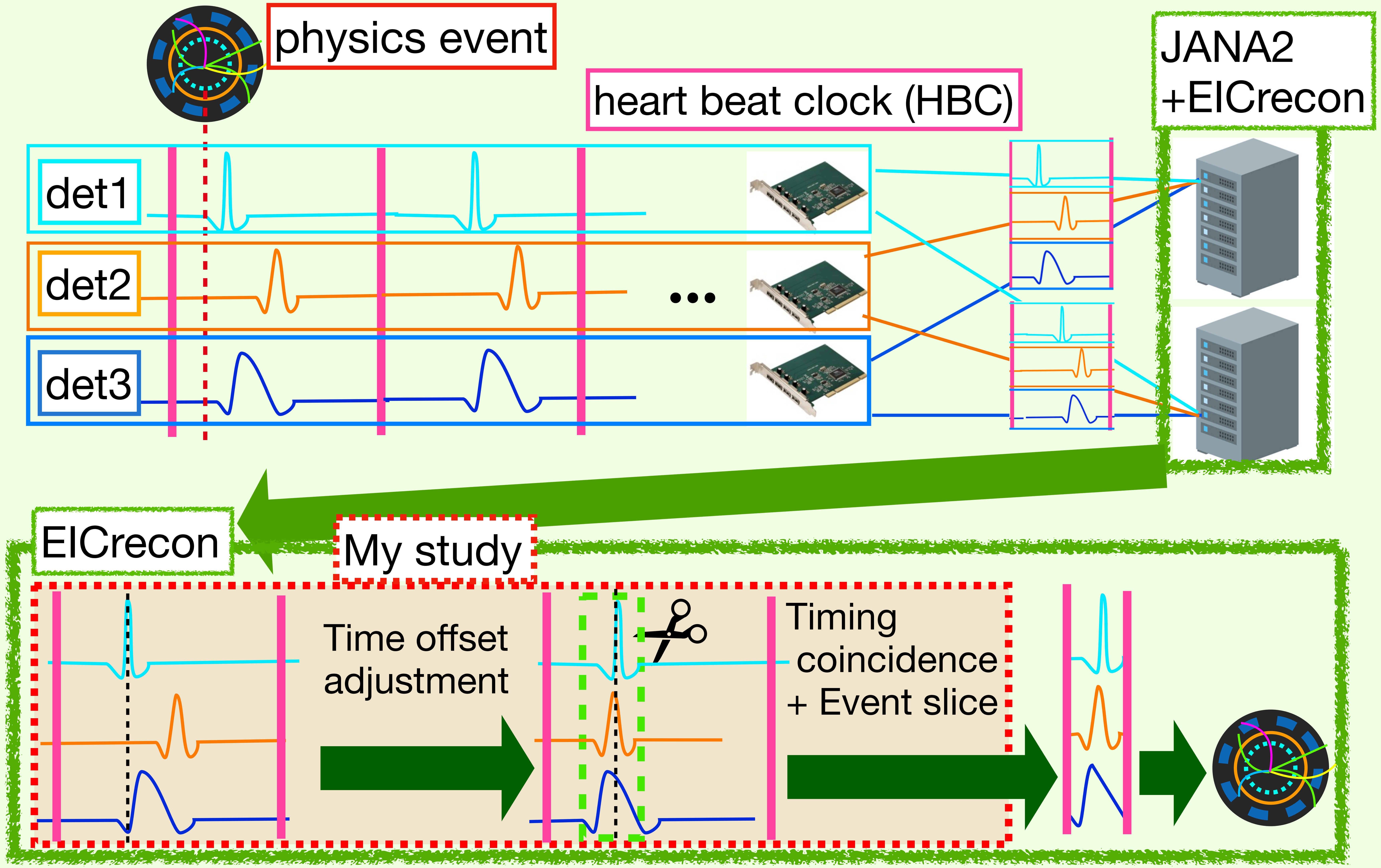
- The continuous signal is segmented by the Heart Beat Clock (common clock).

- The digital signals are sent to servers.

- Integrate all detector data.

- There are extraneous data.

- By building events, the data can be compact.



My study

Event extraction from streamed data and reconstruction into event data.

