NATIONAL CENTRE FOR NUCLEAR RESEARCH ŚWIERK

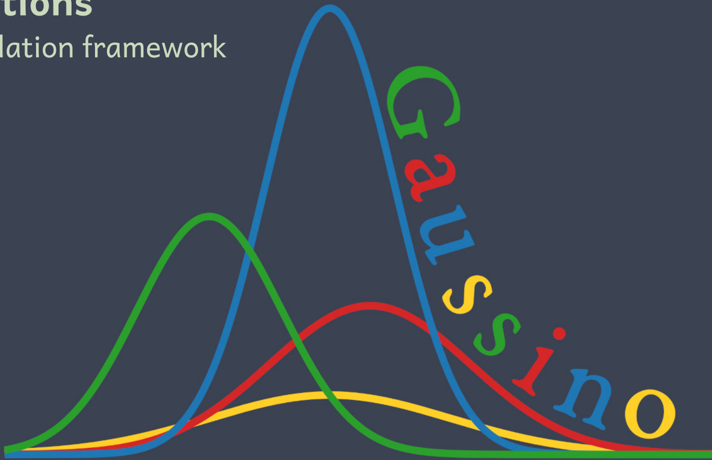
CERN

From prototypes to large scale detectors with Monte Carlo simulations

how to exploit the Gaussino simulation framework for detectors studies

by **Michał Mazurek**

on 15 June, 2023
PhD Seminar, Warsaw

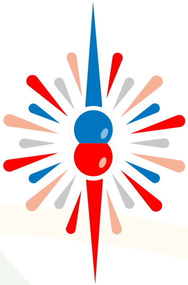


Monte Carlo simulations in HEP

Finding intangibles

Typical workflow in HEP

- **source of particles**
  - collisions in accelerators,
  - cosmic rays,

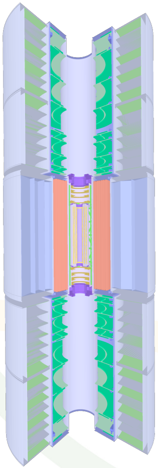


Credit: Phoenix Event Display & Picture with colliding atoms

Finding intangibles

Typical workflow in HEP

- **source of particles**
  - collisions in accelerators,
  - cosmic rays,
- **experimental apparatus**
  - calorimeters, trackers, ...

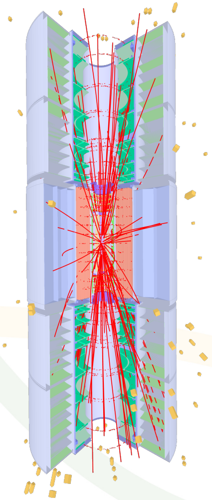


Credit: Phoenix Event Display & Picture with colliding atoms

🎲 Finding intangibles

Typical workflow in HEP

- 👉 **source of particles**
  - collisions in accelerators,
  - cosmic rays,
- 👉 **experimental apparatus**
  - calorimeters, trackers, ...
- 👉 **readout system**
  - controls and data acquisition,
  - event selection (hardware trigger),
- 👉 **data processing**
  - partial reconstr. (software trigger),
  - full event reconstruction (offline),

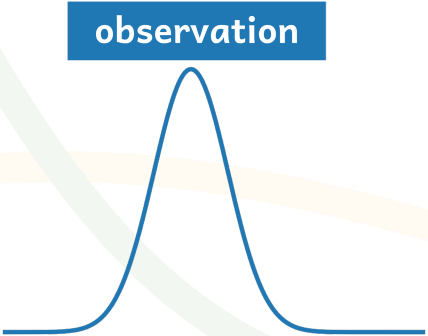


Credit: Phoenix Event Display & Picture with colliding atoms

Finding intangibles

Typical workflow in HEP

- **source of particles**
  - collisions in accelerators,
  - cosmic rays,
- **experimental apparatus**
  - calorimeters, trackers, …
- **readout system**
  - controls and data acquisition,
  - event selection (hardware trigger),
- **data processing**
  - partial reconstr. (software trigger),
  - full event reconstruction (offline),
- **offline data analysis**



Credit: Phoenix Event Display & Picture with colliding atoms

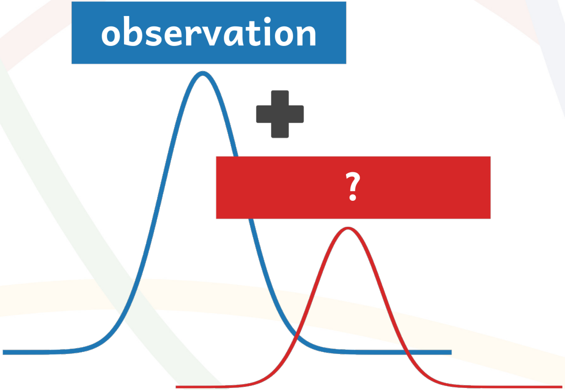
Understanding observations

Examples

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

What is needed?

