CERN IT Machine Learning Infrastructure Workshop:

Inputs from *LHCb*

October 11th 2023

Simon Akar

on behalf of the LHCb collaboration

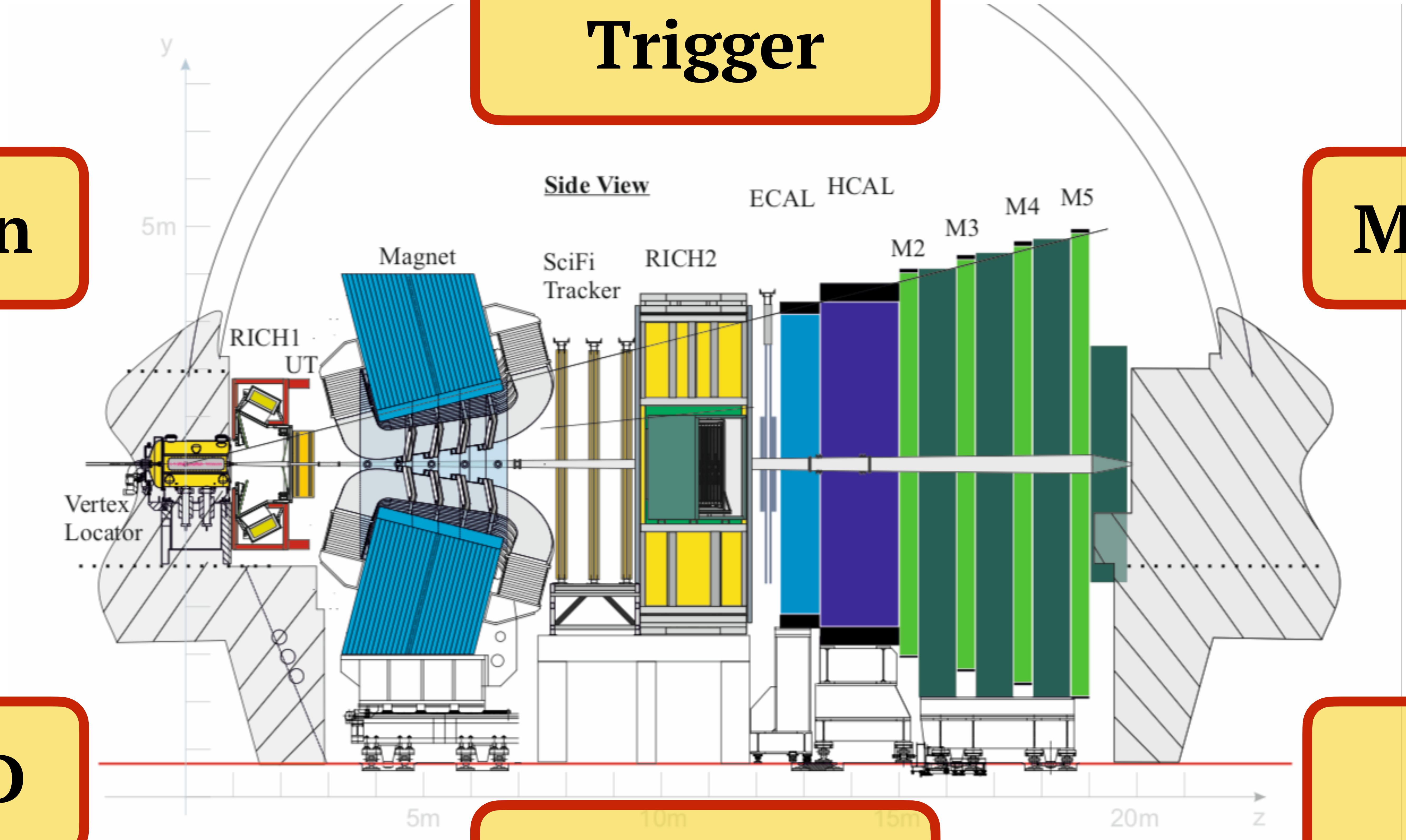
Machine Learning @ LHCb

- **A long and diversified history of ML applications in LHCb:**

Trigger

Simulation

Monitoring



Particle ID

Flavour tagging

Analysis

Machine Learning @ LHCb

- **A long and diversified history of ML applications in LHCb:**

Trigger

Simulation

Monitoring

Disclaimers:

- **Not an exhaustive overview of ML applications in LHCb, but a highlight of relevant use cases for today's workshop**
  - (I) Identify the challenges for a fully exploited implementation of ML/DL
  - (II) Which common services / solutions would we most benefit from as a community
- **Wide range of concepts covered in this talk: Detailed discussions on certain specific technical considerations might be better covered offline**

Particle ID

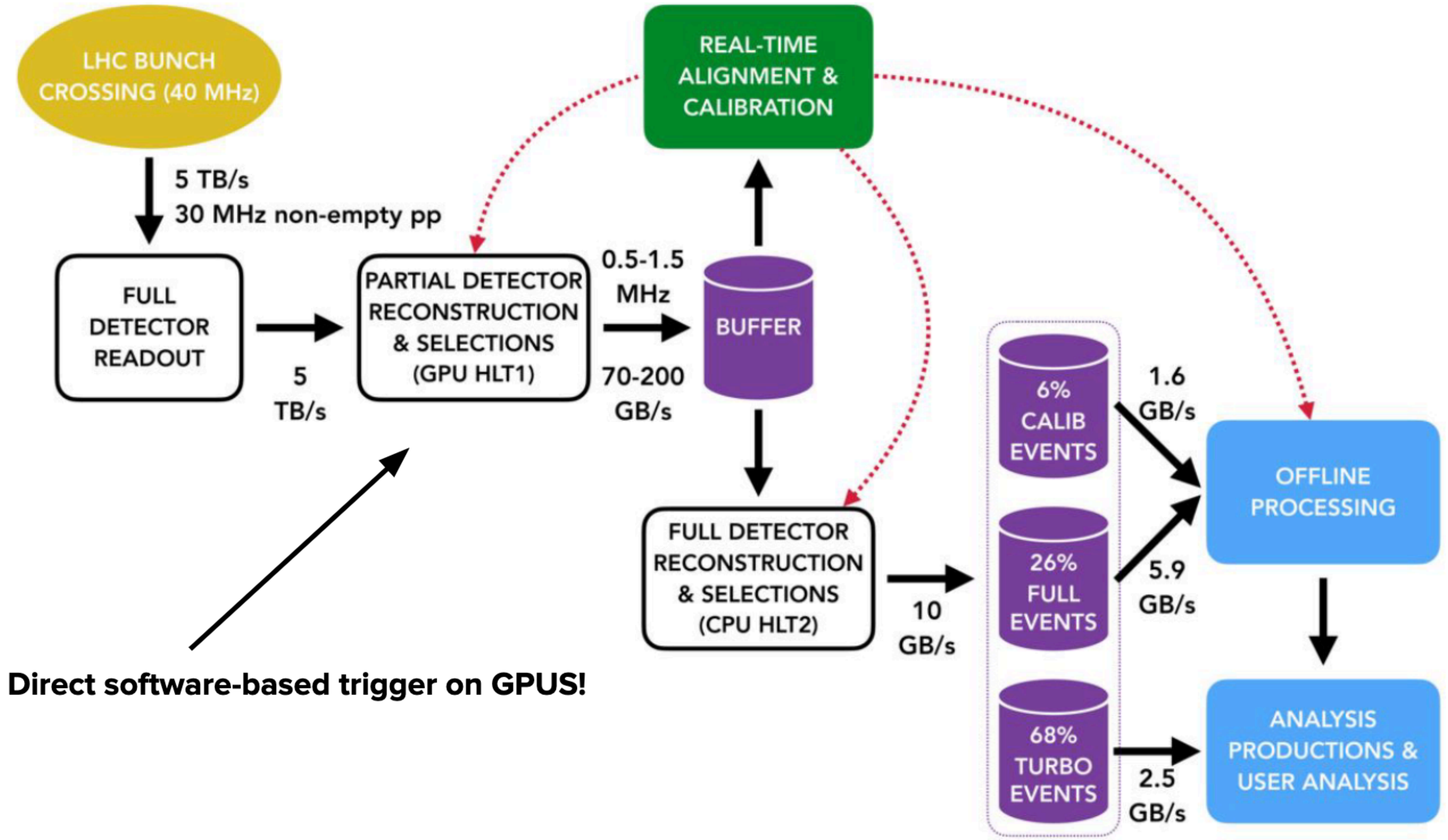
Flavour tagging

Analysis

Machine Learning @ LHCb

Trigger & Online ops.