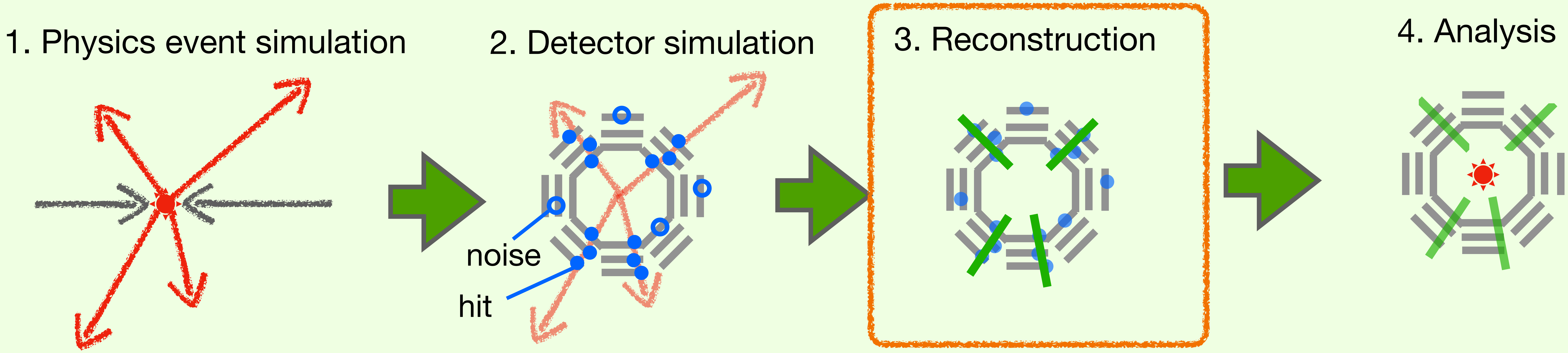
Simulation Flow

1. Event Generator (e.g. PYTHIA) → eventGene.hepmc

2. Detector simulation (npsim [Geant4 base]) → detSim.edm4hep.root

3. Reconstruction (eicrecon) → recon.edm4hep.root
   - Digitization
   - Event Selection ← My work
   - Reconstruction

4. Physics Analysis

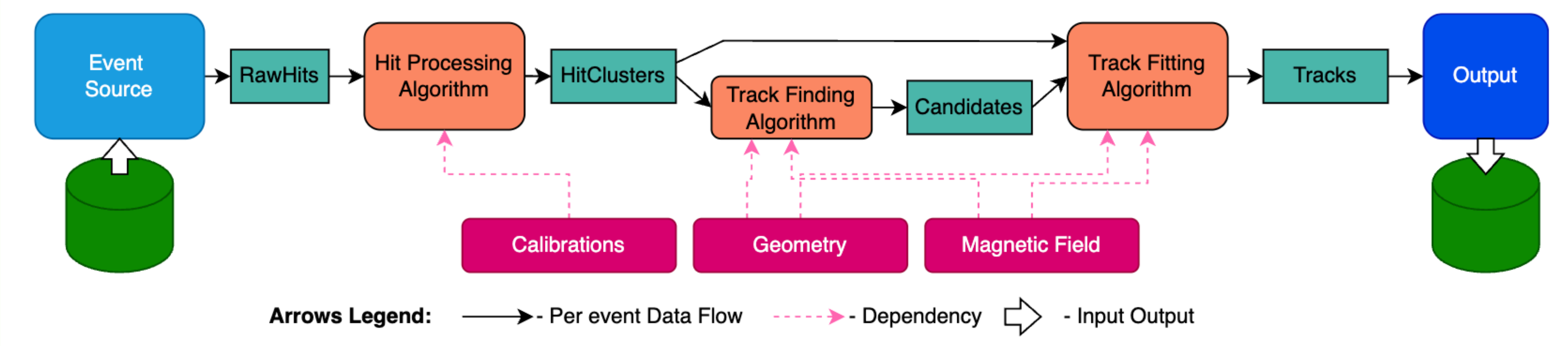


EICrecon Framework

Software for **reconstructing** particle information from simulations or real hit data for physics analysis based on JANA2.

- EICRecon Reference Page: (Main, git, tuto1, tuto2, tuto3).

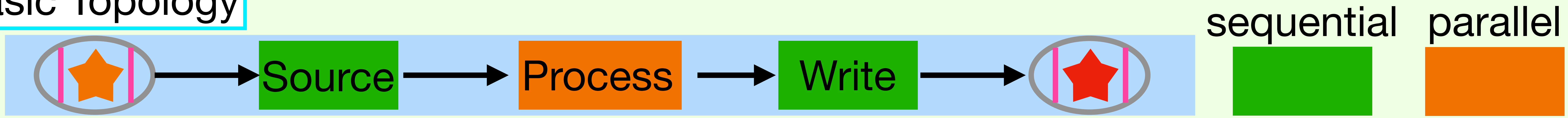
- Code structure: JANA2 (File reader, Reconstruction, File writer)



- Data Structure: edm4eic (git) ← edm4hep (git) ← podio (page)
  yaml → (compile w/ podio) → C++ class codes

JANA2 Data Flow

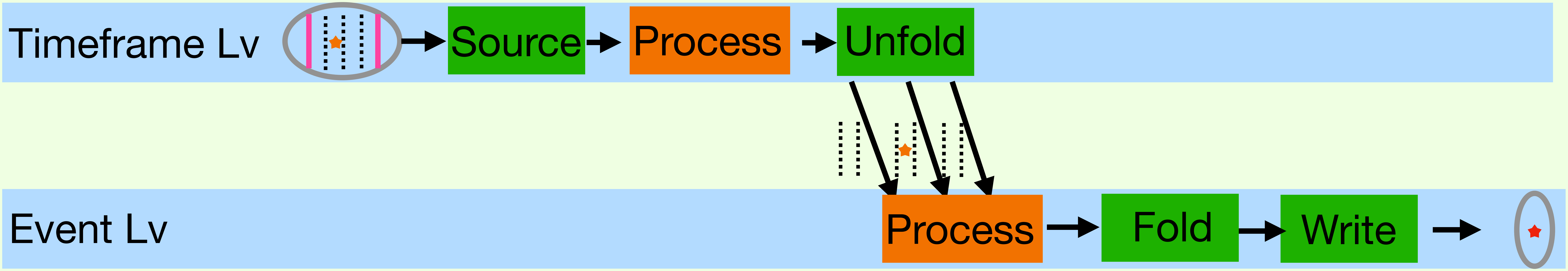
Basic Topology



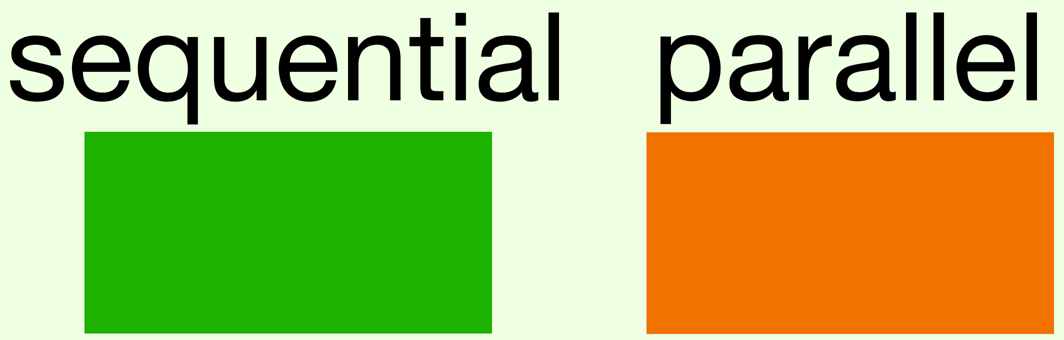
If the input data contains only one physics event, a single-level process is sufficient.

Timeslice Splitting Topology

※ Unfold: Split one frame into multi frames



A split time frame within a time slice needs to be processed.
→ It should be handled in parallel at the event level.



