How to use BIB

simulation data in analysis

General principles and technical implementation

N. Bartosik [a]

on behalf of the

Muon Collider Physics and Detector group

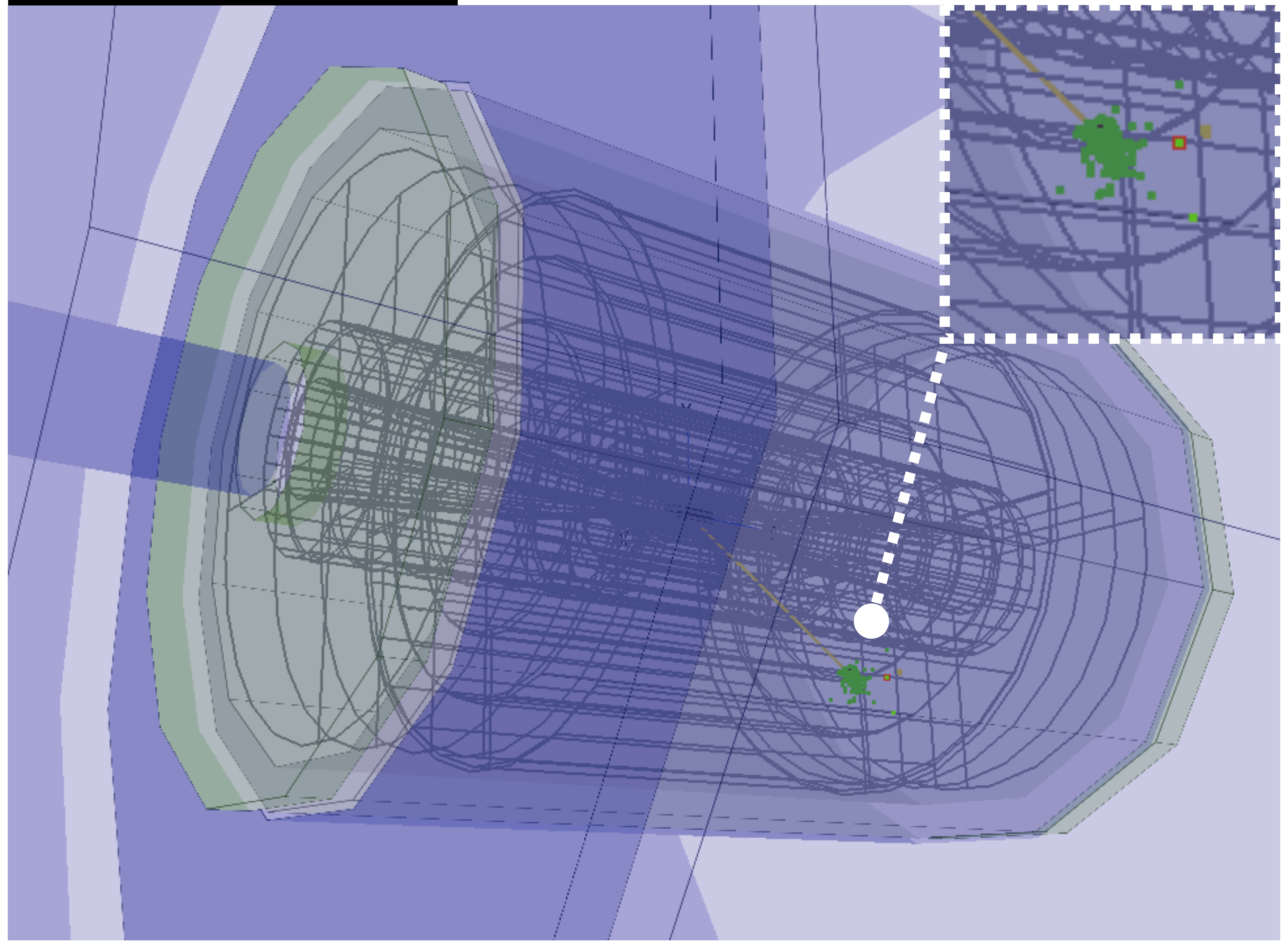
[a] INFN Torino *(Italy)*

Detector design: performance metrics

We want our simulation studies to be representative of what it will look like in the actual experiment

