Generic ML for fast simulations

by **Michał Mazurek**

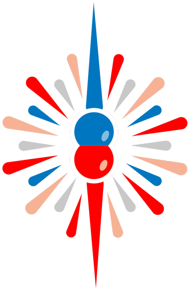
on 4th June, 2024
WMLQ, Warsaw

Finding intangibles

Experimental workflow in HEP

👉 **source of particles**

- collisions in accelerators,
- radioactive isotopes,
- cosmic rays,



Credit: Phoenix Event Display & Picture with colliding atoms

Finding intangibles

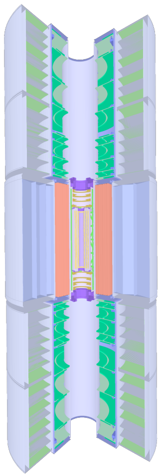
Experimental workflow in HEP

👉 **source of particles**

- collisions in accelerators,
- radioactive isotopes,
- cosmic rays,

👉 **experimental apparatus**

- calorimeter and trackers,
- phototubes in neutrino experiments,

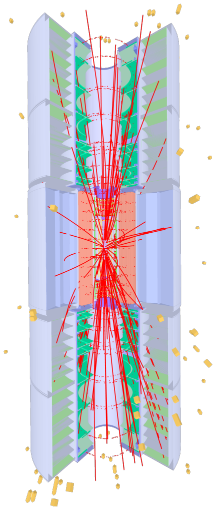


Credit: Phoenix Event Display & Picture with colliding atoms

Finding intangibles

Experimental workflow in HEP

- **source of particles**
  - collisions in accelerators,
  - radioactive isotopes,
  - cosmic rays,
- **experimental apparatus**
  - calorimeter and trackers,
  - phototubes in neutrino experiments,
- **readout system**
  - data acquisition (DAQ),
  - event selection (trigger),
  - event reconstruction,

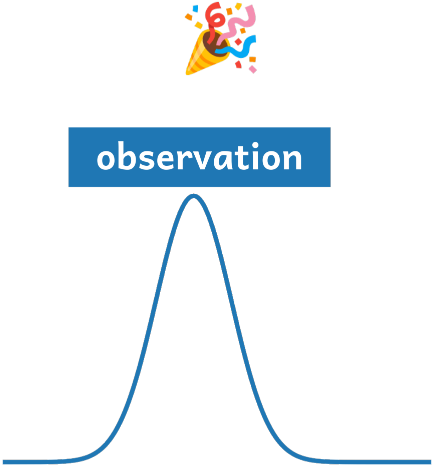


Credit: Phoenix Event Display & Picture with colliding atoms

Finding intangibles

Experimental workflow in HEP

- **source of particles**
  - collisions in accelerators,
  - radioactive isotopes,
  - cosmic rays,
- **experimental apparatus**
  - calorimeter and trackers,
  - phototubes in neutrino experiments,
- **readout system**
  - data acquisition (DAQ),
  - event selection (trigger),
  - event reconstruction,
- **offline data analysis**

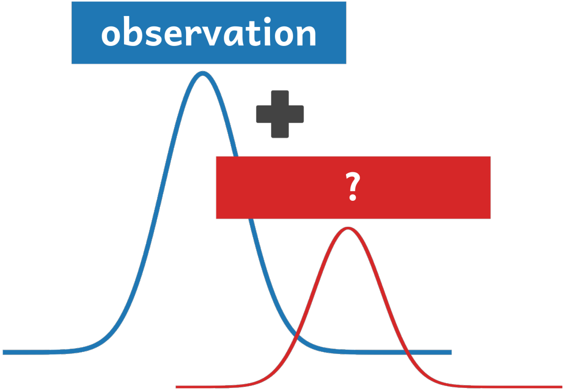


Credit: Phoenix Event Display & Picture with colliding atoms

🎲 Understanding observations

Examples

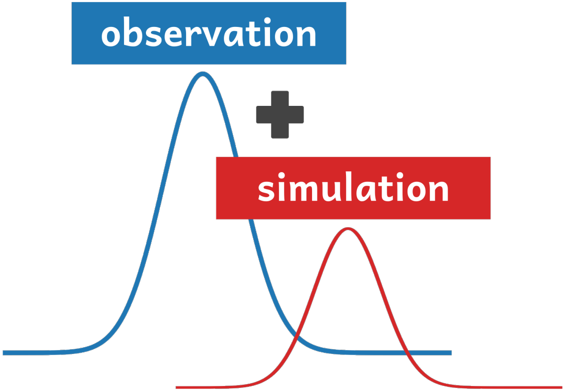
- ❓ new particles and interactions,
- ❓ experimental variables vs. theoretical parameters,
- ❓ estimation of background, efficiencies, etc.



