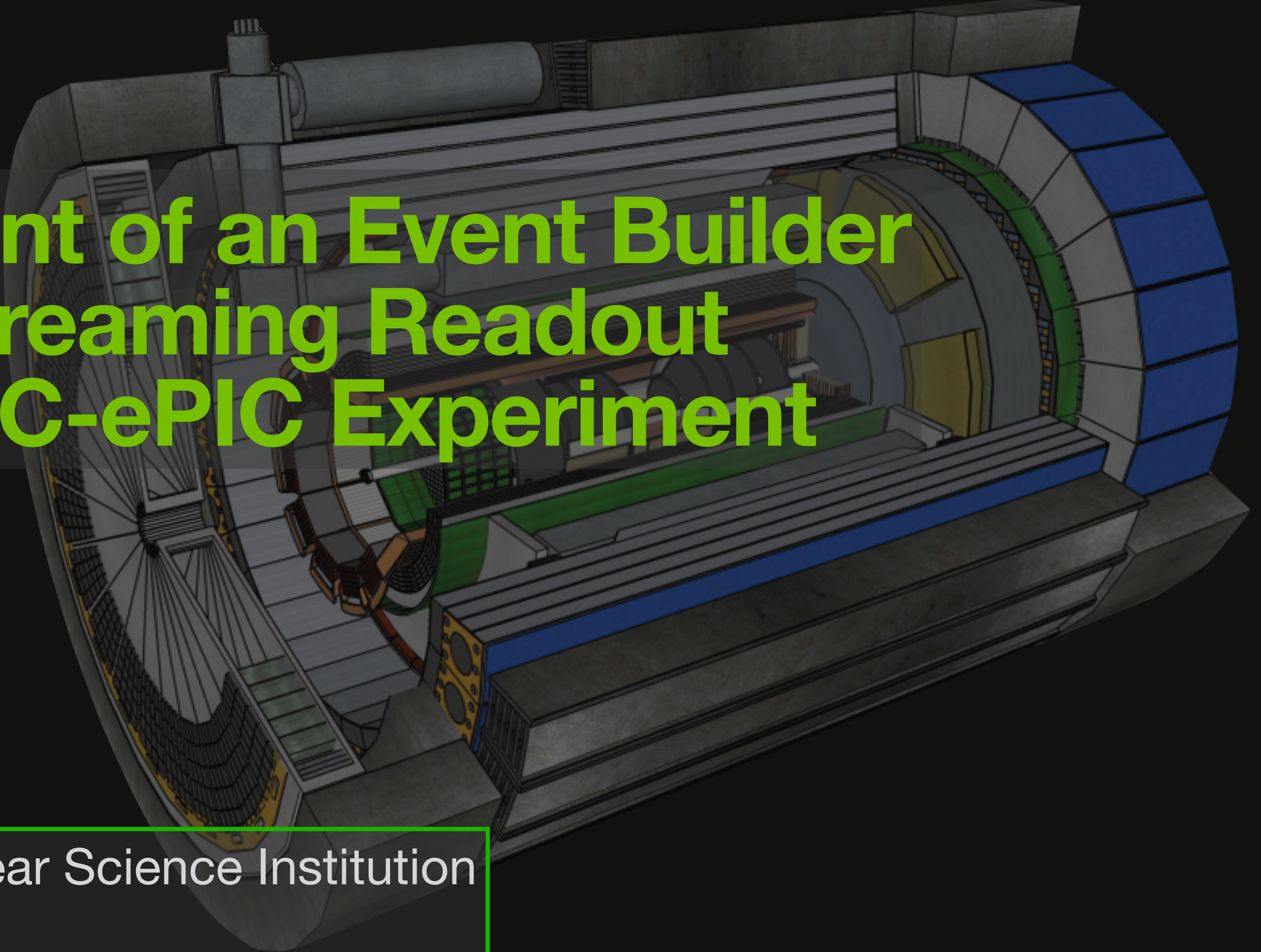
Development of an Event Builder with Streaming Readout for the EIC-ePIC Experiment



The University of Tokyo, Quark Nuclear Science Institution
Takuya Kumaoka

Purpose of this study

Acquire unbiased physics events at 500 kHz (@ 18x275 GeV^2)

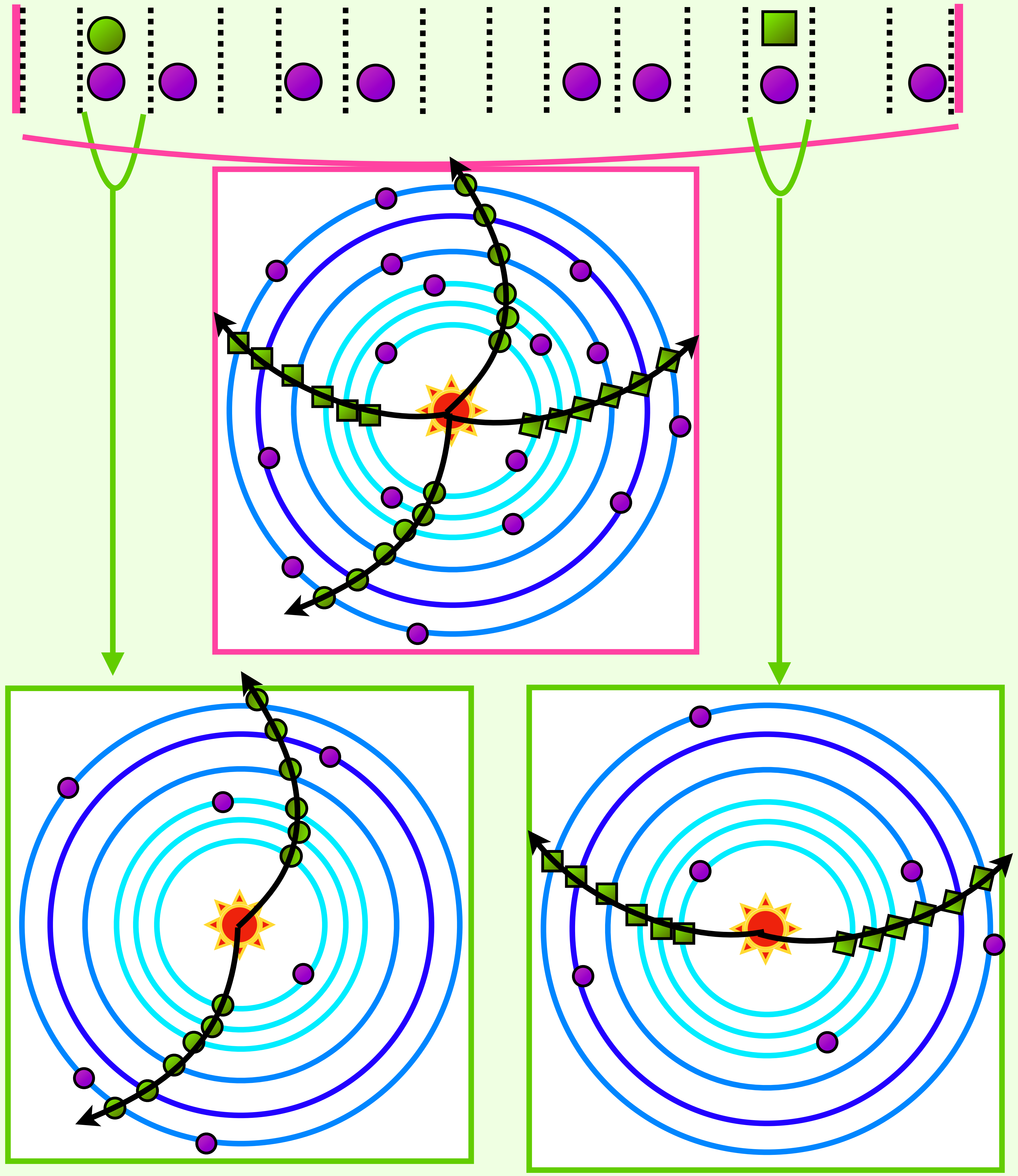
Expected background rate (dominated by Synchrotron radiation) : 3.3 GHz(@ 18x275 GeV^2)