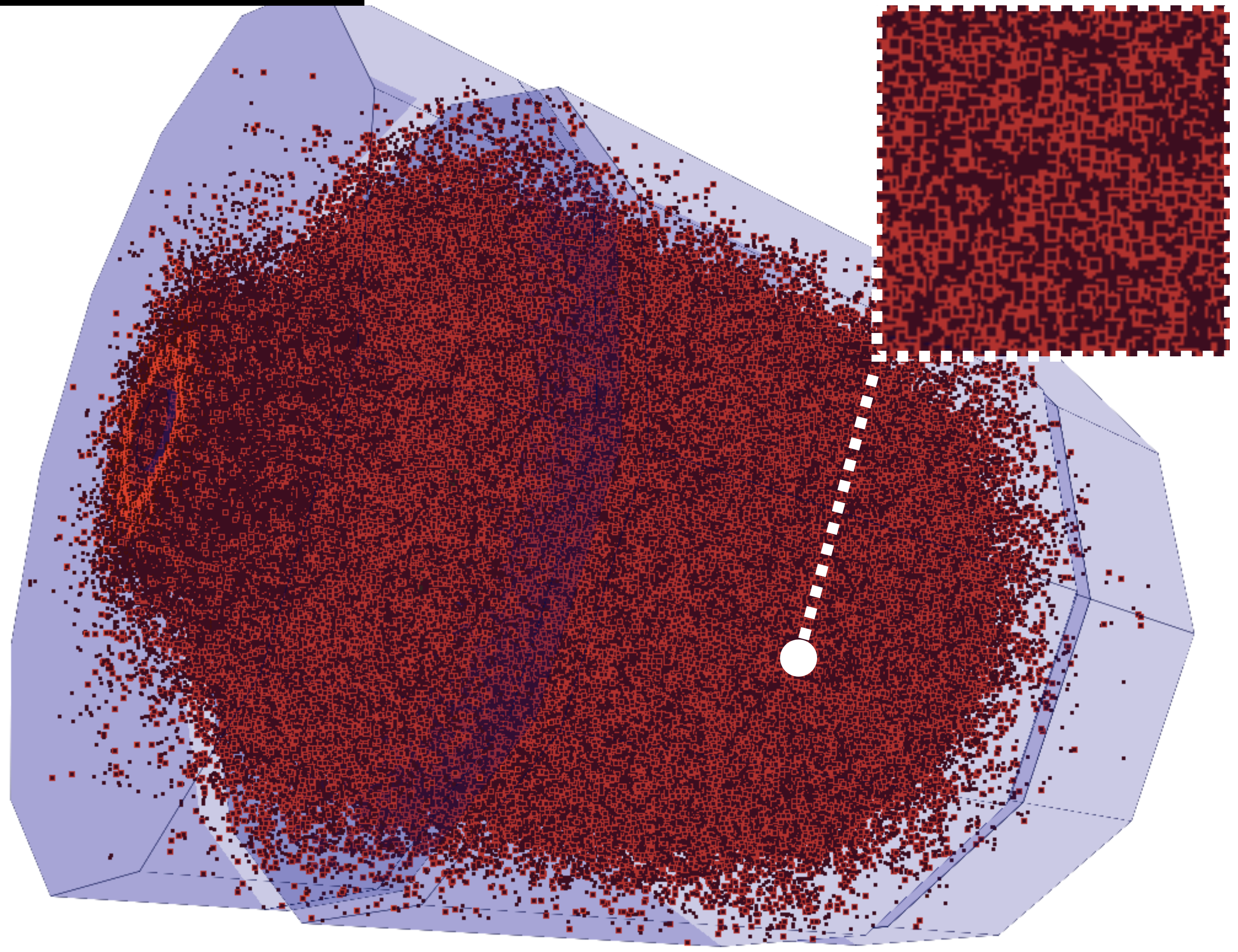
↳ all BIB effects have to be included in the most realistic way

10 GeV photon



➡ physics analyses will be done on this kind of events

10 GeV photon + **BIB** in ECAL



We have a clear separation between the two stages of BIB simulation:

1. **Muons in the accelerator → <u>FLUKA</u> → BIB particles at the MDI surface**
2. **BIB particles → <u>GEANT4</u> → detector signals ready for reconstruction** (within the ILCSoft framework)