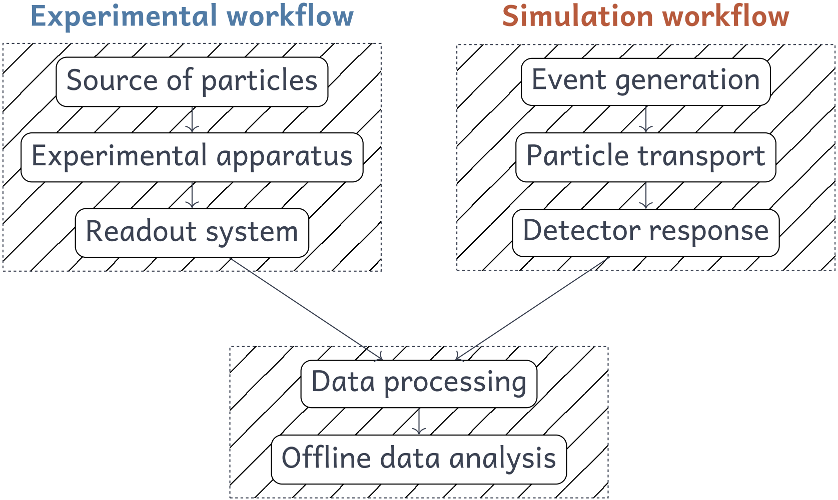
🎲 Monte Carlo simulations to the rescue

Simulation workflow in HEP

- 👉 **event generation**
  - Pythia8, Sherpa, Herwig...
  - particle guns,
- 👉 **decay of unstable particles** (optional)
  - EvtGen,
- 👉 **particle transport**
  - Geant4 (general toolkit),
  - FLUKA (beamline and radiation env.),
- 👉 **detector response**
  - usually detector specific,

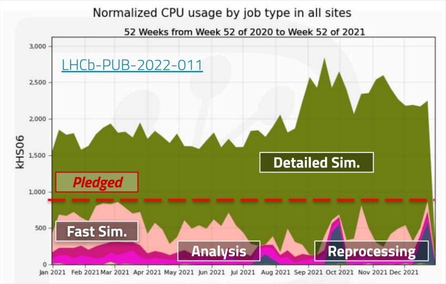


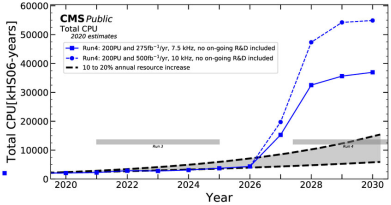
Putting it all together

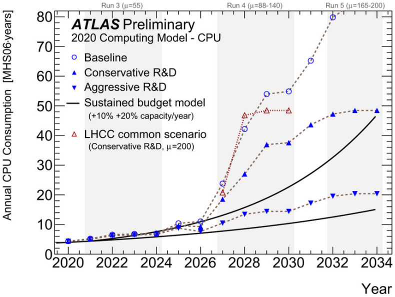


Simulations become very challenging for computing

- large increase in luminosity very challenging for computing,
- LHCb: simulation for Run 2, takes up to 90% of the experiment computing resources,

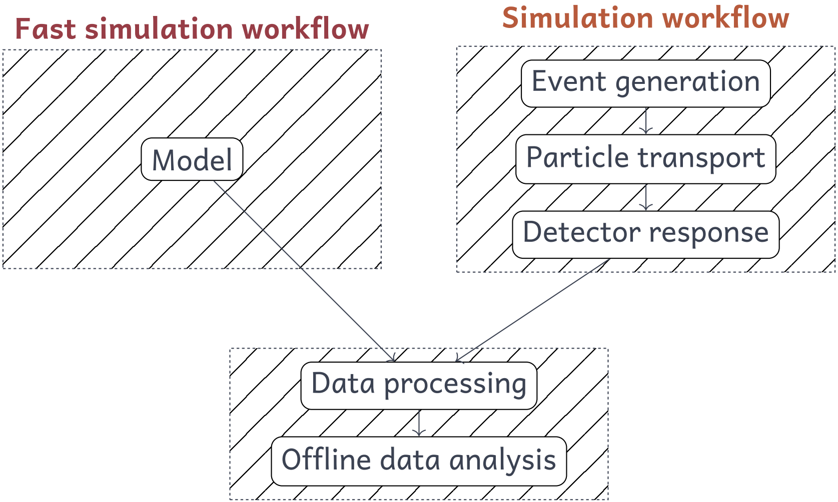






Credit: LHCb-PUB-2022-011, CMS plots & ATLAS plots

Can we make it faster?



🎲 Can we make it generic?