→ **Essential to deploy streaming reconstruction to select physics event and reject background**

(10 Tbps → 100 Gbps)

Purpose

Development of streaming reconstruction and algorithms for event selection



Online Event Filtering System

Progress

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

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

3. Time splitting test (**Done**)

4. Timing Coincidence (**Done**)

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

This Presentation Topic

1. Introduction new time splitting algorithm (ver.3)

2. Event filter efficiency

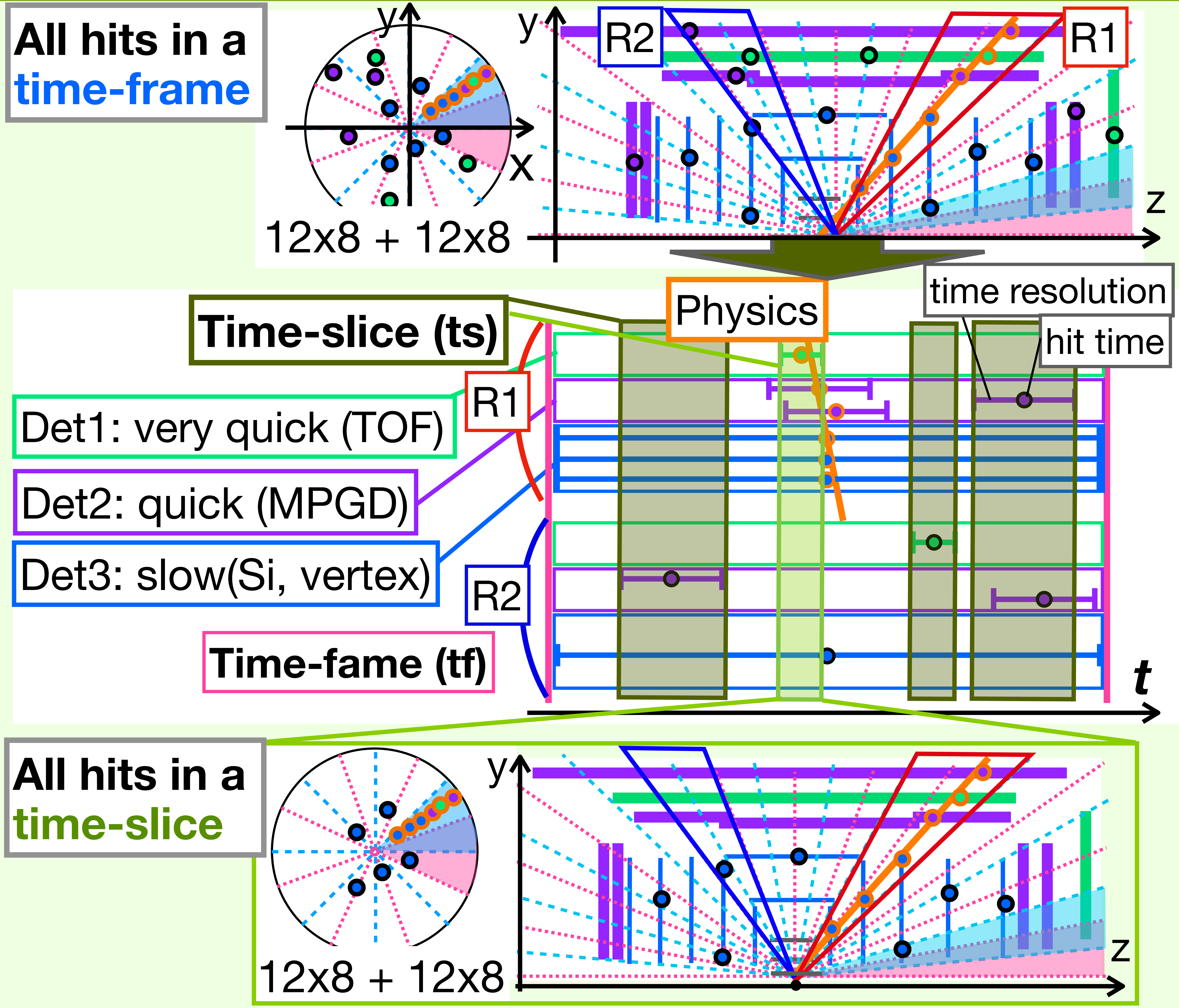
3. Process time

4. Track and Vertex Parameters

5. Track's Purity and Efficiency

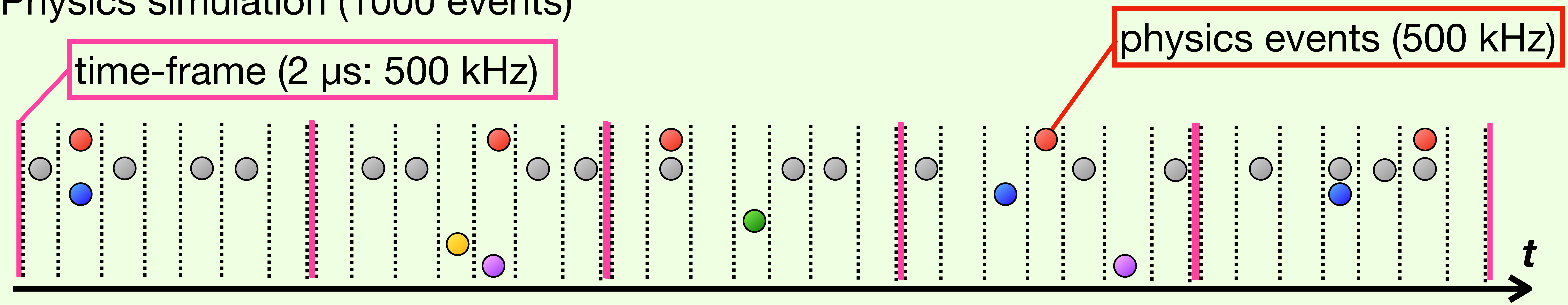
Pre-filter Algorithms

1. Using the fastest detector (TOF) slices the time-frame.