MODEL



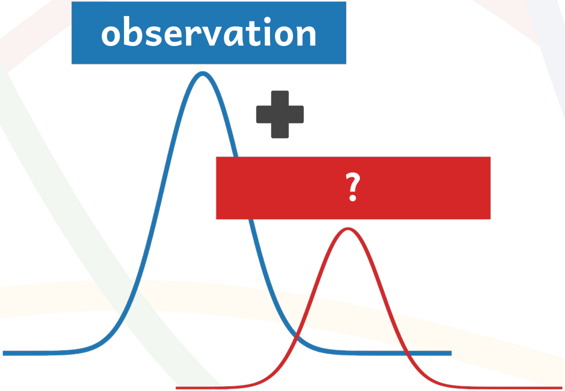
Stochastic nature

A lot of choices possible

- event with n particles involves $\mathcal{O}(10n)$ random choices,
- @LHC: $\sim$300 in each collision,
- stochastic particle transport and interaction with matter,

What is needed?

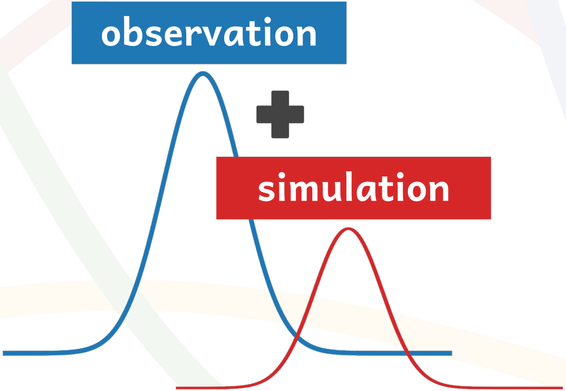
RANDOMNESS



🎲 Monte Carlo simulations to the rescue

MODEL ➕ RANDOMNESS
👉 MC SIMULATIONS

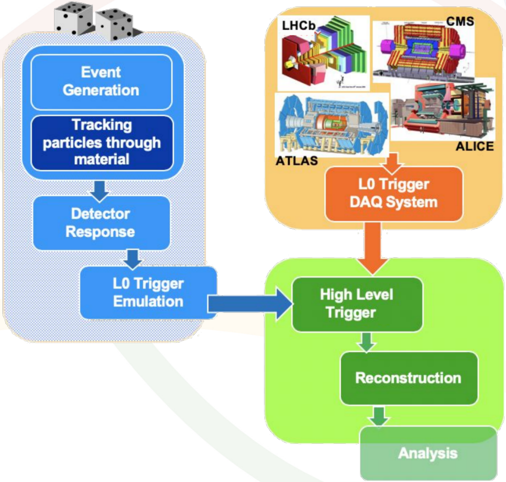
- 💡 sample **randomly** from a **given distribution** significant number of times,
- 💡 works with **difficult PDFs**,
- 💡 **error** does **not depend** on the number of **dimensions**,