My Study Targets

Map of functions to be developed

Progress and Plan

1. Create a new factory to do time alignment (**Done**)

2. Unfolding trivial test (**Done**)

3. Time splitting test (**On going: End of Jun**)

4. Timing Coincidence (**Mid of July**) ← Collaboration meeting?

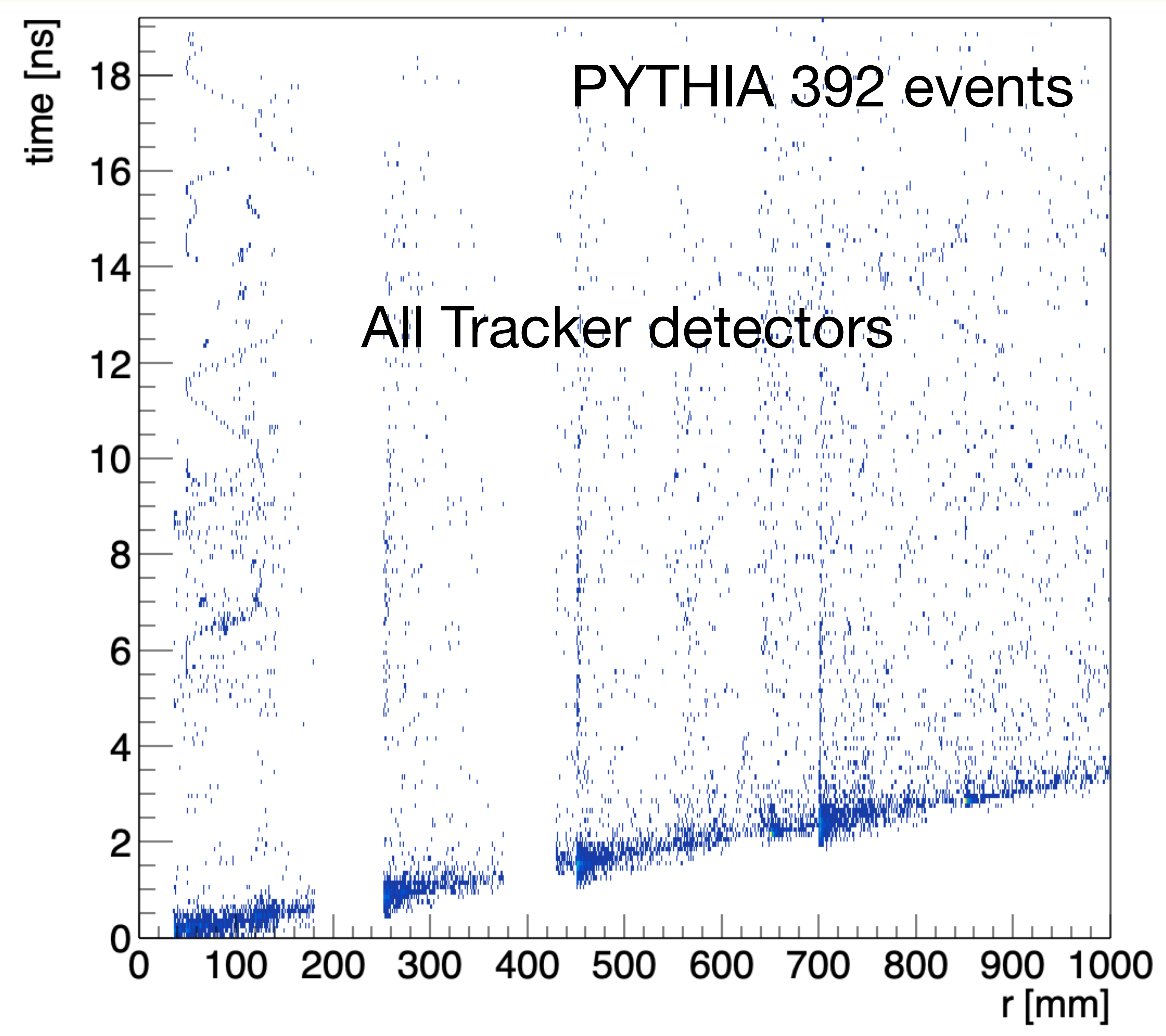
5. Injection of background events and detector noises and evaluation of timing coincidence (time windows vs. rejection, efficiency, primary tracking, etc) (Mid of August)

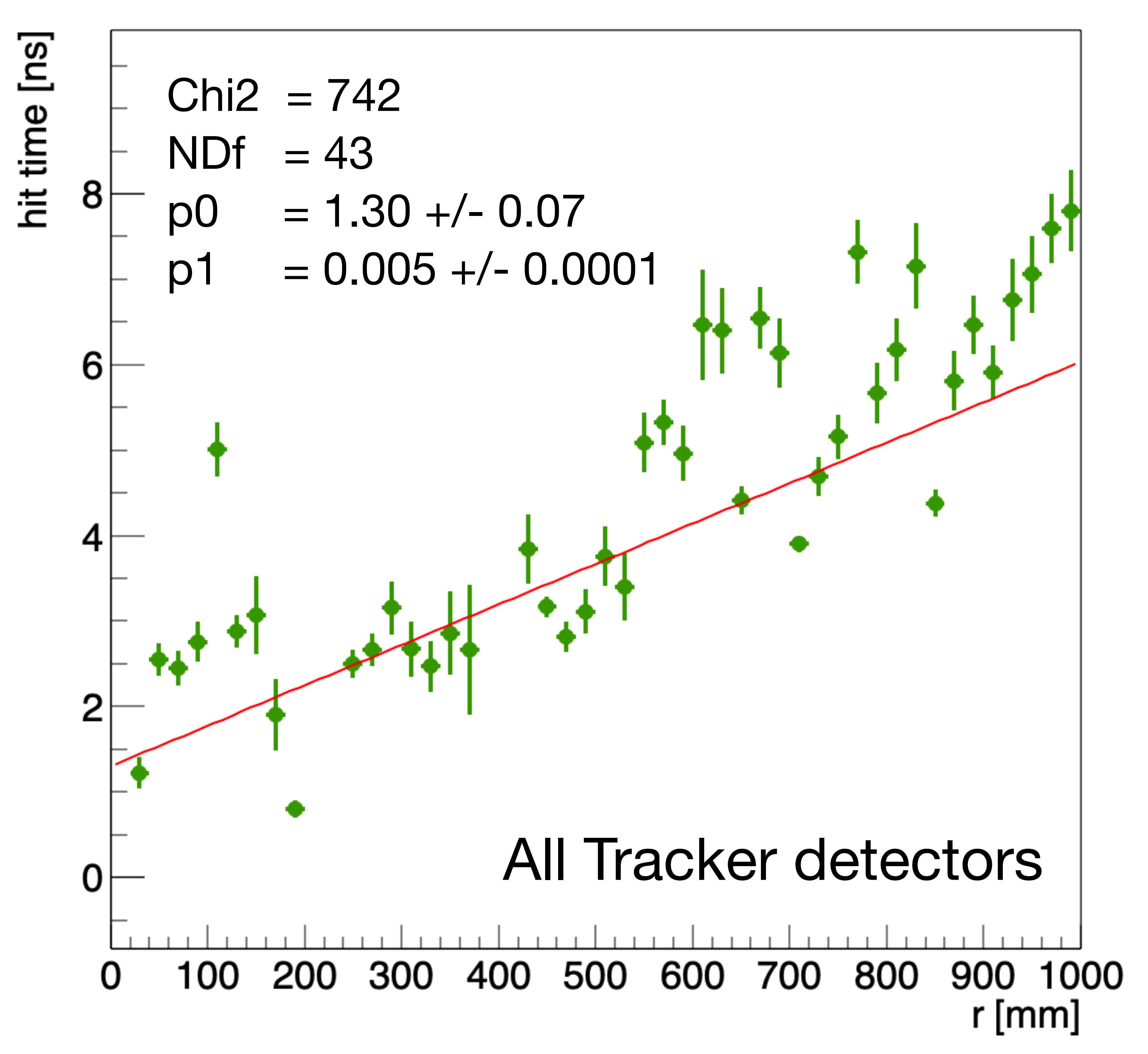
6. Apply detector response (September?)