Simulation process: main components

Full simulated event obtained via three distinct stages:

GEANT4 simulation of Signal: straightforward and fast

GEANT4 simulation of BIB: ~10^8 particles/event

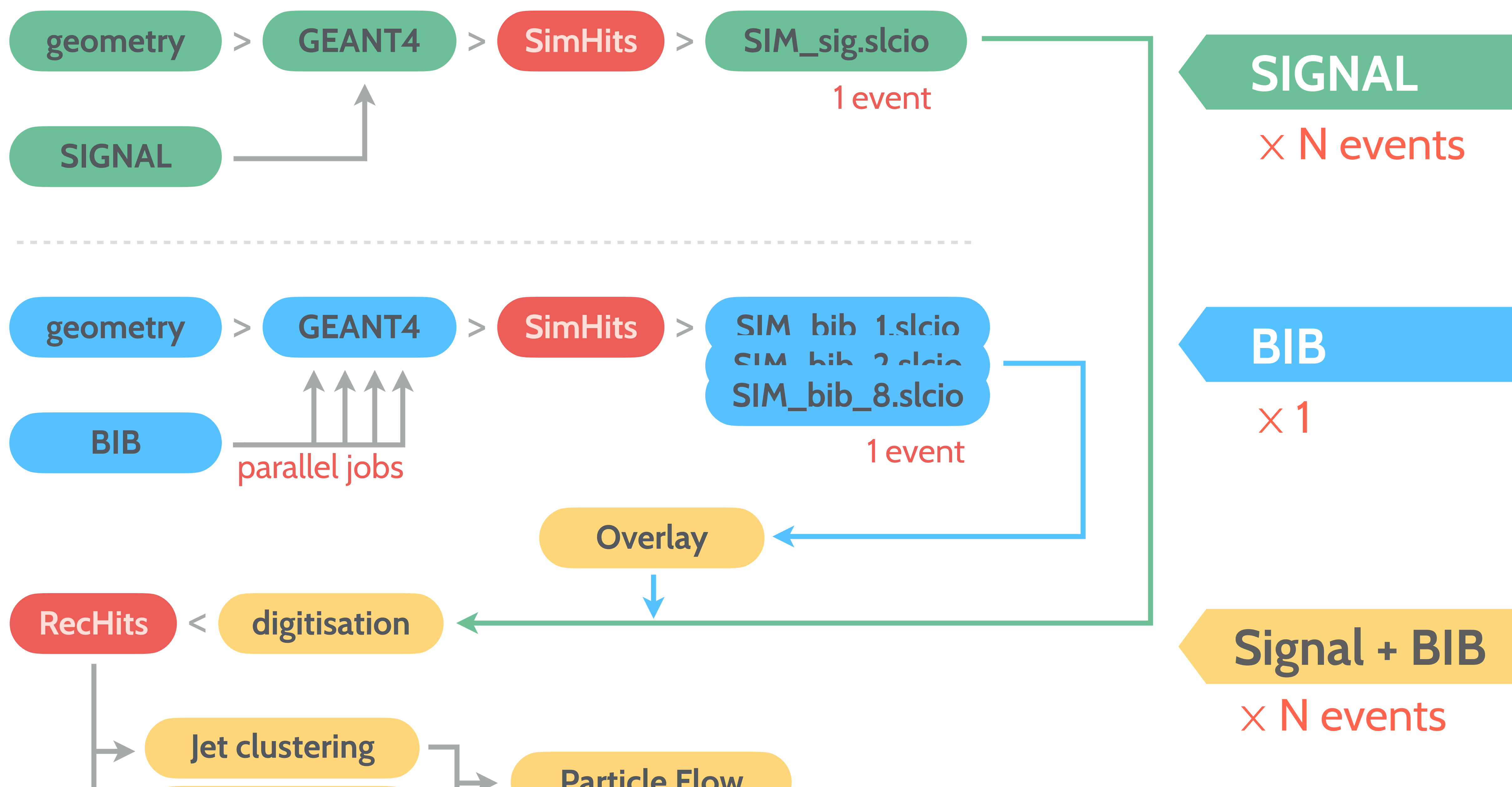
↳ **extremely slow** → need a pool of reusable events