Putting it all together

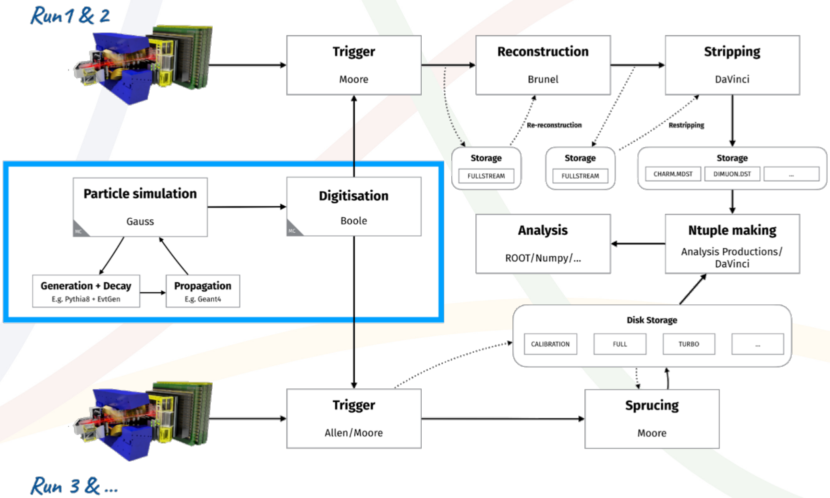
Simulation workflow in HEP

- **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,



Simulations for large scale detectors

Example: workflow in LHCb 😓

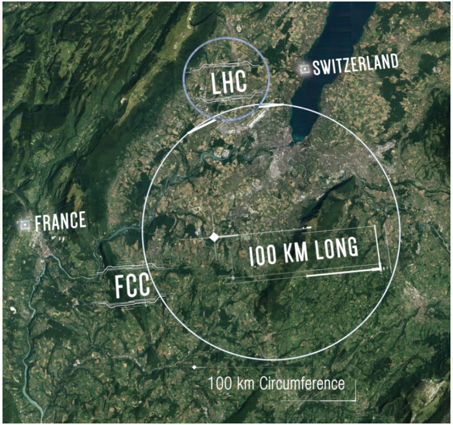


🎲 Simulation frameworks

Experiments and their frameworks

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

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



Credit: FCC picture

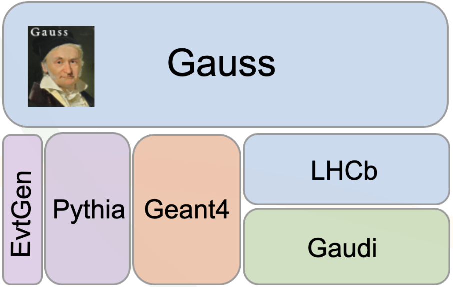
Core simulation framework for large scale detectors

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

— CERN SFT, 2015

Features as in Gauss

- 👉 **modular generation** phase,
- 👉 based on common **core software** framework: **Gaudi**,
- 👉 follow **detector transport** from **Geant4**,
- ⌛ parallelism, fast simulations, etc. to be added later...



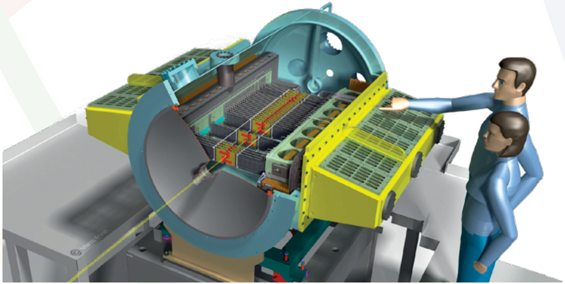
Early detector studies

👶

Planning with Monte Carlo simulations

Applications

- evaluate **physics reach**,
- develop **reconstruction**, **analysis** and other **applications**,
- assess **detector design choices**,

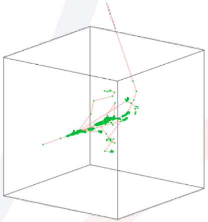


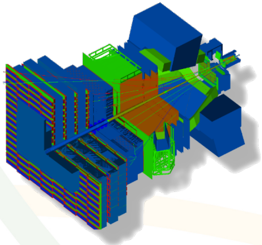
Credit: Marco Kraan, CERN Courier

🎲 Typical scenario for detector R&D studies

👶 **test beams** / **early studies** with sub-detector prototypes in **Geant4 standalone**,

🧔 porting to **the experiment's main simulation framework** once tested



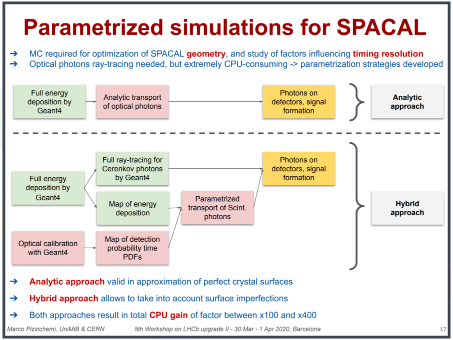


‼️ early simulations already very complex: optimizations, custom simulations, machine learning, etc.
😓 the path to the final detector is **not easy**: new software technologies, complex geometries, etc.

Example: CALO upgrade in LHCb

This is how it looks like in reality...