2. Check the topology of the hits.
3. If there are more than any three hits in the same region, the time slice is accepted as event candidate.
4. Repeat steps 1–3 for all TOF hits.
5. If any unassociated MPGD hits remain, perform the same process using MPGD hits.
6. if there are no hits in TOF and MPGD in certain time slice, perform the same process based on silicon hits.



Simulation Setup

Physics simulation (1000 events)

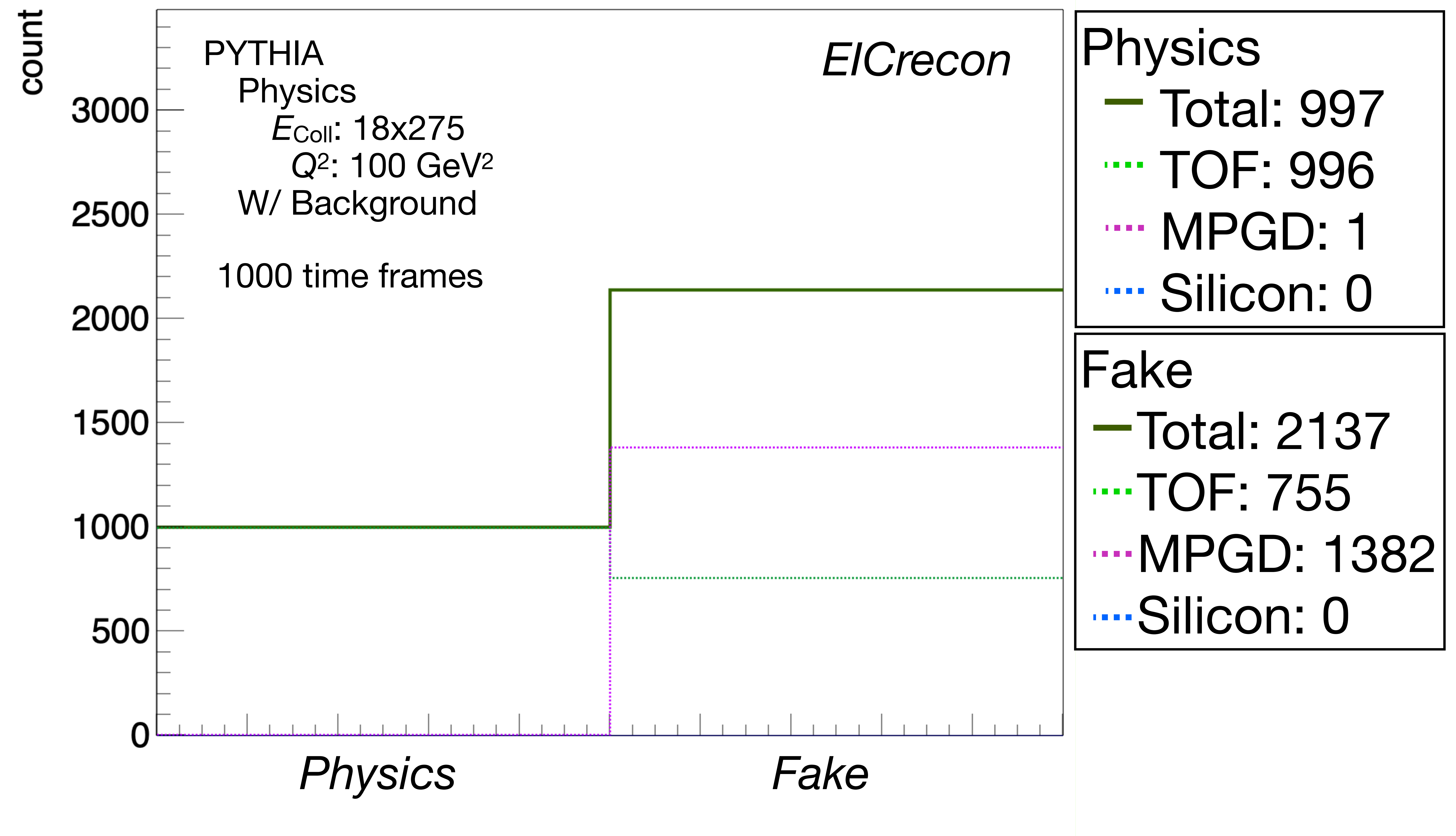


🔴	**DIS NC 18x275 Q^2 > 100 GeV^2 (Deep inelastic scattering neutral current)**	500 kHz
⚪	**Synchrontron Radiation**	3.3 GHz
🔵	**Electron bremsstrahlung radiation**	317 kHz
🟢	**Electron Touscheck scattering (intrabeam dcattering)**	1.3 kHz
🟡	**Electron Coulomb scattering processes**	0.72 kHz
🟣	**Proton beam gas interactions**	22.5 kHz