Overlay of BIB: performed in each event before digitisation

↳ sensitive to the # of BIB SimHits and merging logics

Reconstruction speed of higher-level objects strongly depends on the amount of input RecHits from BIB

- especially relevant for track reconstruction *(combinatorics)*
- BIB contribution has to be suppressed as early as possible



Current workflow based on ILCSoft

BIB contribution creates tremendous amount of data → every step requires careful treatment of computing resources

DISK STORAGE | DISK I/O | CPU TIME | RAM USAGE | DISTRIBUTION

Not all of the ~10^8 BIB particles arriving to the detector are relevant for its performance in a real experiment

↳ **detectors have finite readout time windows → only a subset of particles relevant for the event reconstruction**

1. No GEANT4 simulation of particles arriving too late ×6 less CPU

hits at t > 10ns will be outside of the realistic readout time windows

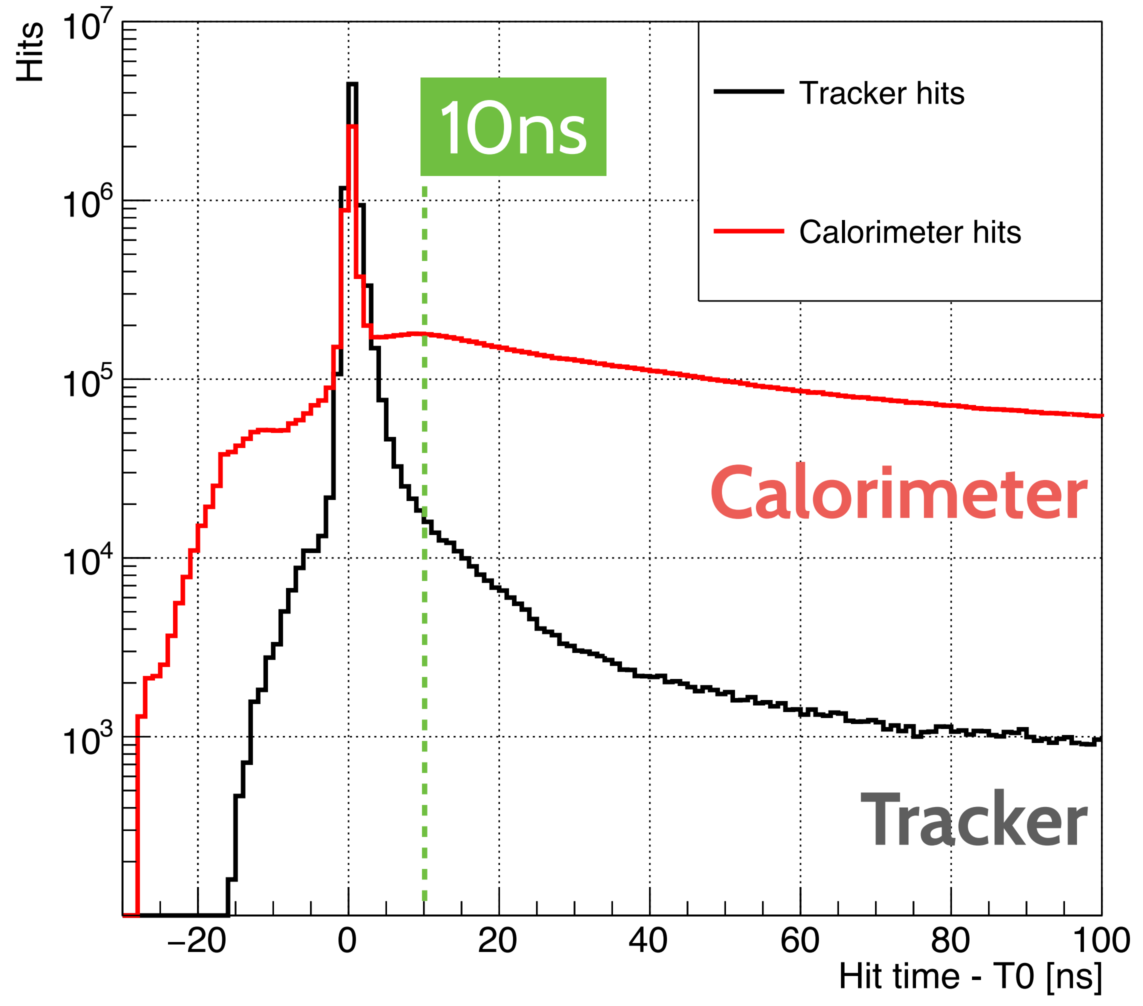
↳ all particles with t > 25ns at the MDI surface are discarded *(accounting for TOF)*