- custom simulations in place of Geant4-detailed simulation,
- parametrization of optical photons ray-tracing,

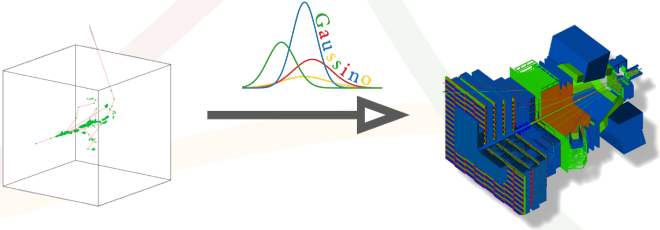


Credit: M. Pizzichemi, 5th workshop on LHCb Upgrade II, Barcelona

Core simulation framework for early studies

use the core simulation framework **for both**:
test beams and **final integration** in the experimental setup!

- to be used not only as a **toolkit**, but also in a **standalone mode**,
- **simplified geometry** and **minimal** list of **dependencies**,
- **transition** to the final integration **as easy as possible**,



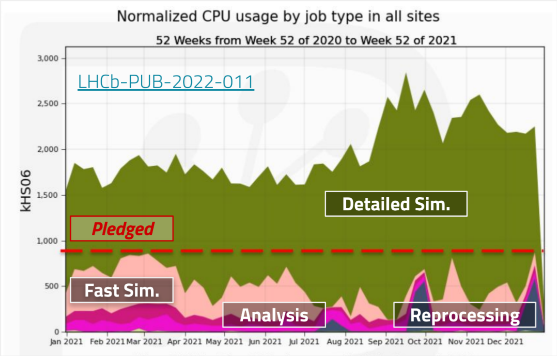
From Gauss to Gaussino

It all started in LHCb...

Landscape for Run3 and beyond

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

Simulation software upgrade needed!

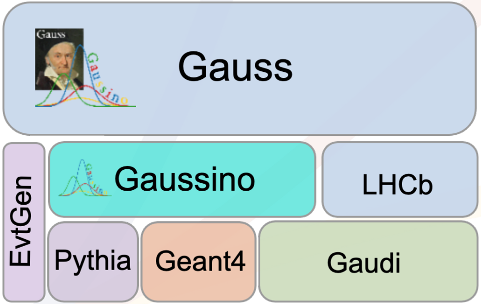


Credit: M. Barbetti, CHEP 2023 & LHCb-PUB-2022-011

From Gauss to Gauss-on-Gaussino

LHCB-TDR-017

💡 **Introduce an experiment-independent layer!**



Gaussino

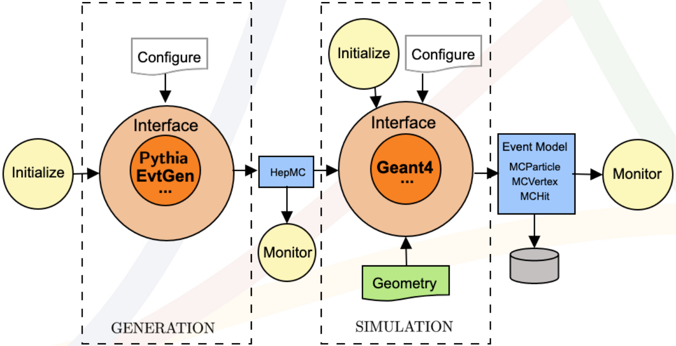
- new core simulation framework,
- only experiment-independent components,
- ideal test bed for new developments,

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

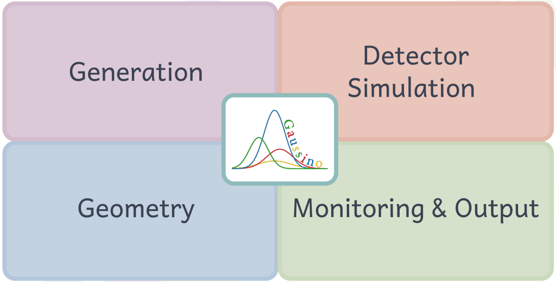
- high-level configuration in python,
- multi-threaded event loop,
- generic detector description tools (DD4Hep),
- generic event model (EDM4hep),
- new fast and ultra-fast simulations,
- machine learning,

And many more! 😅

Gaussino's main components

Configuration

- python configurables steering C++ classes,
- modular structure with 4 main configurables, one for each module