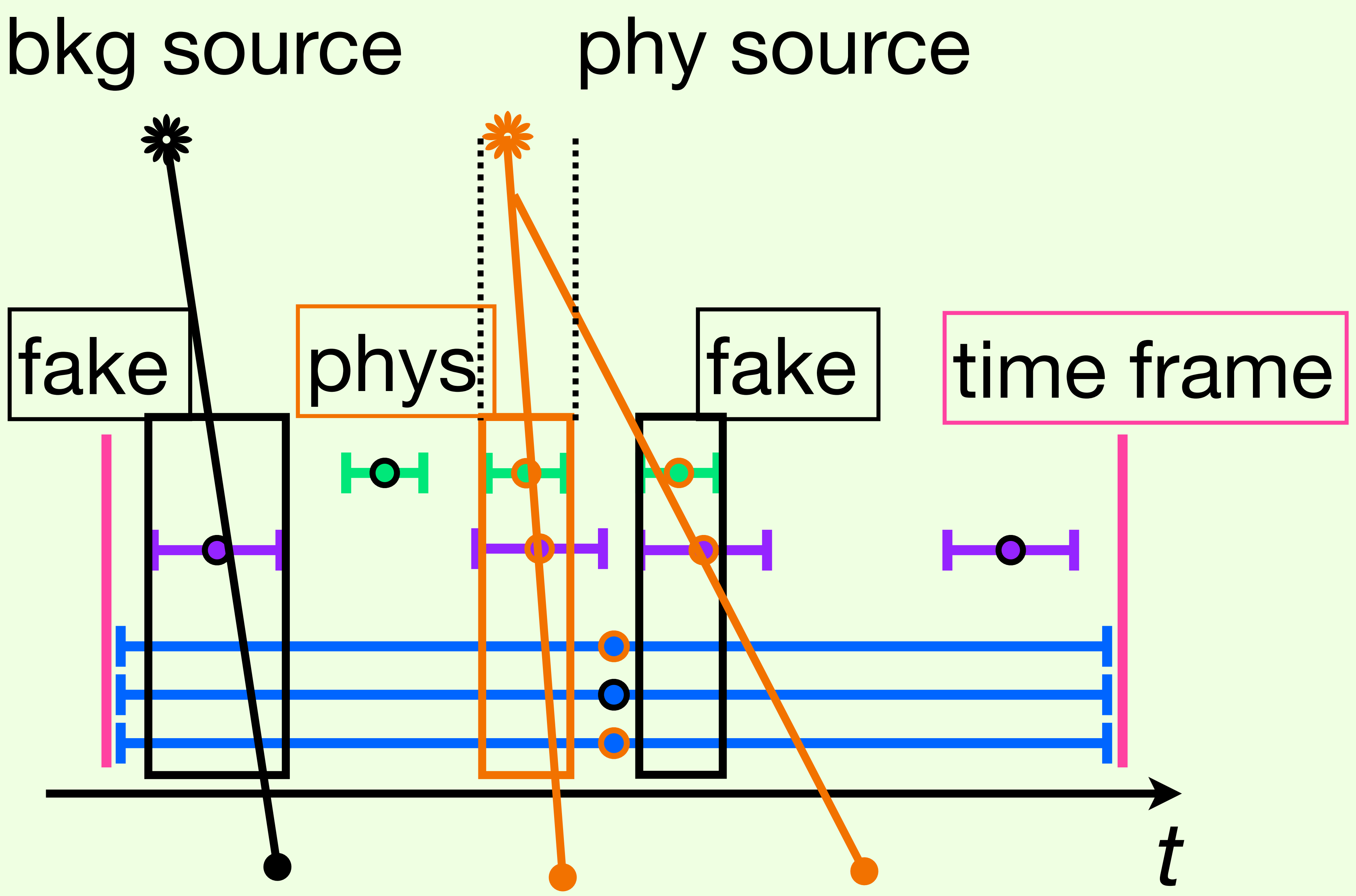
※ Very specific case to easy trigger (high Q^2).

In the real case, the high Q^2 events are lower than 500 kHz.