7. Optimize time window, Time alignment, Interval selection (October?)

1.1 Hit time vs Hit position (Time Alignment)

These plots show the relation between hit time vs hit position to alignment hit time.





→A correlation between distance and hit time is observed.
However, since we are looking at MC hits, this result is expected (W/O detector response).

1.2 Time Alignment

Create Factory to do time alignment
→ That value itself has no meaning.

Original input data

```
HitChecker: Event 188640 Hits in: 13.1356, 1.12963, 1.2371, 1.03809, 1.02581, 24.2638, 10.5798, 9.64017, 1.77883,
1.93878, 1.63005, 1.61152
```

Sort + time alignment ($t' = t$ - 0.003 [ns/mm] * R [mm])

```
HitChecker: Event 188640 Hits in: 0.863942, 0.867679, 0.889219, 0.943039, 1.06582, 1.37656, 1.41053, 1.49302, 1.5067,
1.66467, 1.68391, 1.76771
```

It seems work well.
→ **Future plan: Time response of detectors will be included.**

2.0 Create Unfold Code

JEventUnfolder (JANA2):

Base of unfold code. It handles to IDs of a time frame (parent) and events (child).

MyTimesliceSplitter (JANA2, unfolding tutorial):

This code is to test JEventUnfolder.
It deparses each time region as event level.
However, it cannot be used in EICrecon directly.

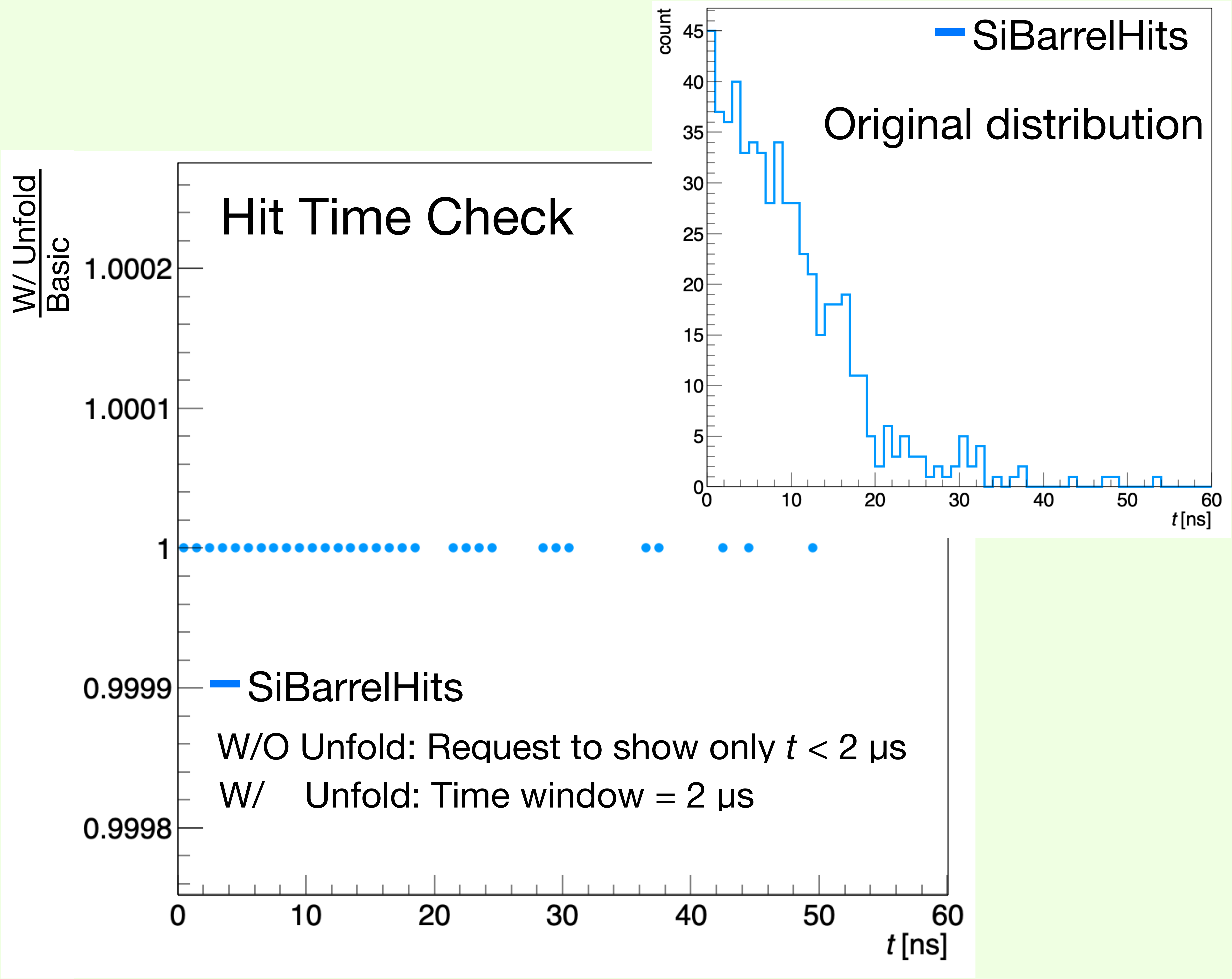
TimeframeSplitter (EICrecon):