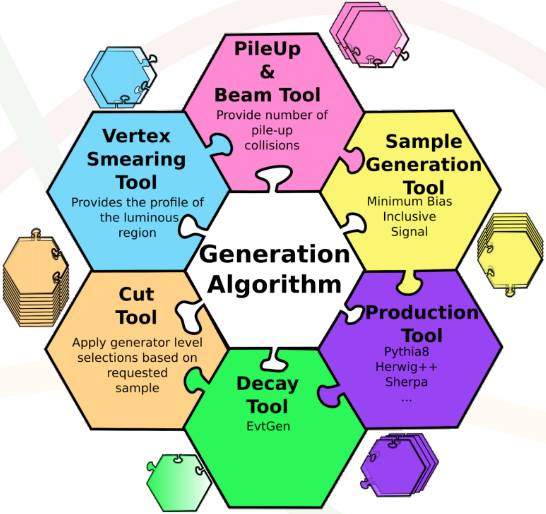


- Gaussino()
- GaussinoGeneration()
- GaussinoSimulation()
- GaussinoGeometry()

Generation phase

LHCb-PROC-2010-056

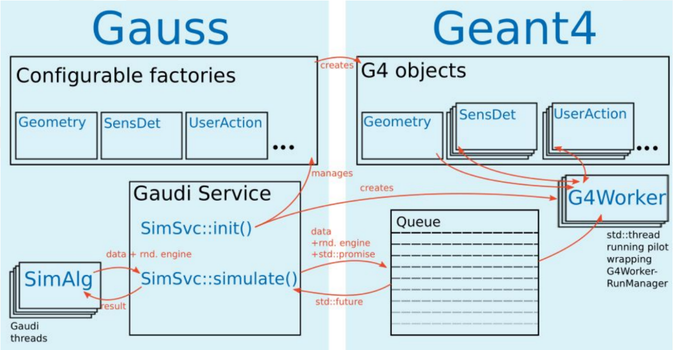
- highly modular as in Gauss,
- thread safety of generators,
- interface to Pythia8,
- some particle guns,



Detector simulation

LHCB-FIGURE-2023-010

- Geant4 with multi-threading,
- set of factories for G4 objects,
- flexible python configuration:
  - detailed simulation,
  - fast simulations,



Geometry

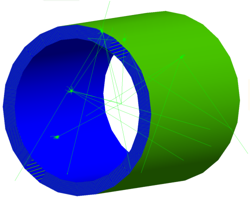
LHCB-FIGURE-2021-004

- interface to generic geometry description toolkit (**DD4Hep**) for full and complex geometries,
- **internal geometry** service for **simple geometries**,

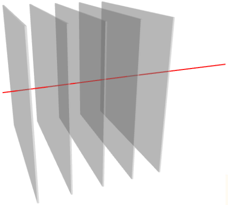
Geometry in **Gaussino**	
DD4hep	Internal

Many examples available!

- cylindrical calorimeter for **general-purpose detectors**



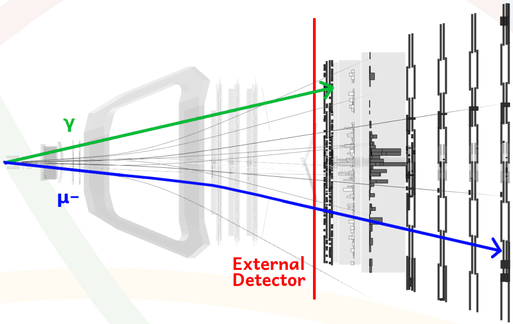
- tracking stations for **forward spectrometers**



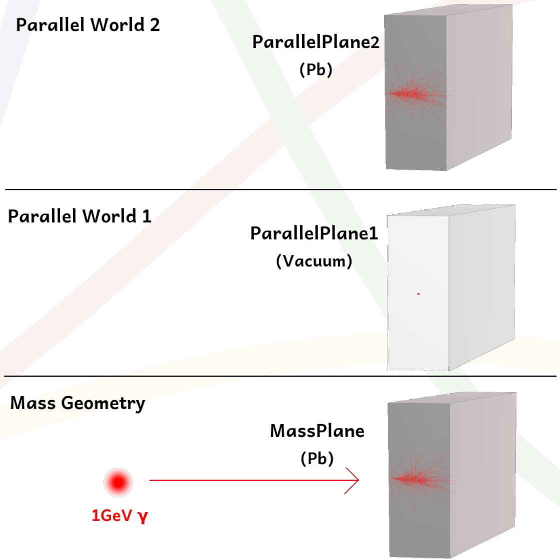
Producing custom datasets

LHCB-FIGURE-2021-004

- Required data for custom studies is not always present in the standard output.
- An abstract, external detector can be used as a collector of the required information at any position in the detector.
- A built-in mechanism can take care of potential volume overlaps by placing extra volumes in parallel geometries.