Pre-filter Efficiency

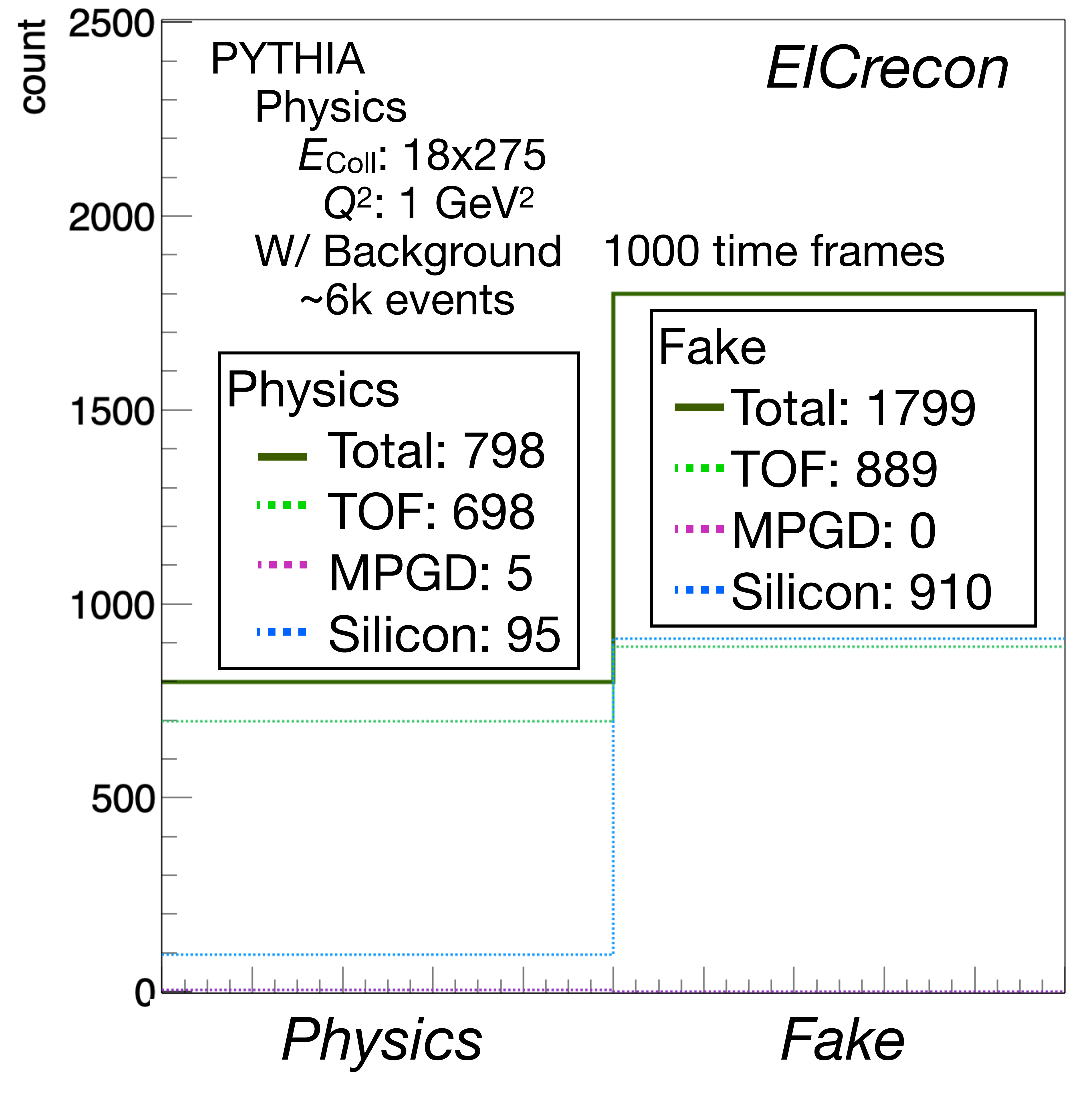


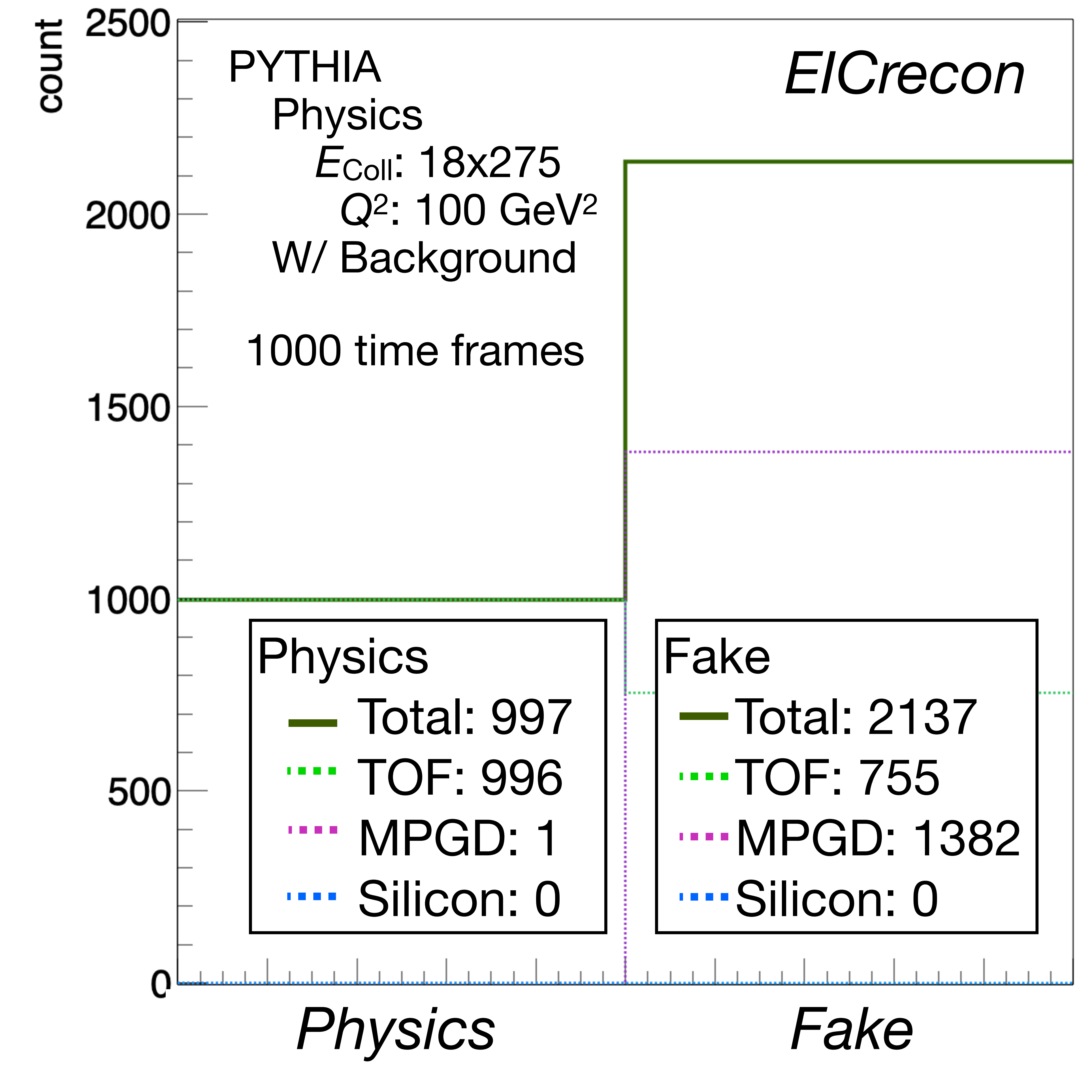
Time slice containing the production time of a physics
→ ✔ **physics trigger**
→ ❌ **fake triggers**



For Q^2 = 100 GeV^2 events, the pre-filter efficiency is over **99%**.
→ Need to check what makes these fake triggers.

First Event Filter Efficiency





For Q^2 = 1 GeV^2 events, the first event filter efficiency is about 80%.

→ This quality is not enough. → What type of events are missed? → Lower than Q^2 = 1 GeV^2?

Total Process Time W/O and W/ TimeSplitter

1000 time-frame

W/O TimeSplitter: **1.7 h** ➡ W TimeSplitter: **1.5 h**

The processing time with the time splitter is **faster** than without it (~10%).