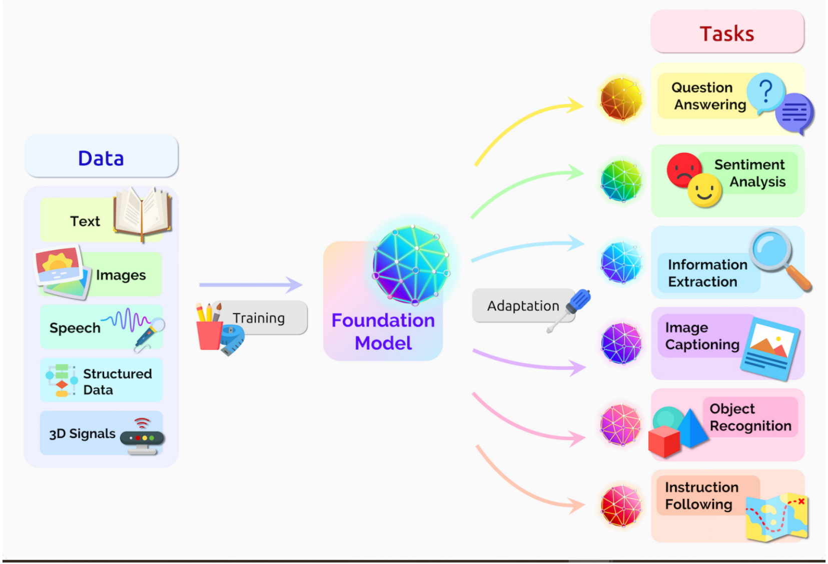
Demanding process…

👉 relatively **high throughput** and **precision required**,
👉 long physics validation required,
👉 large models require training on **large datasets**,

…but feasible! Recipe:

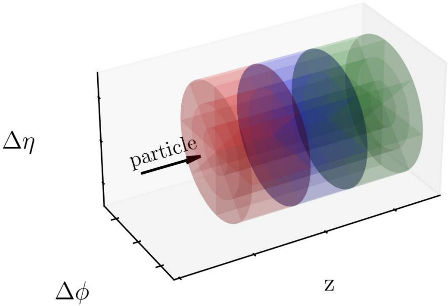
👉 **generic models** trained on **experiment-agnostic** datasets,
👉 interface to machine learning libraries to **run inference**,
👉 **simulation framework** to **adapt** to the target geometry.

🎲 Generic ML models

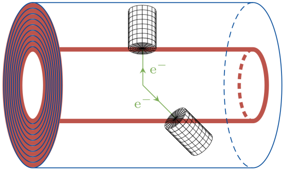


Credit: arXiv:2108.07258v3

Generic ML models: example **CaloChallenge**



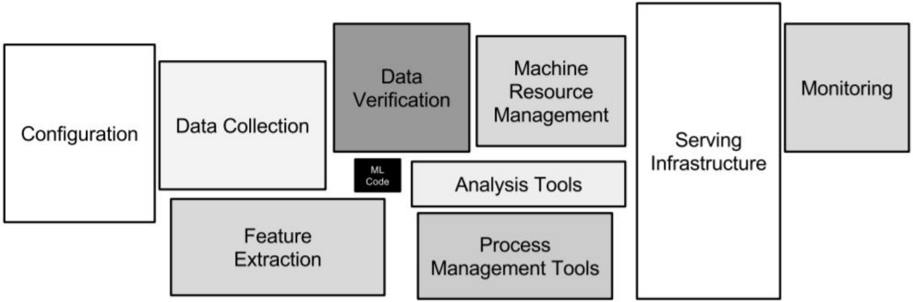
- train on **experiment-agnostic training dataset**
- **compare** various **models objectively**,
- **retrain** the chosen model on the **target geometry**!



Credit: A. Zaborowska, D. Salamani, et al. & W. Pokorski (Geant4 Techn

Generic ML model serving

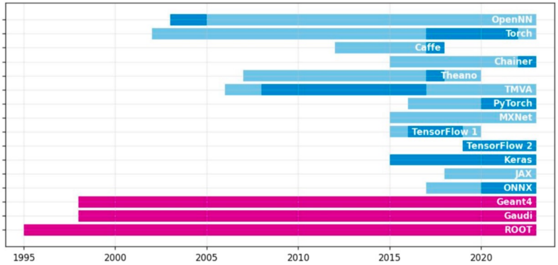
How do we deploy, maintain and serve ML models?



Credit: image from Medium

🎲 Generic ML model serving

- 👉 many frameworks exist now: pyTorch, TensorFlow, ONNX...
- 👉 long term support needed,
- 👉 must run efficiently in C++,



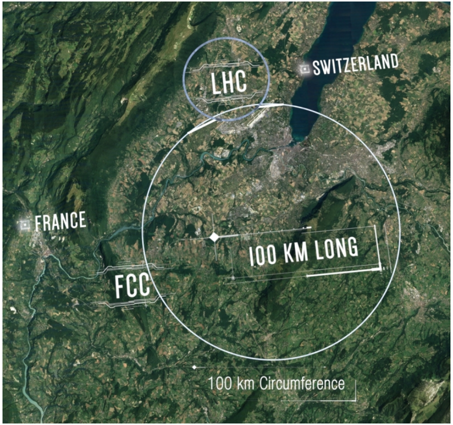
Credit: G. Corti, 6th Upgrade II Workshop, Barcelona

🎲 Choosing a simulation framework

Experiments and their frameworks

- 👉 Athena in ATLAS,
- 👉 CMSSW in CMS,
- 👉 Gauss in LHCb,
- 👉 VMS in ALICE,
- ❓ future experiments (FCC, etc.)