2. No GEANT4 simulation of low-energy neutrons ×20 less CPU

high-precision neutron model required for accurate simulation: `QGSP_BERT_HP`

but they are slow → arrive to the detector with a significant delay

↳ neutrons with E_{kin} < 150 MeV can be safely excluded + faster model: `QGSP_BERT`



1

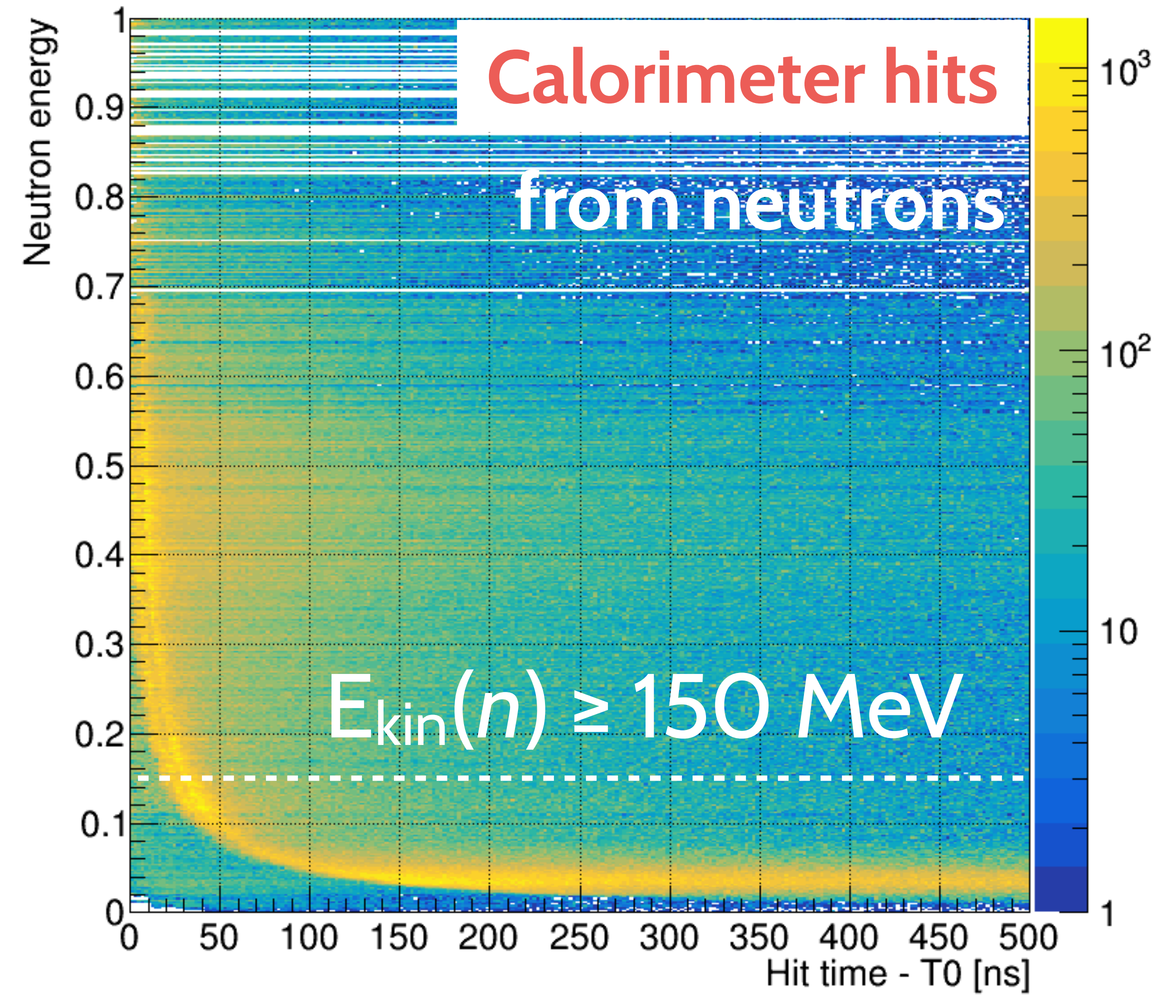
GEANT4 simulation of a single BIB event improved from 127 days → 1 day

↳ ~10-100 reusable events can be generated in several days *(parallelisation)*

CAUTION!