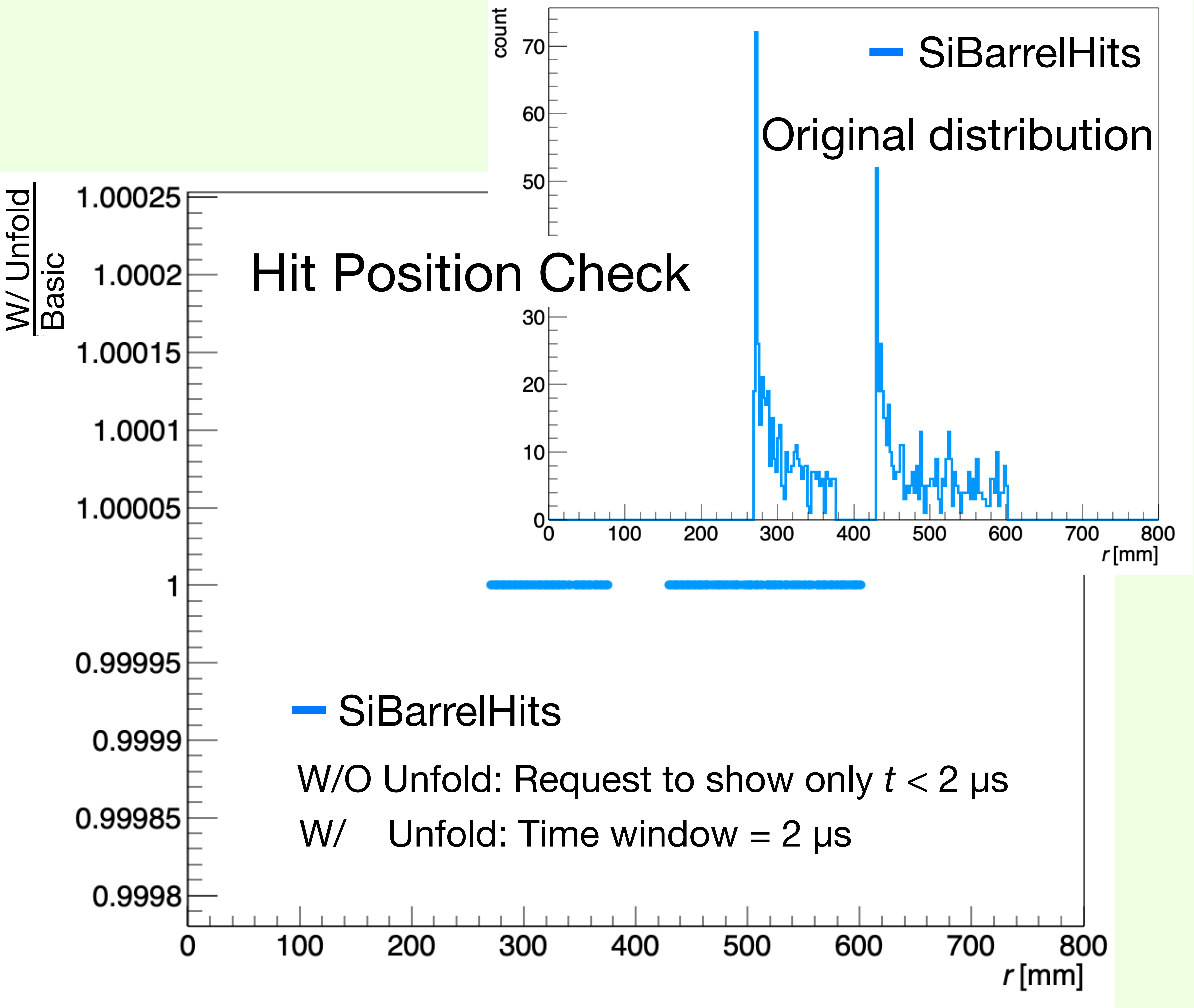
This code performs the actual time frame splitting based on JEventUnfolder.

Need to implement time-splitting function without losing physics events.

2.1 Compare Original or W/ Unfolding for Position and Energy

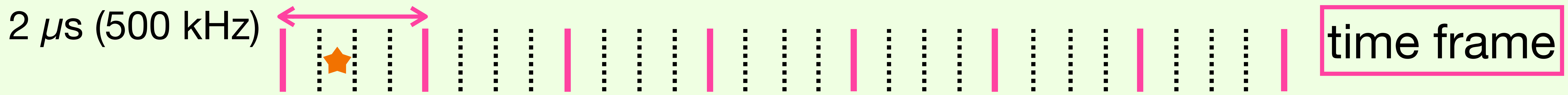
Trivial test: Compare the outputs between the basic topology and the unfold topology





The results show a complete match between cases W/O and W/ unfolding.
These results are reasonable.

3. Time Splitting results




```
23:44:23.426  [info] Timeslice
HitChecker: Event 188640 Hits in: , 1.02581, 1.03809, 1.12963, 1.2371, 1.61152, 1.63005, 1.77883, 1.93878, 9.64017, 10.5798, 13.1356, 24.2638
```