- **ML-based algorithms (BDT) in inclusive particle selections already during Run1 & Run 2** covering the majority of the collaboration's published analyses in Run 1 already [*LHCb-PUB-2011-016*, *arxiv:1510.00572*]
- **New paradigm in Run 3 with full software trigger implementation** [*CERN-LHCC-2018-014 ; LHCB-TDR-018*]
  - 326 GPUs reduce incoming data rate from 5 to approximately 0.1 TB/s
  - All subdetectors data available at trigger level
  - Opened window for **ML application (inference) at earliest selection level as possible directly in the online environment**



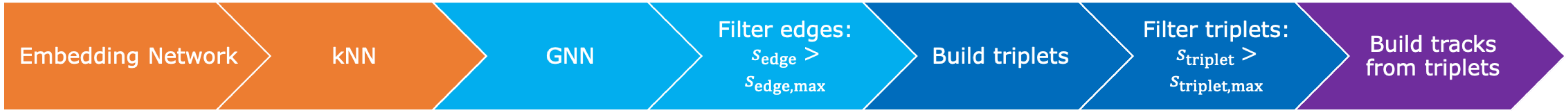
[*LHCb-FIGURE-2020-016*]

Machine Learning @ LHCb

Trigger & Online ops.

Several on-going efforts to implement ML algorithms inside LHCb online system

- **High-throughout Graph Neural Network track reconstruction at LHCb:** [*talk@CTD2023*]
  - **Track reconstruction** in the Velo (high-granularity tracking system)
  - Using GNN pipeline is based on the work of the Exa.TrkX collaboration



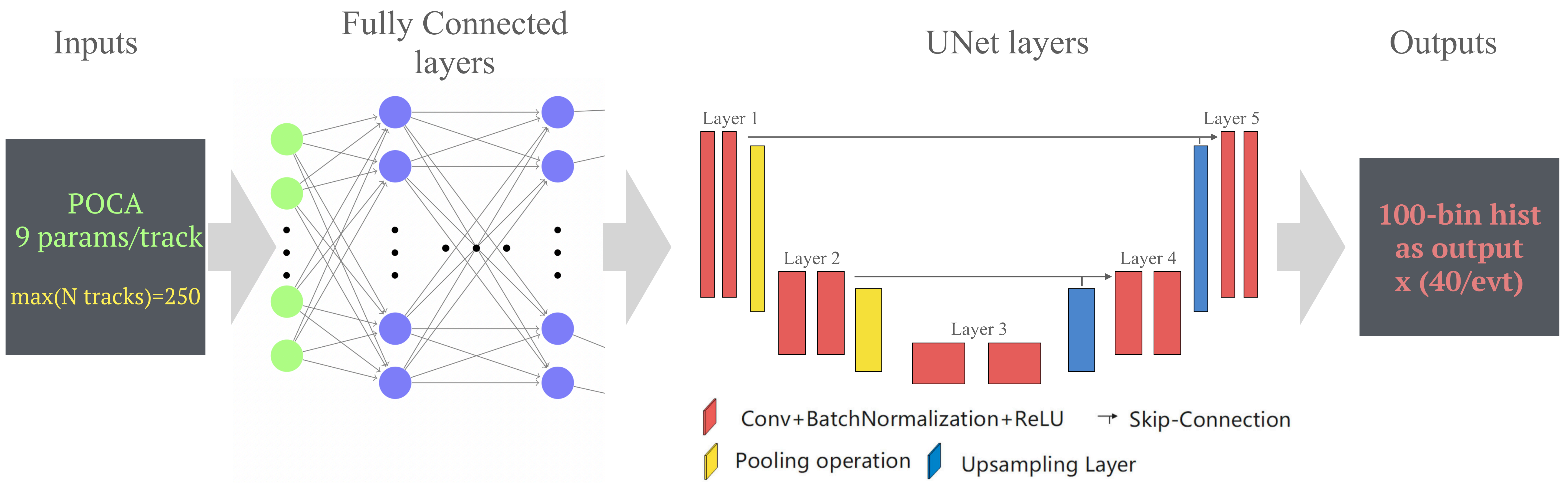
  - Training performed using **PyTorch** on **local ressources** (LIP6 Paris — Sorbonne Université)
  - Inference on low-level features
  - High parallelization over hits / edges ⇒ adapted to GPUs
  - Very promising preliminary physics performances
  - **On-going R&D to run inference on GPUs in Allen (HLT1)**
    ↳ **what throughput on HLT1 GPU farm** (identify potential bottlenecks for future upgrades)
    ↳ **study the possibility to extend approach to full detector**