These "irrelevant" particles are excluded based on assumptions about the detector obtained by first simulating all the particles to come up with acceptance definitions

↳ **optimisations must be reevaluated when MDI or detector designs change**



2

BIB organisation: logistics

The result of a single BIB simulation in GEANT4 are 2 files: SimHits from the μ^+ and μ^- beam
representing all the energy deposits in the detector produced by BIB particles from a single BX

Currently we have 1K simulated bunch crossings for $\sqrt{s}$ = 1.5 TeV: entry 001 in the table of MC samples
all obtained from a single sample of BIB particles produced by MAP using MARS15 simulation software

In each of the 1K bunch crossings the polar angle of each BIB particle is randomised before detector simulation
+ sampling of the path of each particle is randomised within GEANT4

↳ provides a reasonable level of statistical independence considering only 1 list of BIB particles available at $\sqrt{s}$ = 1.5 TeV
but not fully correct → some decays are not symmetric in polar angle + particle compositions might fluctuate

Till now our main focus has been on solving computational bottlenecks and optimising reconstruction algorithms
↳ reusing the same BIB sample in every simulated event has been acceptable

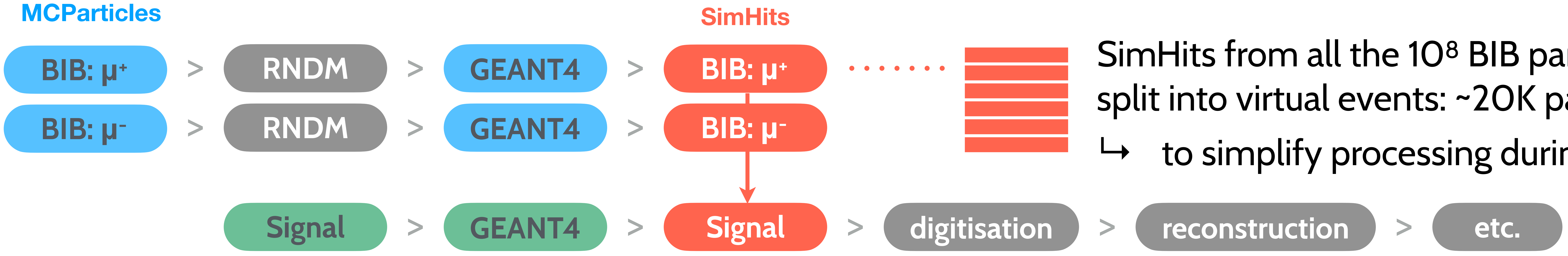
Now a number of physics cases are being studied at Muon Collider
relying on full simulation of thousands of events, including also ML techniques

↳ **statistical independence of BIB contributions between different events is becoming a necessity**

We are now in the process of adopting a more efficient and flexible approach to BIB Overlay
taking advantage of full control over the BIB generation process in FLUKA