💡 **Can we extract core simulation components?**



Credit: FCC picture

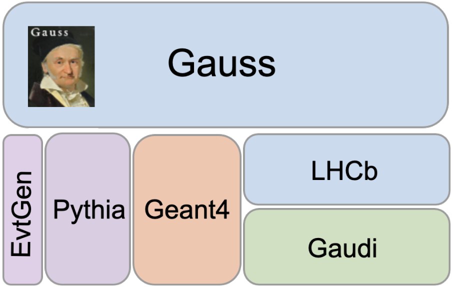
🎲 Core simulation framework for large scale detectors

"We should join forces for an experiment-independent Gauss-core"

— *CERN SFT, 2015*

Features as in Gauss

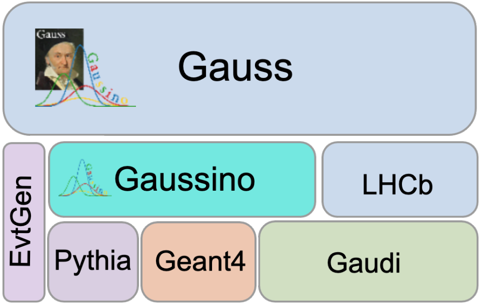
- 👉 **modular event generation** phase,
- 👉 based on common **core software** framework: **Gaudi**,
- 👉 follow **particles transport** from **Geant4**,
- ⌛ parallelism, fast simulations, machine learning...



From Gauss to Gauss-on-Gaussino

LHCB-TDR-017

Introduce an experiment-independent layer!



Gaussino

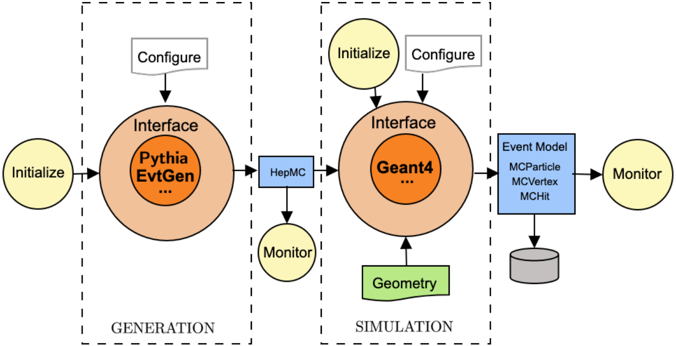
- new core simulation framework,
- only experiment-independent components,
- ideal test bed for new developments,

Gauss-on-Gaussino

- new version of LHCb simulation framework,
- based on Gaussino's core functionalities,
- adds LHCb-specific components and configurations,

Gaussino: keep what's good and works well...

i.e. the complete simulation framework architecture
well-served in the LHCb experiment



- integrated generation and simulation phase,
- similar modularity,
- similar event model,

Credit: LHCB-TDR-017 & LHCB-TDR-011

🎲 … and support new developments and ideas!

LHCB-TDR-017

Key concepts

- high-level configuration in python,
- multi-threaded event loop,
- generic detector description tools (DD4Hep),
- generic event model (EDM4hep),
- **new fast and ultra-fast simulations**,
- **machine learning libraries**,

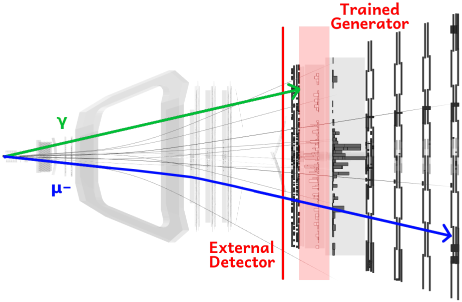
And many more! 😅

Let's see how this can be done!

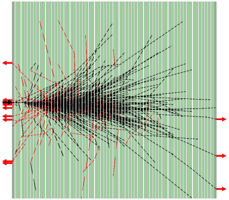
ML in fast simulations in particle transport

Fast simulations with Geant4...

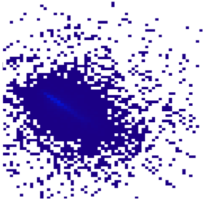
- stop detailed simulation in a particular region of the detector,
- use machine learning to produce a similar output,



What happens in Geant4?



What is actually stored?



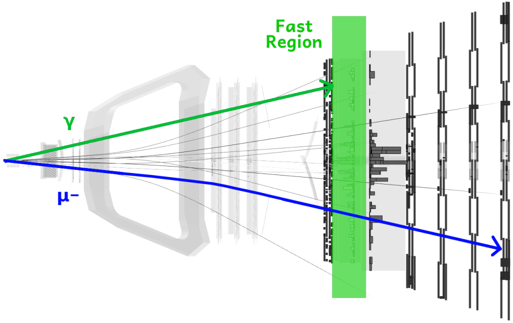
...and machine learning

- **train** a **ML model** to be able to produce the same output as Geant4,
- produce hits by **running inference** on the **generator**,

Fast simulations with Geant4...