Machine Learning @ LHCb

Trigger & Online ops.

Several on-going efforts to implement ML algorithms inside LHCb online system

- **DNN for finding primary vertices in *pp* collisions at the LHC:** [*talk@CHEP2023*, *arxiv:2309.12417*]
  - **Identify PV positions** from tracks low-level features
  - High parallelization over tracks / events ⇒ adapted to GPUs
  - **Non trivial hybrid MLP + CNN architecture**



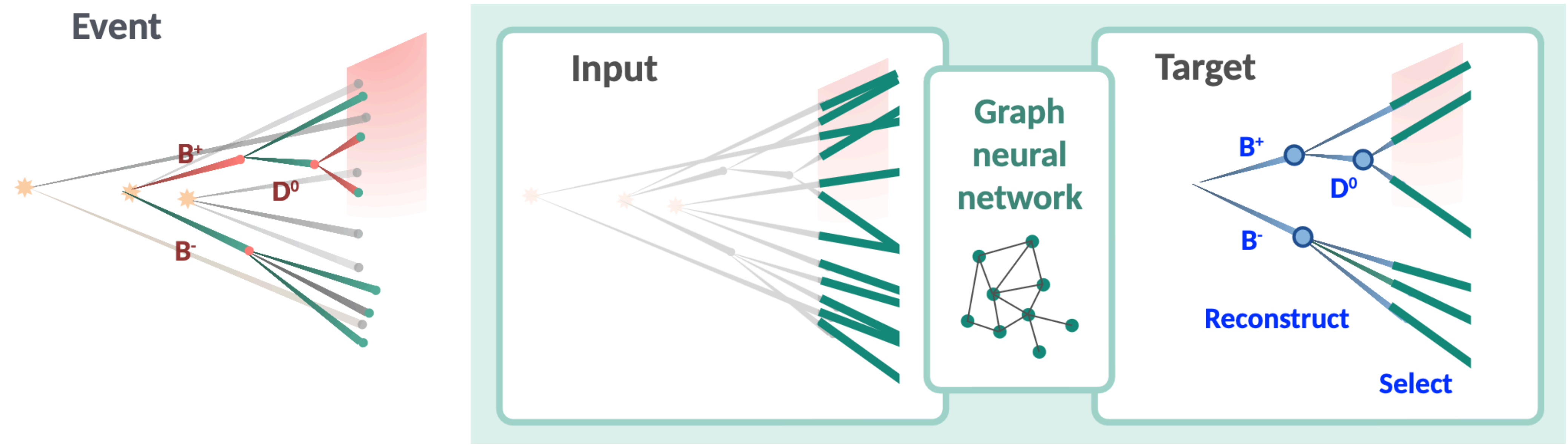
  - **Common training platform for LHCb and ATLAS**
  - Training performed using **PyTorch** on **local ressources** (University Cincinnati)
  - Very promising preliminary physics performances
  - **On-going R&D to run inference for LHCb on GPUs in Allen (HLT1)**