BIB organisation: event structuring

With the current approach we package BIB from the whole BX into just 2 files



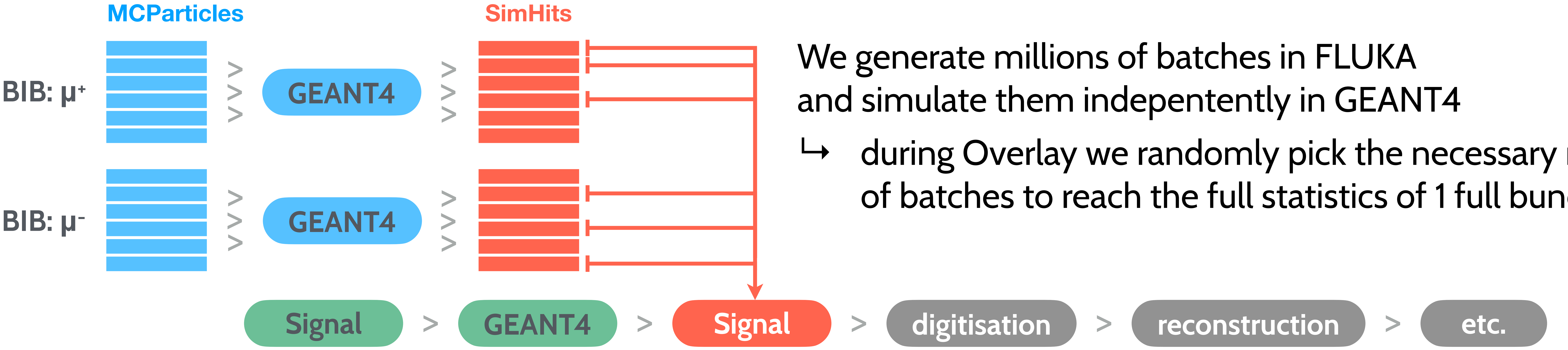
SimHits from all the 10^8 BIB particles are split into virtual events: ~20K particles/event

↳ to simplify processing during Overlay

BIB from the whole BX is treated as a single entity: 1K BIB simulations → 1K independent events

In FLUKA we force every muon to decay and record the resulting particles reaching the MDI surface

↳ muon decays are simulated in batches: e.g. 200 muons/batch → results written to small files (1 batch/file)



We generate millions of batches in FLUKA and simulate them indepentently in GEANT4

↳ during Overlay we randomly pick the necessary number of batches to reach the full statistics of 1 full bunch crossing

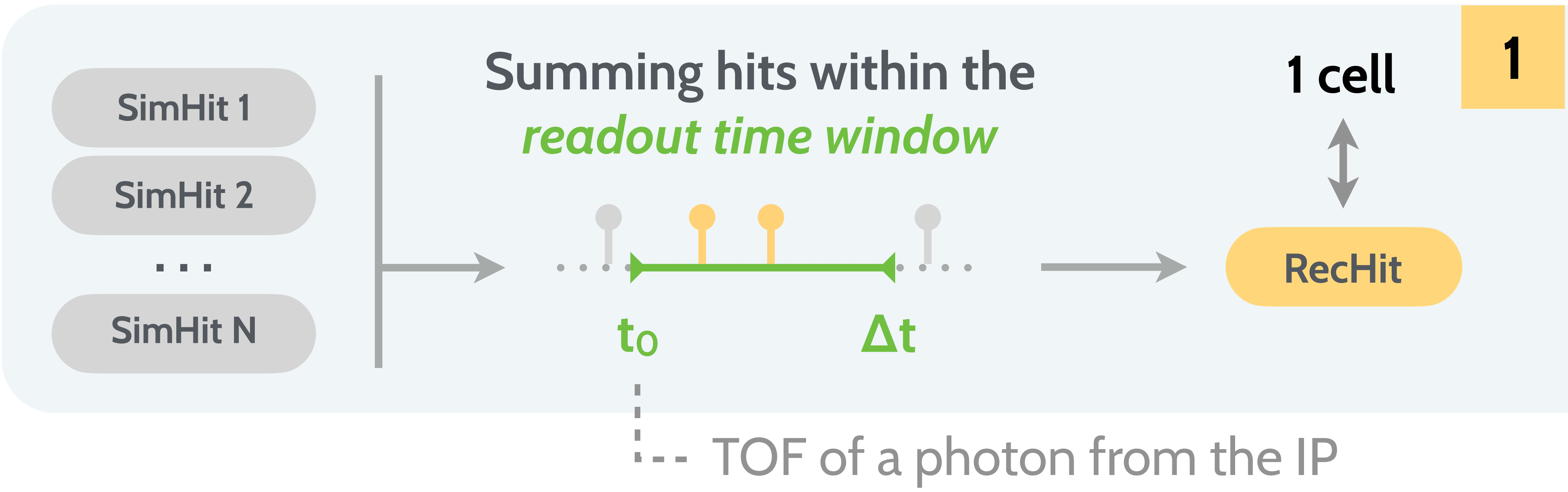
GEANT4 hits produced separately for Signal and BIB → merging + detector effects added during digitisation