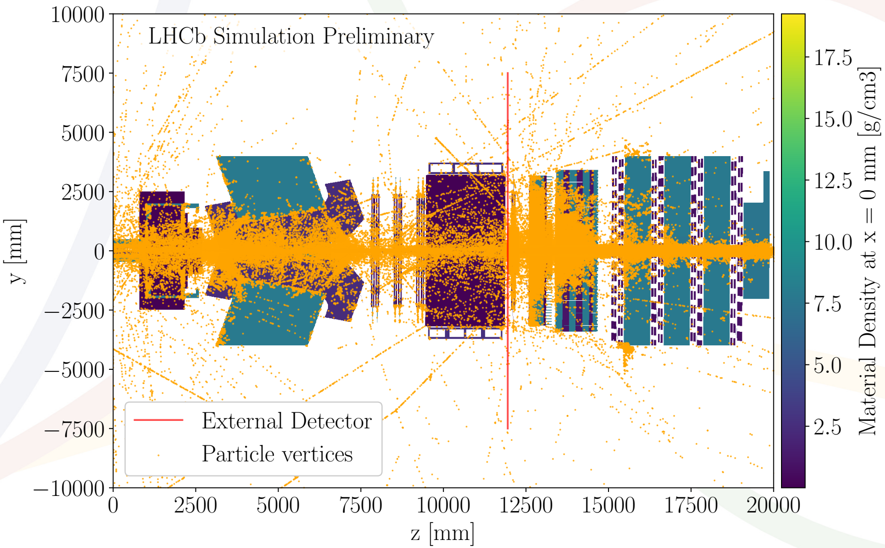


Cool feature: how to deal with volume overlaps?



Example: first production in LHCb

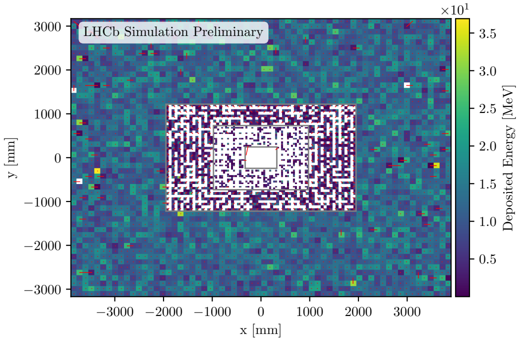
LHCB-FIGURE-2021-004



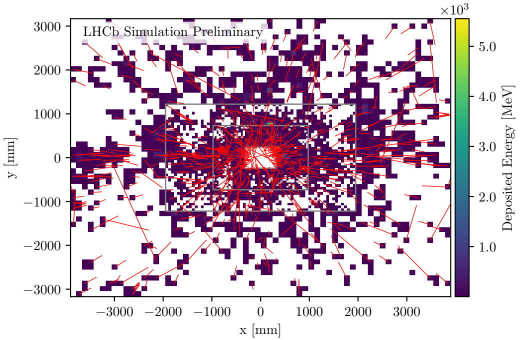
Example: custom datasets in LHCb

LHCB-FIGURE-2021-004

external plane (incident particles info) ECAL hits



Evenly-spaced grid of photons



Minimum bias sample

Visualization of geometry and data

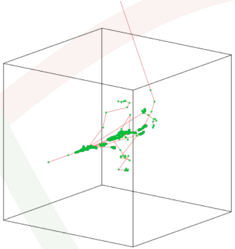
CERN-STUDENTS-Note-2022-205

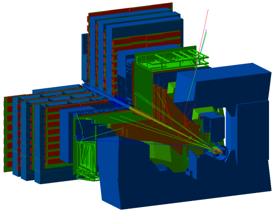
Geant4 visualization drivers

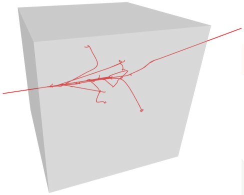
- available at **runtime**,
- works for **simulation data** only,