Input data in one time frame

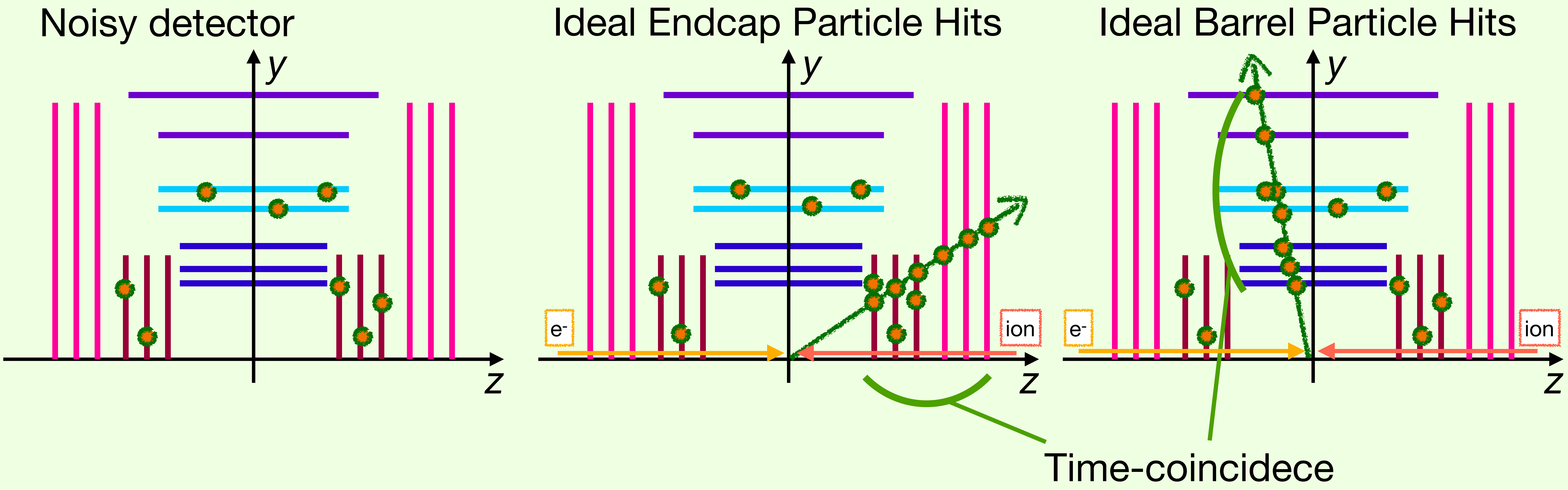
```
child_idx = 0:: TimeframeSplitter: timeslice 188640 iTimeSlice 0 eTimeSlice 4
23:44:23.426  [info] PhysicsEvent
HitChecker: Event 188640 Hits in: , 1.02581, 1.03809, 1.12963, 1.2371, 1.61152, 1.63005, 1.77883, 1.93878

child_idx = 1:: TimeframeSplitter: timeslice 188640 iTimeSlice 4 eTimeSlice 8
23:44:23.925  [info] PhysicsEvent
HitChecker: Event 188640 Hits in:

child_idx = 2:: TimeframeSplitter: timeslice 188640 iTimeSlice 8 eTimeSlice 12
23:44:24.267  [info] PhysicsEvent
HitChecker: Event 188640 Hits in: , 9.64017, 10.5798

child_idx = 3:: TimeframeSplitter: timeslice 188640 iTimeSlice 12 eTimeSlice 16
23:44:24.590  [info] PhysicsEvent
HitChecker: Event 188640 Hits in: , 13.1356

child_idx = 4:: TimeframeSplitter: timeslice 188640 iTimeSlice 16 eTimeSlice 20
23:44:24.889  [info] PhysicsEvent
HitChecker: Event 188640 Hits in:
```

separate into 4 ns
→This value itself has no meaning now.
It should be determined by hit time distributions.

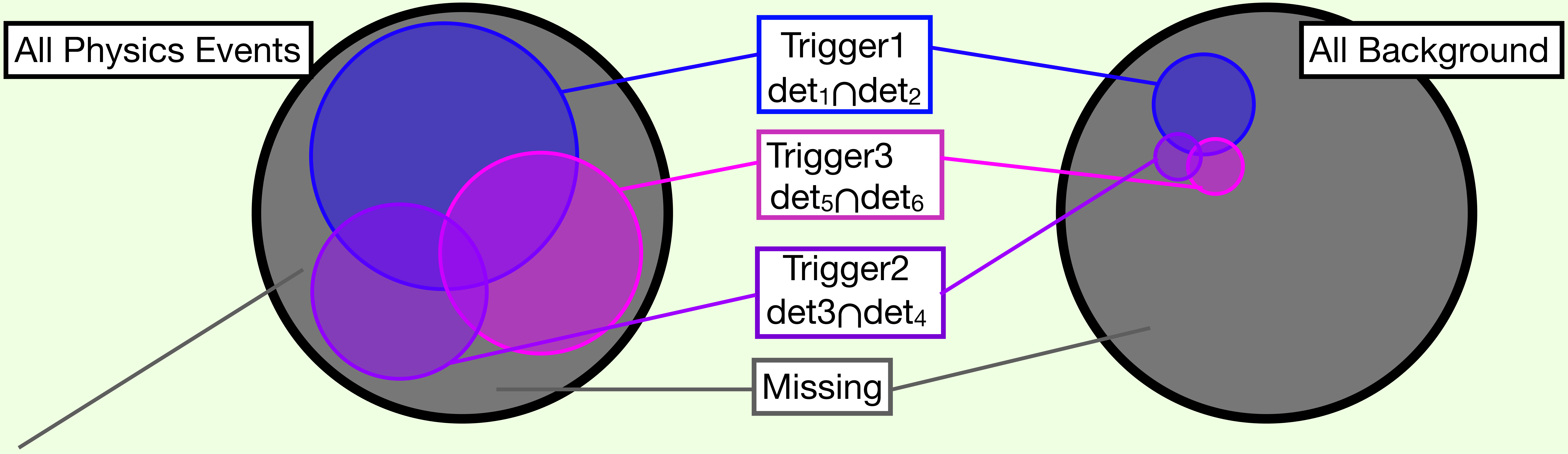
As the time split test, it works well.

Next: Develop algorithms to decline the intervals of empty or small hits. (on going)

4.1 Time Coincidence



Even if a detector can capture physics events with high efficiency, it must still be time-coincident with other detectors when there is significant noise.



Search for the detector the combination which can capture the remaining physics events

Reduce wrong triggers
→ Strong requirement (number of hits, hit timing, number of detectors)

Make a map of triggers
(ex: trig1, trig2, trig1∩trig2….)