↳ **two distinct classes of hits: CalorimeterHit** (ECAL, HCAL, Muon detector) **+ TrackerHit** (Tracking detector)

1. Calorimeter hits: cell ID + E_{dep} + timestamp

large cells (0.5×0.5 - 3×3 cm) → manageable # of cells

↳ hits merged within a fixed readout time window *(0-10ns)*

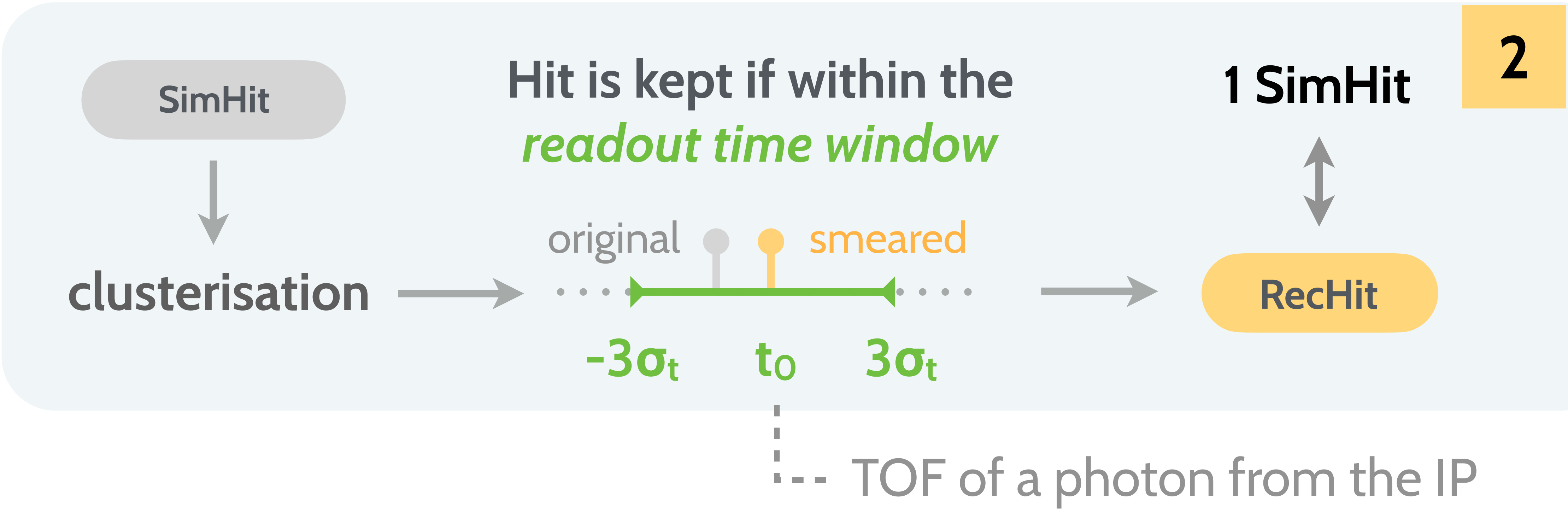


2. TrackerHits: sensor ID + 2D position + time *and more*

from small pixels (25×25 μm) to macro-pixels (0.05×10 mm)

↳ too many channels to treat them individually in GEANT4

Hits kept if within a fixed readout time window



Overlay module has configurable time filters (indepentent for each collection) to only keep the most relevant hits

↳ **saves a lot of processing time + RAM used during digitisation**

Summary

Starting new analysis using full simulation is fairly straightforward:

1. **Install the necessary simulation software:** installation instructions
   or use a precompiled container: list of distributions
2. **Get access to the signal and BIB samples:** list of available samples
   or generate your own signal sample
3. **Set up configuration files for the whole analysis chain:** see tutorial + example configuration
4. **Run the actual chain of modules and study the results**

Some technical details will change in the near future:

- **organisation of BIB samples into batches**
- **migration to a new software framework**
- **extension to global computing and data-storage resources**

Plentry of room for new developements and optimisations
to support the ongoing and future large-statistics physics analyses