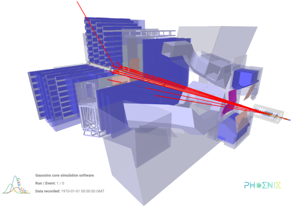
Phoenix event display

- available in an **external** tool,
- **simulated vs. reconstructed** data comparison possible,





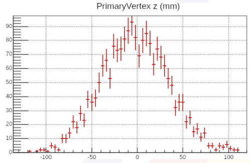


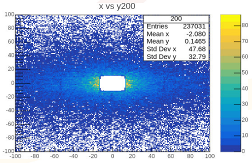


D.Popov EPJ Web Conf., 214 (2019) 02043
M. Szymański, B. Couturier EPJ Web Conf., 214 (2019) 05014

Monitoring & Output

- persistent output format,
- flexibility on what to keep,
- custom n-tuples,
- monitoring histograms,





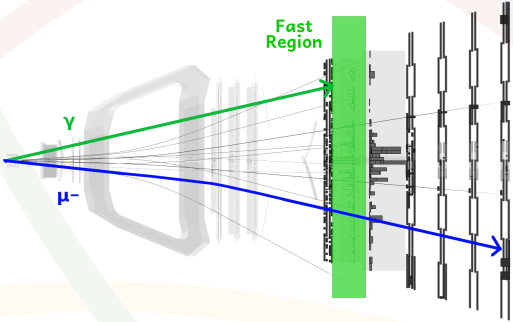
Example: LHCb validation

Machine learning in fast simulations

Interfacing fast simulations with G4

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,



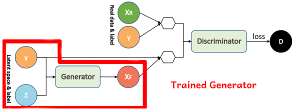
🎲 Machine-learning based fast simulations

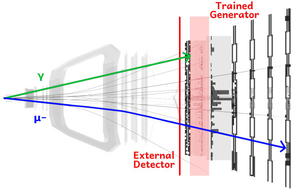
🔥 currently a very **hot topic** 🔥

- 👉 a lot of contributions in recent years,
- 👉 all the experiments are required to produce enough simulated samples,
- 👉 relatively high throughput and precision required,

Example: GANs in LHCb

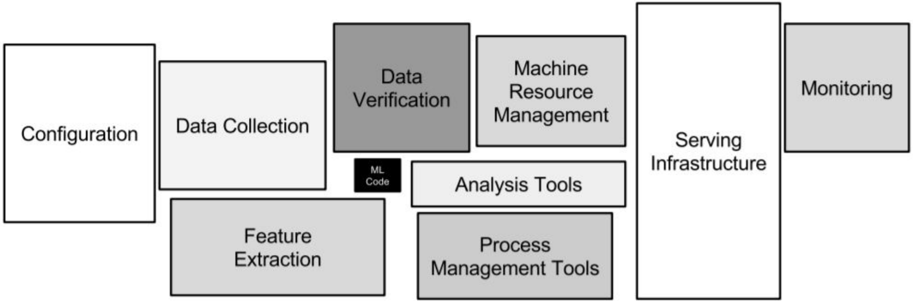
💡 use Generative Adversarial Networks (GANs) trained on the data produced by a detailed simulation to generate showers in the calorimeter simulatiom





Generic approach to machine learning

How do we deploy, maintain and serve ML models?

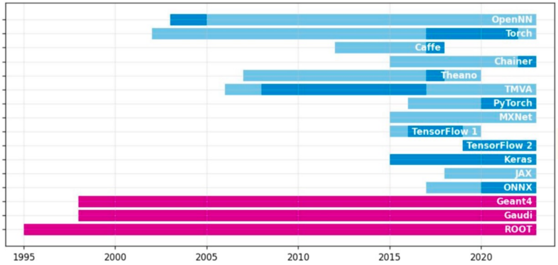


Credit: image from Medium

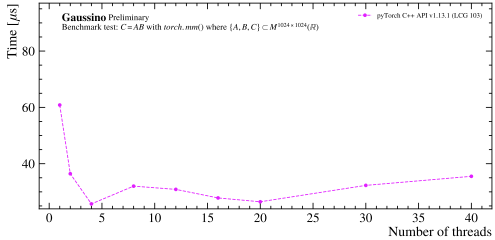
in industry: **machine learning operations**