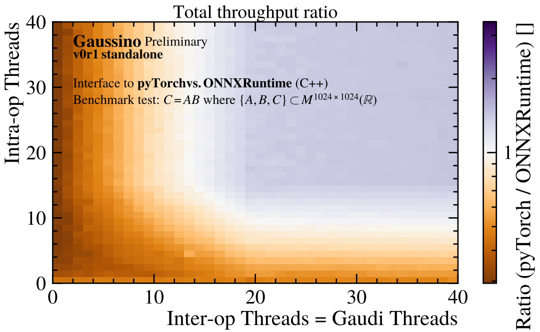
Recipe

1. Where?
   - region where the fast simulation takes place
2. What?
   - what types of particles should be tracked
3. How?
   - conditions when to fast simulate,
   - fast hit generation algorithm,



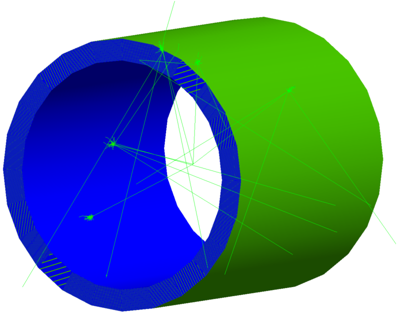
… and machine learning

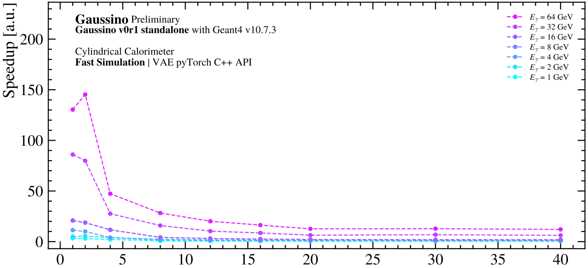
- **choose** the **best model** for the task:
  - Variational Autoencoder (VAE),
  - Generative Adversarial Network (GAN),
  - Diffusion models, etc.
- adapt & test **ML interface** to **ONNXRuntime** & **PyTorch** C++ backend,
- other ML libraries (TensorFlow, etc.) can be added later,



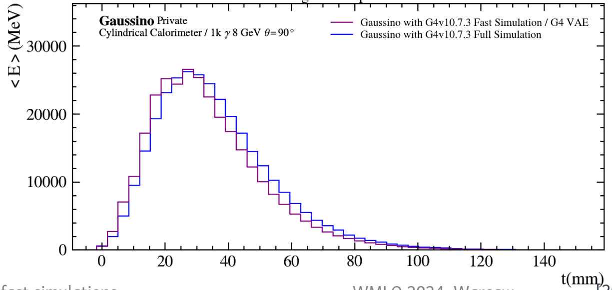
Generic models in Gaussino

Gaussino's adaptaion to CaloChallenge



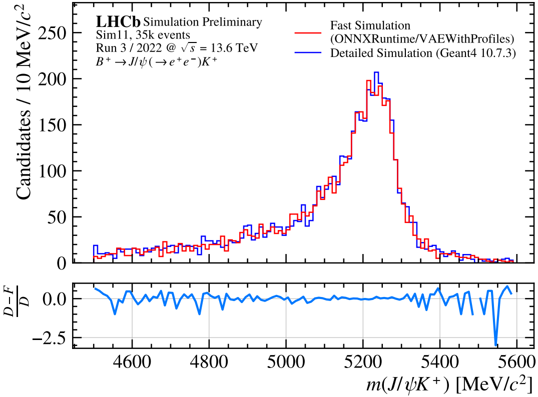


Longitudinal profile



Adaptation in LHCb

- **One model** only for e^+, e^- and γ in the electromagnetic calorimeter
- Up to 400$\times$ **speedup** in the simulation throughput
- Around 1-4% **energy difference** vs. Geant4-based simulation



What is the **status** of the new framework?

- first versions already released on CVMFS,
- aim to use for all LHCb simulation in the future,
- already used in Upgrade II studies,
- production tests on the grid ongoing.

🎲 Conclusions

- 👉 **ML-based fast simulations** became **crucial** for the future of HEP experiments,
- 👉 Porting new models from a prototype to a production-ready framework is **challenging**,
- 👉 Gaussino is a core simulation framework that can be used to
  - 👉 **test new ideas** and **implement generic models**,
  - 👉 seamlessly **integrate** with the existing infrastructure,
- 👉 **Full-scale production** in LHCb with the setup described in this presentation are planned by the end of 2024.

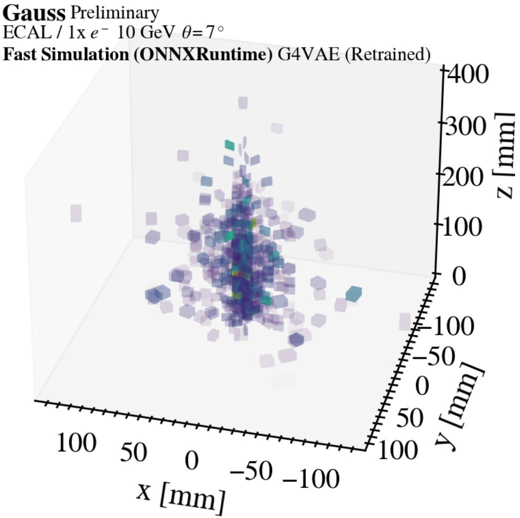
Thank you!