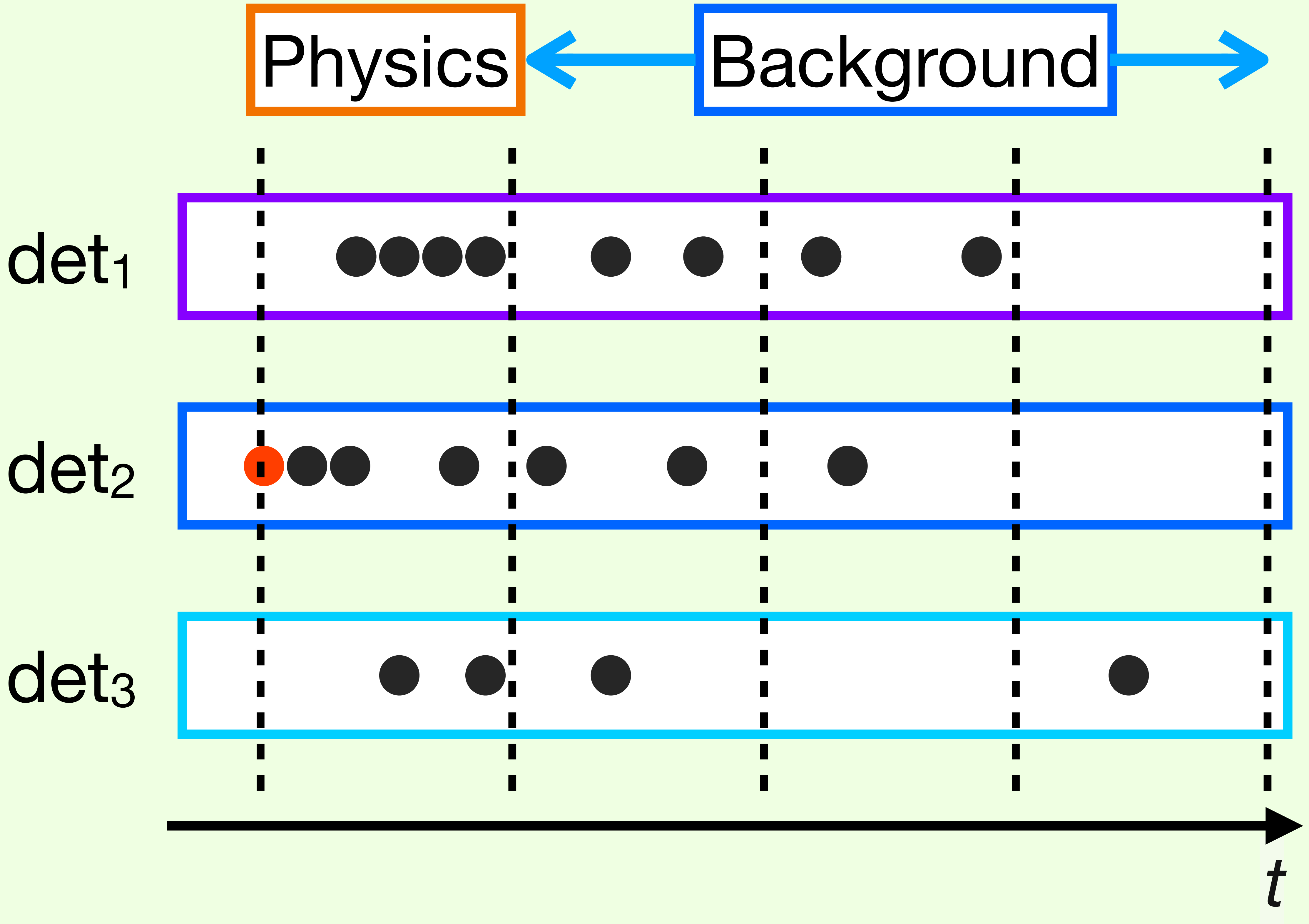
4.3 Trigger Evaluation

For physics simulation, where we know the event properties (time, # of hits etc), we can define physics region and noise region of time-frames

1. Search for the fastest hit.

2. Determine the first 20 ns as physics region. (It is possible because this simulation have only physics events)

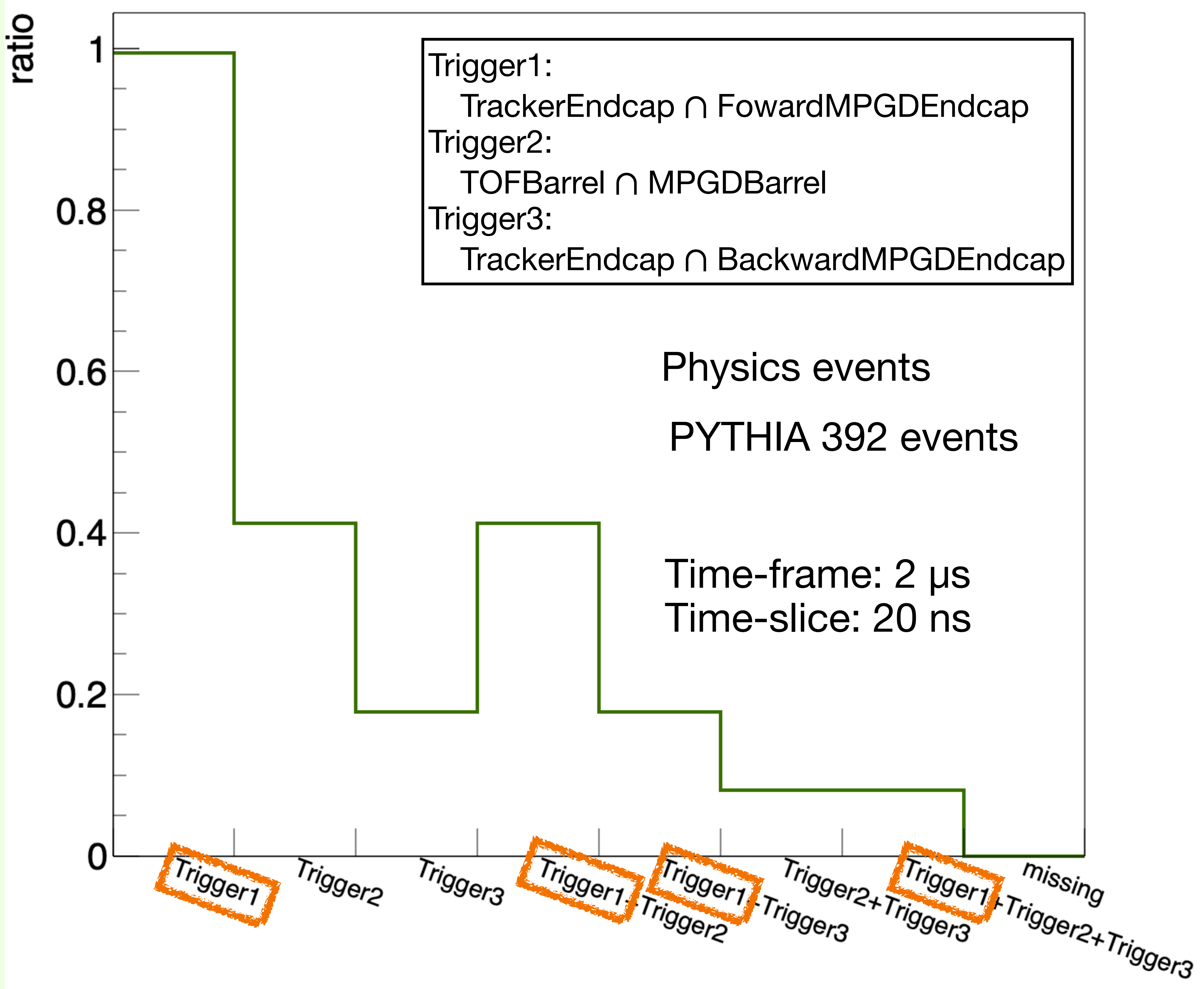
3. Check the what kinds of sub-detectors have hits and make triggers.