Generic ML model serving

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



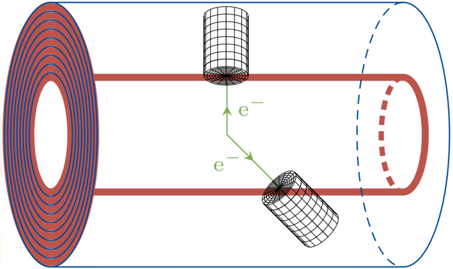
- first candidate: **pyTorch C++ API**,
- preliminary throughput tests:



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

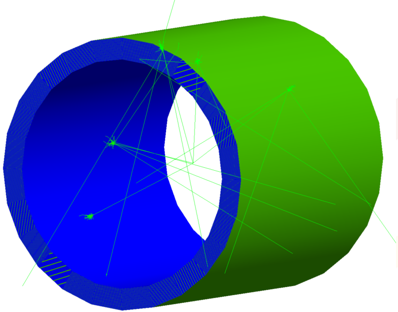
Generic ML model validation

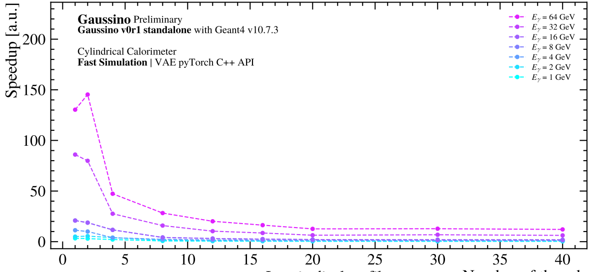
- new Geant4 initiative: **CaloChallenge**
- train on **experiment-agnostic training dataset**
- **compare** various **models objectively**,
- **retrain** the chosen model on the **target geometry**!



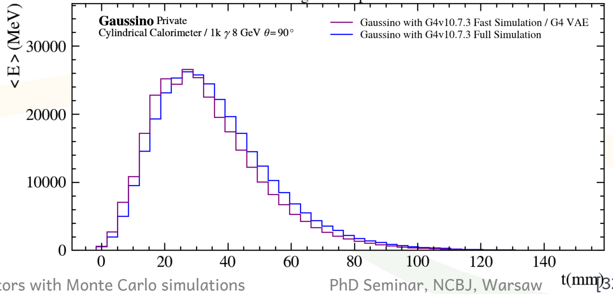
Credit: W. Pokorski (Geant4 Technical Forum)

First benchmarks in Gaussino





Longitudinal profile



Summary & future plans

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

PROTOTYPE → COMMISSIONING → PRODUCTION

we're here!

Gaussino v1r0
Gauss v61r0
very soon!

PROTOTYPE:
- multi-threading prototype,
- early benchmarks,
- first DD4hep sub-detectors,

COMMISSIONING:
- extensive testing,
- Gaussino standalone,
- configuration revisited,
- stable software,
- custom simulations,

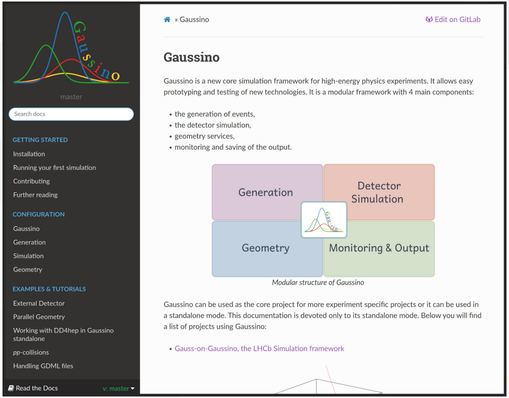
PRODUCTION:
- full testing,
- port all needed functionality from Gauss (Sim10). e.g. full generators palette,
- test productions with LHCbDirac,
- full scale physics validation,

Documentation

https://gitlab.cern.ch/Gaussino/Gaussino

https://gaussino.docs.cern.ch/

- each new development in Gaussino is documented
- the documentation provides the description of:
  - how to install and run simulations
  - high-level python configuration
  - simple examples
- versioning of the documentation



Conclusions

- Gaussino is the **new core simulation framework** extracted from the LHCb simulation framework.
- Can be used as the **core simulation toolkit** for **large scale detector frameworks** in HEP...
- ... **or** it can be used in a **standalone mode** as an ideal test bed for **early detector studies** and **new software technologies**:
  - easy configuration in python,
  - integrated generation and detector simulation phase,
  - internal geometry service,
  - support for fast simulation, machine learning, etc.
- **Gaussino is mature enough for its first beta release!**

Thank you!