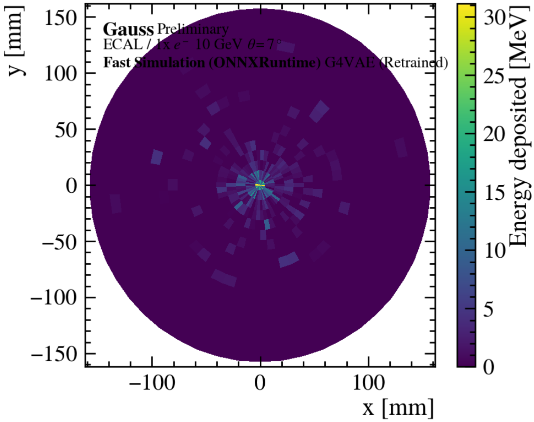
BACKUP

🎲 VAE With profiles

Redefined VAE model (VAEWithProfiles):

- 👉 **Input**:
  - 👉 shower tensor of shape (batch_size, 18 x 50 x 45),
  - 👉 particle tensor of shape (batch_size, latent_v + geo_v + θ-angle + pid + ϕ-angle).
- 👉 **Output**:
  - 👉 total hits number per shower,
  - 👉 total energy deposited in each shower,
  - 👉 1D profiles: z, ρ, ϕ, e.


Gauss Preliminary
ECAL / 1x e^- 10 GeV $\theta = 7^\circ$
Fast Simulation (ONNXRuntime) G4VAE (Retrained)
x [mm]
y [mm]
z [mm]


Gauss Preliminary
ECAL / 1x e^- 10 GeV $\theta = 7^\circ$
Fast Simulation (ONNXRuntime) G4VAE (Retrained)
x [mm]
y [mm]
Energy deposited [MeV]

🎲 Validation of the retrained models

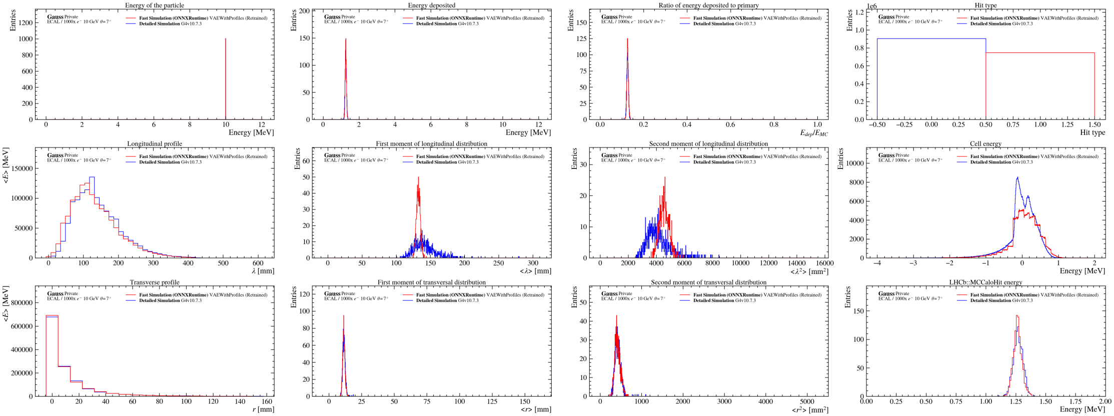
Simulation

- 👉 training datasets: MomentumRange pgun, 1-100GeV, θ=3.36-12.7°, ϕ=0-360°, pid = 11 (electrons), 22 (photons),
- 👉 inference datasets: FixedMomentum pgun, selected values & pids,
- 👉 truth datasets: FixedMomentum pgun, selected values & pids,
- 💡 measure performance and low & high level monitoring of the retrained models,
- 👉 digitize the truth and inference datasets with Boole.

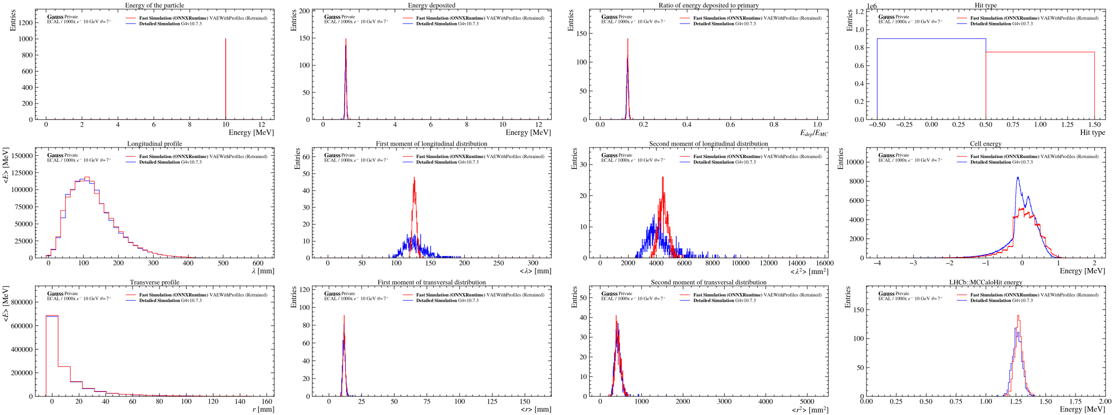
Reconstruction

- 👉 run the reconstruction on the digitized datasets,
- 💡 compare the reconstructed cluster energy with the truth.