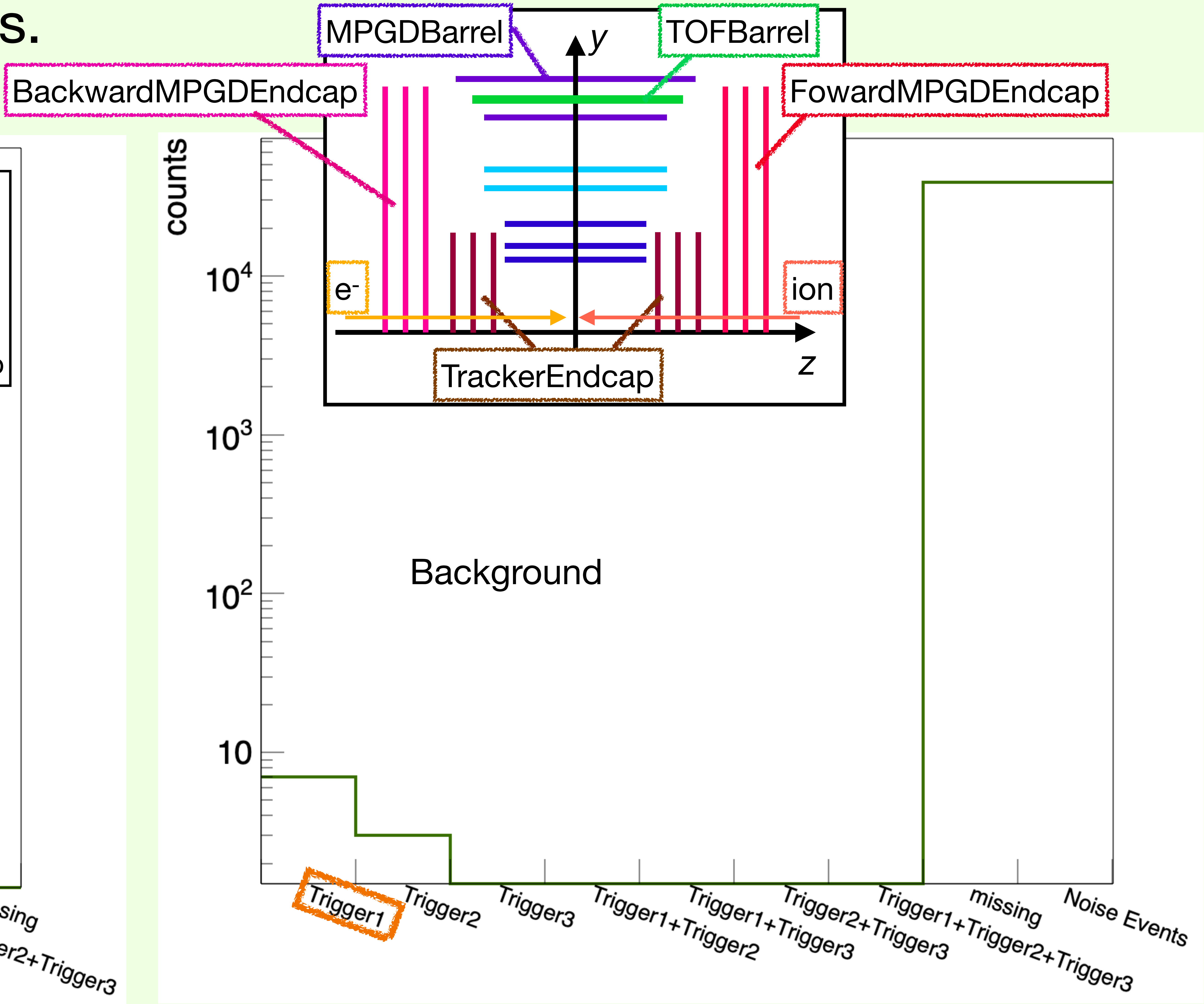


※ Background: Slow hits came created by physics event simulation

4.5 Results of Trigger Contributions

The triggers could cover all physics events.
However, many noise region also remain.





The triggers can keep the full physics events and reduce unnecessary time slices (0.02%)

Next Work

1. Implement the trigger in the EICrecon.

2. Estimate trigger efficiency

3. Test the unfolding using a simulation with only noise

4. Test the unfolding using the physics simulation embedded noise.

Progress and Plan

1. Create a new factory to time alignment (**Done**)

2. Unfolding trivial test (**Done**)

3. Time splitting test (**On going: End of Jun**)

4. Timing Coincidence (**Mid of July**) ← Collaboration meeting?

5. Injection of background events and detector noises and evaluation of timing coincidence (time windows vs. rejection, efficiency, primary tracking, etc) (Mid of August)

6. Apply detector response (September?)

7. Optimize time window, Time alignment, Interval selection (October?)