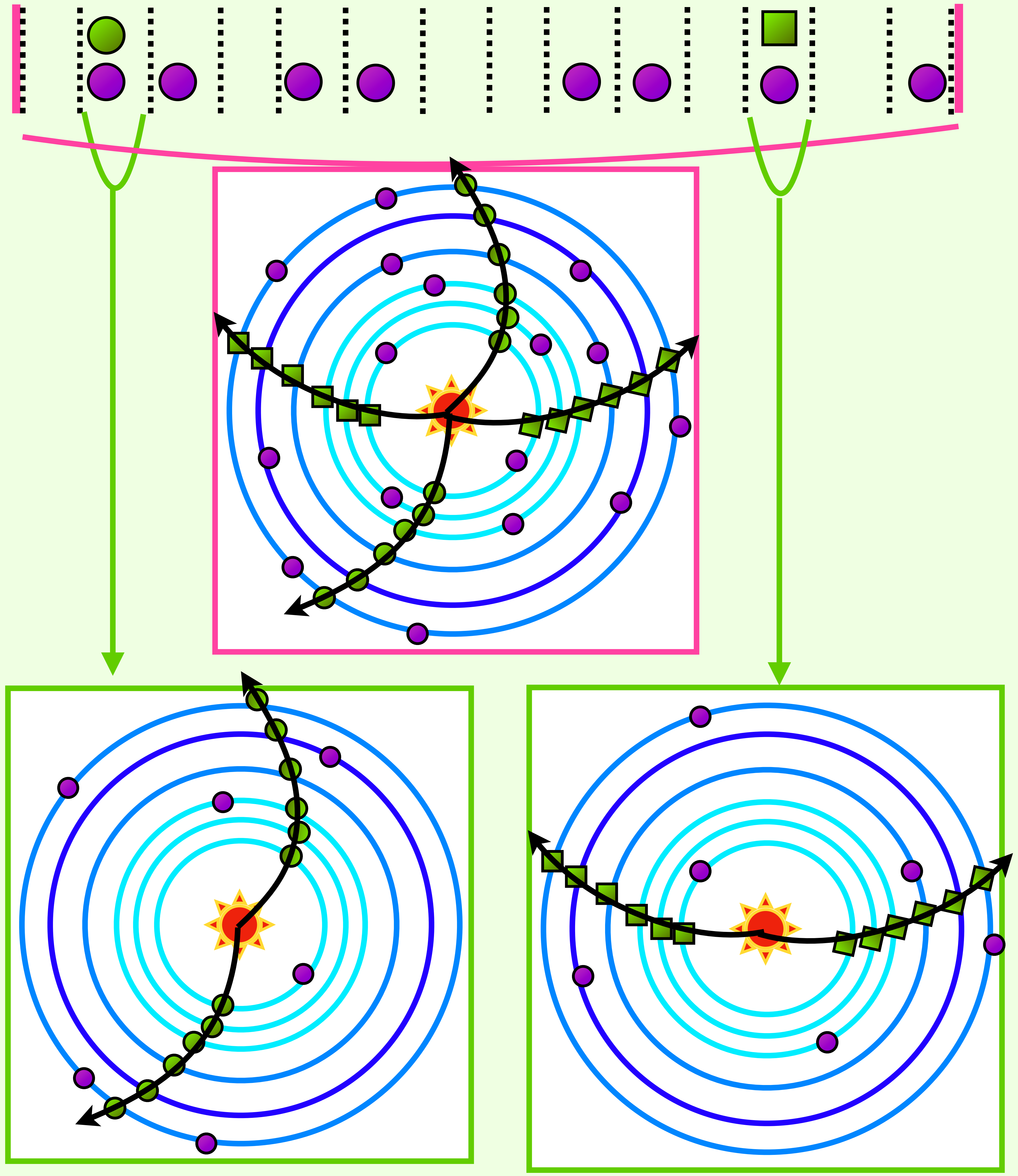
→ This is expected because the splitter reduces background hits and helps the tracking.

Single event with large number of hits

W/O TS: **468 s** ➡ W/ TS: **242 s**

→Particularly, this splitting works well for large number of hits.

It is expected it become more faster using multi-threads or GPU.



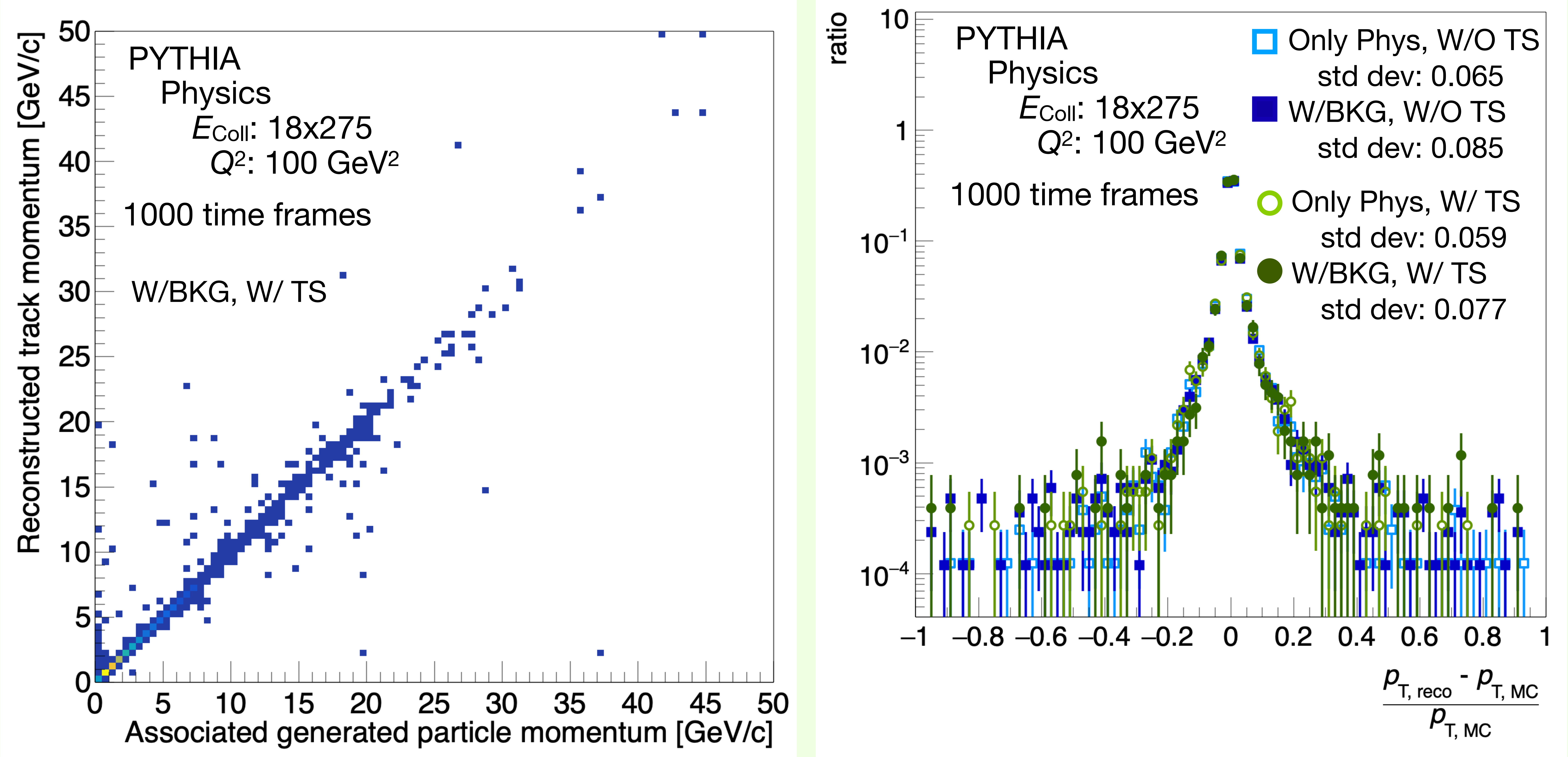
Reconstructed Tracks' and Vertices' Parameter Comparison

ACTS's tracking and vertex output comparison

Overall, the results with time splitting are consistent with those without it.