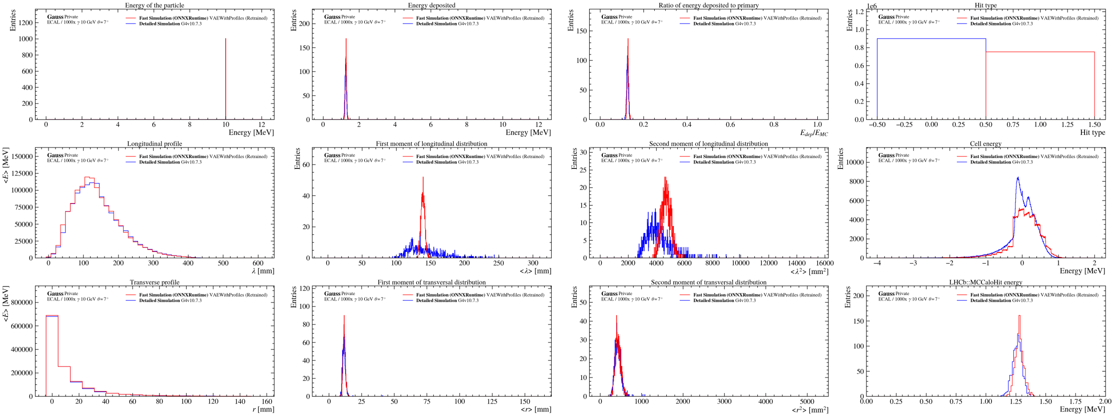
Simulation, Electrons, $\theta = 3.36°$



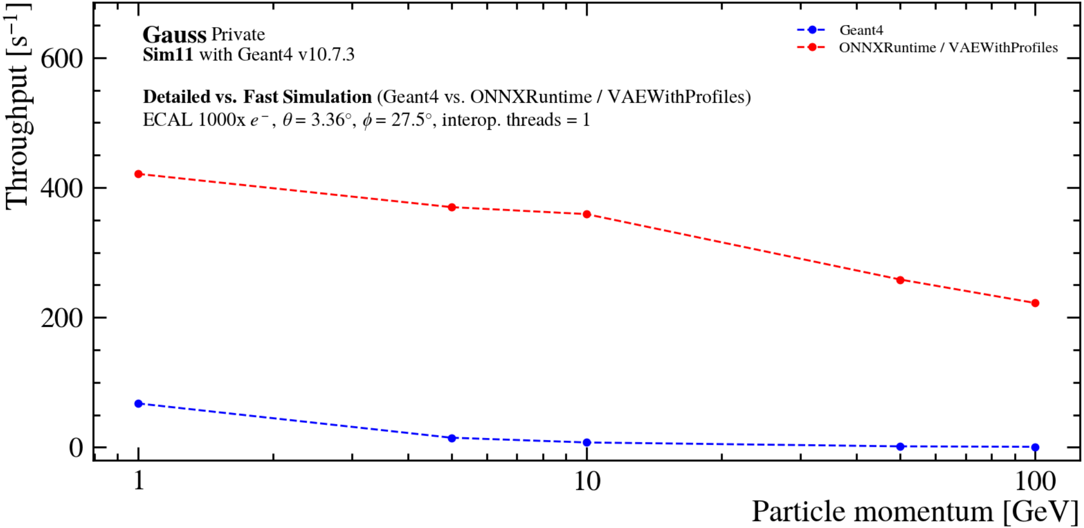
Simulation, Electrons, $\theta = 12.7°$



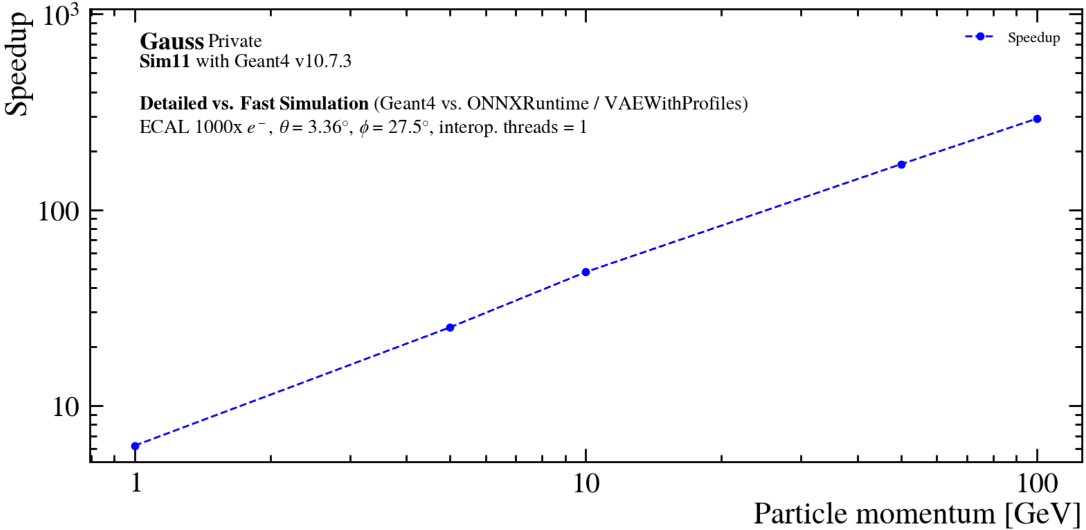
Simulation, Photons, $\theta = 12.7°$