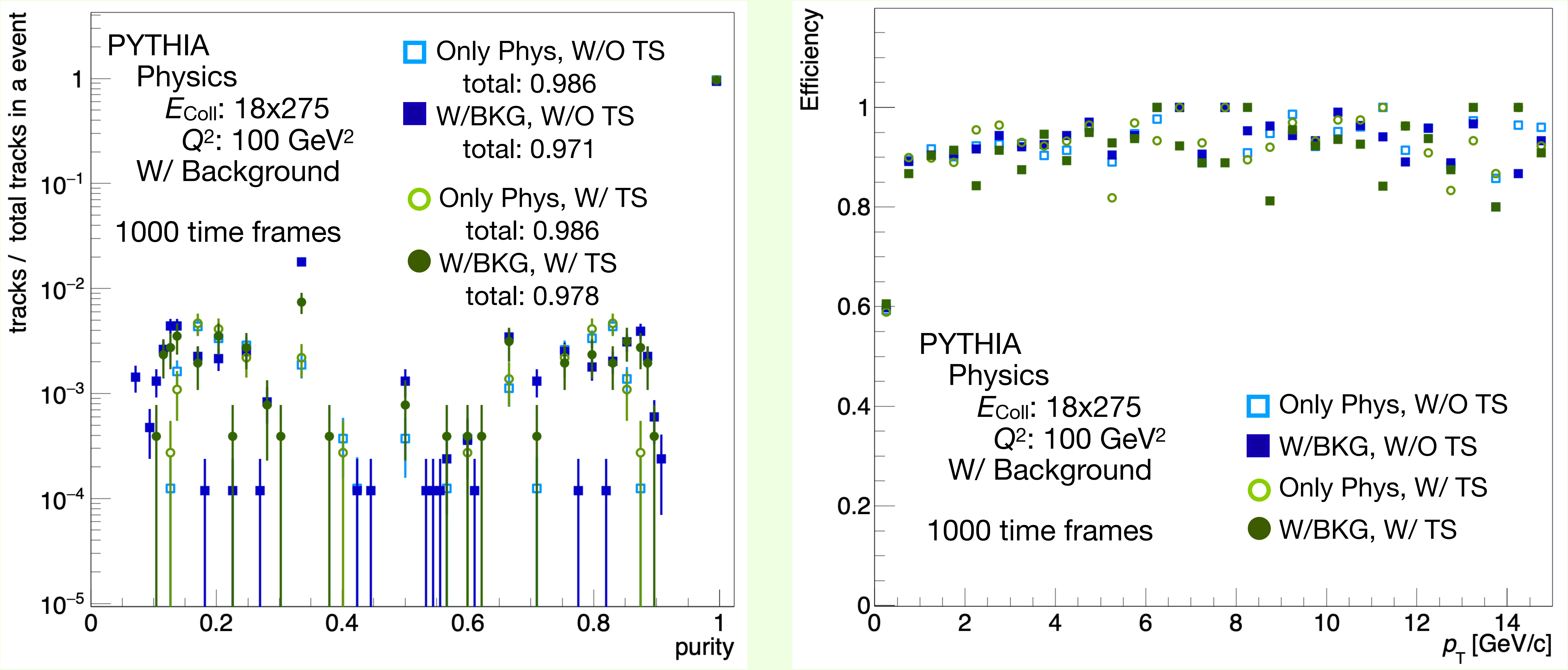
For both the tracking χ^2 and the vertex χ^2, the time-split results show **better** performance than the no-splitting case.

Track p_T Resolution



Good linearity between MC and reconstructed-track p_T.
The p_T resolution of the time splitting is a little better than W/O time splitting.

Track Purity and Efficiency Evaluation



The purities of W/ time slice were a **little better** than W/O ones.
The efficiency of W/ time slice were mostly consistent with W/O ones, but a **little worse**.

Difference between physics and background time slice

- The most fake time slices have only a small number of hits.
- The tendencies of values are not an obvious difference.
- Some time slices do not have a vertex despite the physics time slice.

Next Plan

- Understand what type of events are missed and when the fake triggers are generated.

- Test wiring system developed by Nathan.

- Test **multi-threads** run. (GPU usage?)

- Implement a new factory for **strict second filtering** and **event identification**.

- Apply this splitting code to various physics (Particularly, **low Q^2** simulation) and realistic event ratio （Simon’s developments should support this study.）

Preparing new system to re-write existing EICrecon codes

When the system changes, we often need to update the code in a consistent way.
e.g. When adding new parameters or renaming global variables.

Nathan Talk
https://indico.bnl.gov/event/30167/contributions/115026/attachments/65786/113020/2025-10-26%20external%20wiring%20update-1.pdf

TOML: https://toml.io/en/

Just preparing and test the system with Nathan.

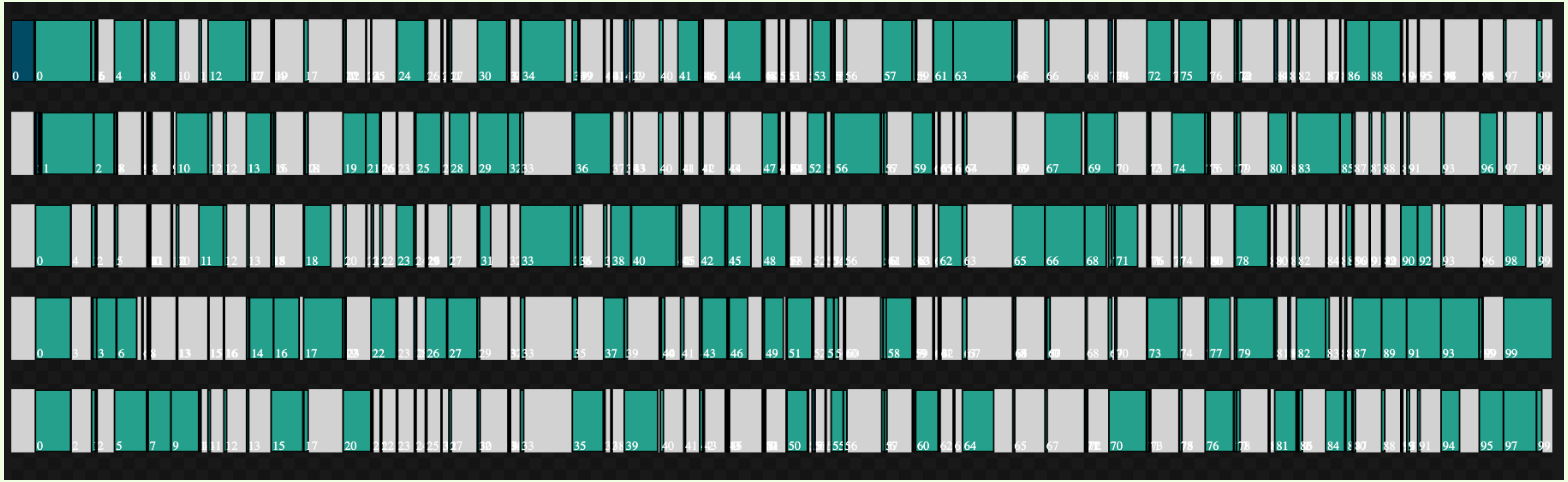
```C++

app->Add(new JOmniFactoryGeneratorT<SiliconTrackerDigi_factory>(
  "TOFBarrelRawHits", {"EventHeader", "TOFBarrelHits"},
  {"TOFBarrelRawHits", "TOFBarrelRawHitAssociations"},
  {
      .threshold      = 6.0 * dd4hep::keV,
      .timeResolution = 0.025, // [ns]
  },
  app));

```

```toml

[[wiring]]
action = "update"
plugin_name = "BTOF"
type_name = "SiliconTrackerDigi_factory"
prefix = "TOFBarrelRawHits"
level = "Timeslice"
input_names = ["EventHeader", "TOFBarrelHits"]
output_names = ["TOFBarrelRawHits", "TOFBarrelRawHitAssociations"]

    [factory.configs]
    threshold      = "6000.0"   # dd4hep::keV,
    timeResolution = "0.025"    # [ns]

```

nthread test

nthread=1: 456.223 s / 100 time frame

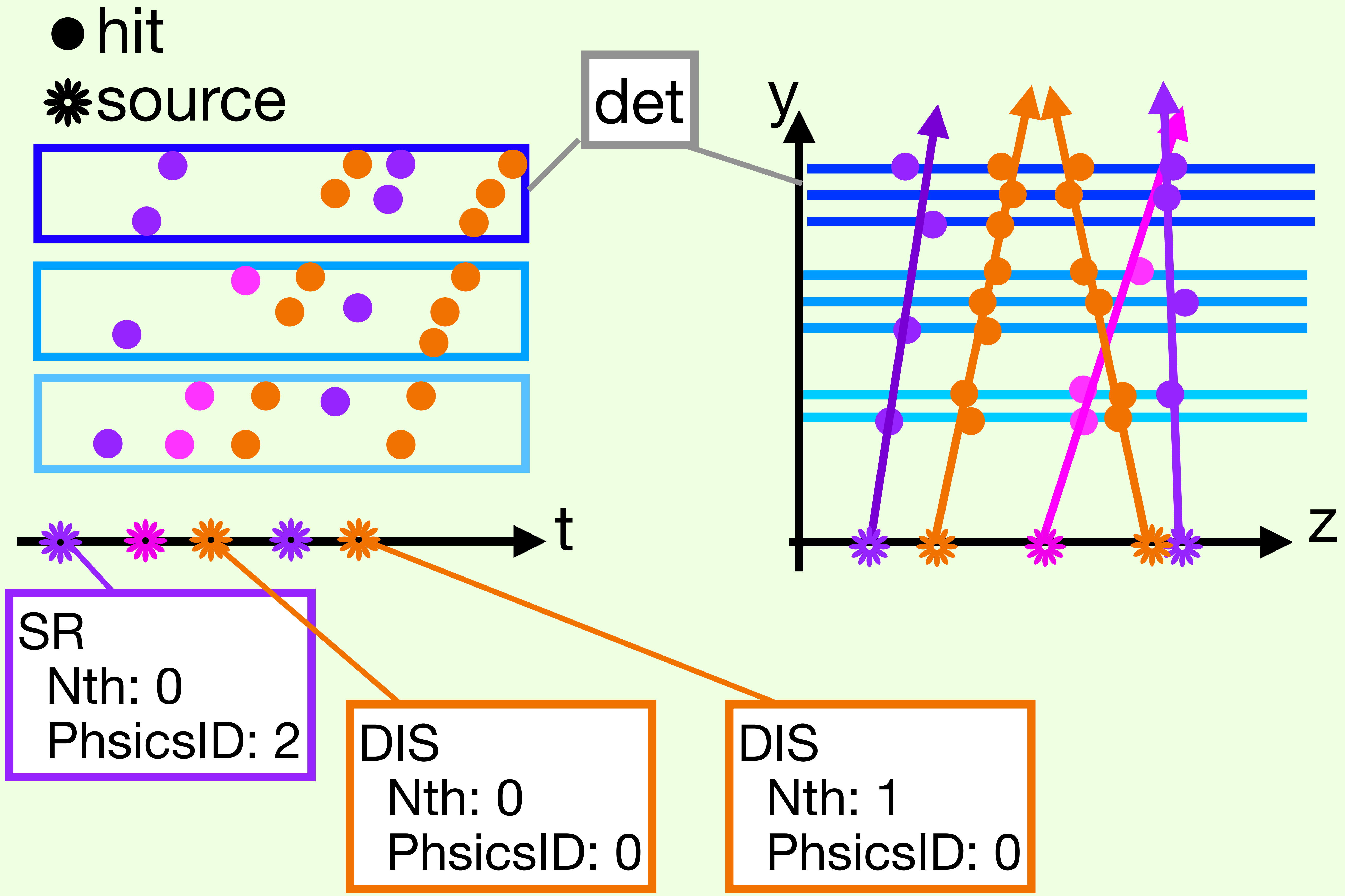
nthread=5: 541.52 s / 100 time frame



Despite increasing the number of threads correctly, the processing time increased.
→ Need to understand the Nathan's evaluator and the reason.

New Source Data Type Proposal

Currently we are missing some event source information (source time, numbering, Q^2…).
We have MC particles information and it has generation status to distinguish physics and background. However, this status cannot distinguish different sources of same type source.
→The proposing data type enable to distinguish sources and give valuable information.
And it is helpful for both MC and reconstruction (a little duplicate reco vertex).



✸ EventInfo ← NEW
- Nth
- CollisionTiming
- CollisionEnergy
- CollisionPosition
- CollisionAngle
- Q2
- PhysicsID

background →

[DIS, EXCLUSIVE, SR, etc…
0 1 2 …]

● DetRecHits
- ⋮
- EventAssociation ← NEW