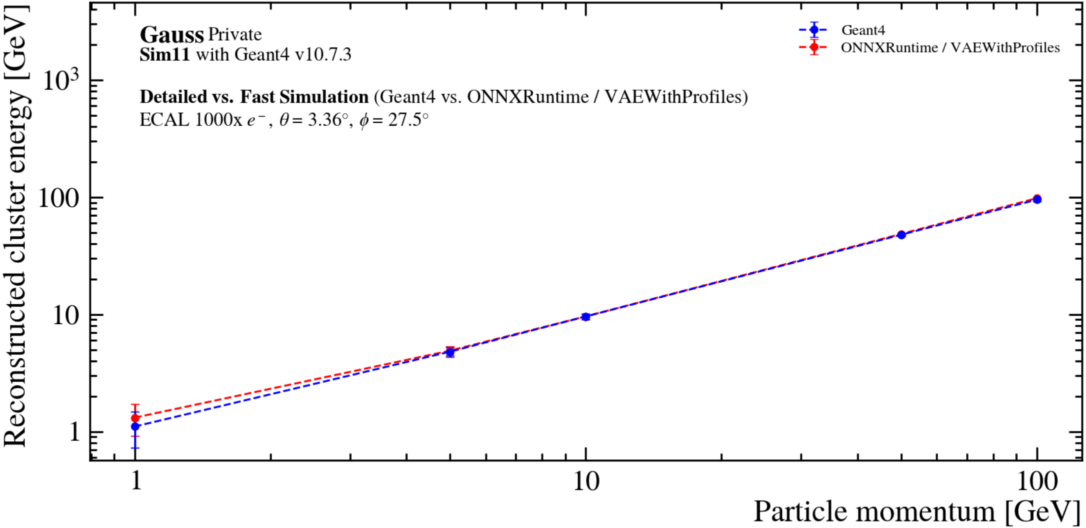
Simulation, Throughput



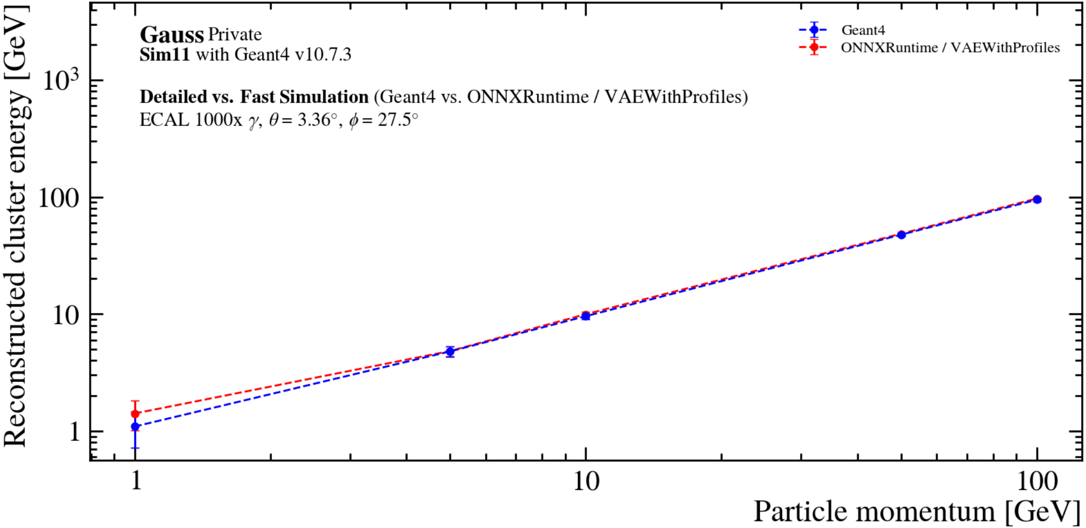
Simulation, Speedup



Reconstrution, cluster energy, electrons



Reconstruction, cluster energy, photons



Reconstruction, cluster ratio, electrons