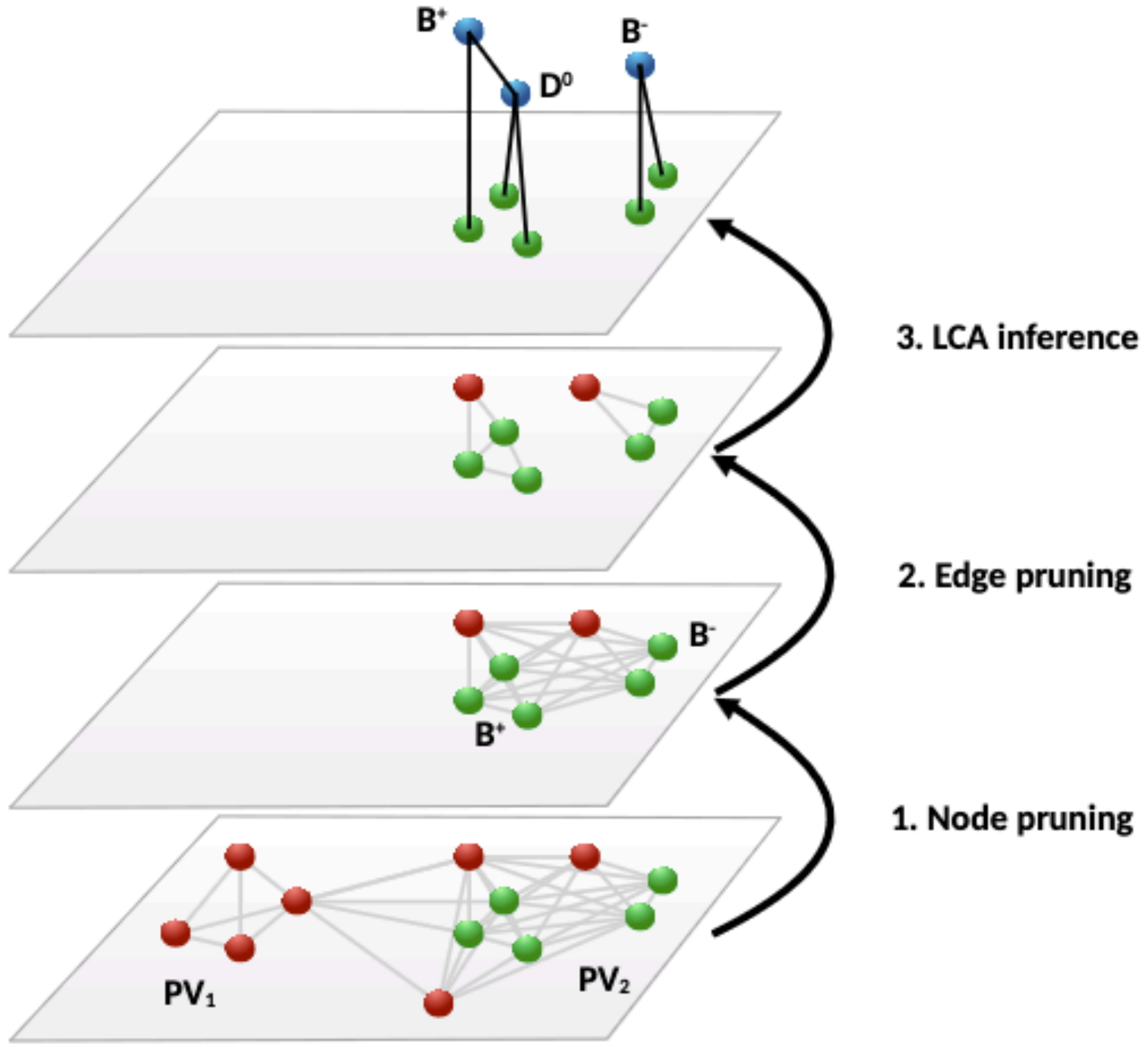
Machine Learning @ LHCb

Trigger & Online ops.

Several on-going efforts to implement ML algorithms inside LHCb online system

- **Graph Neural Network for Full Event Interpretation at LHCb** [*talk@CHEP2023*, *arxiv:2304.08610*]
  - Proof of concept: **Reduction of event size** by a holistic one-go analysis of the full event





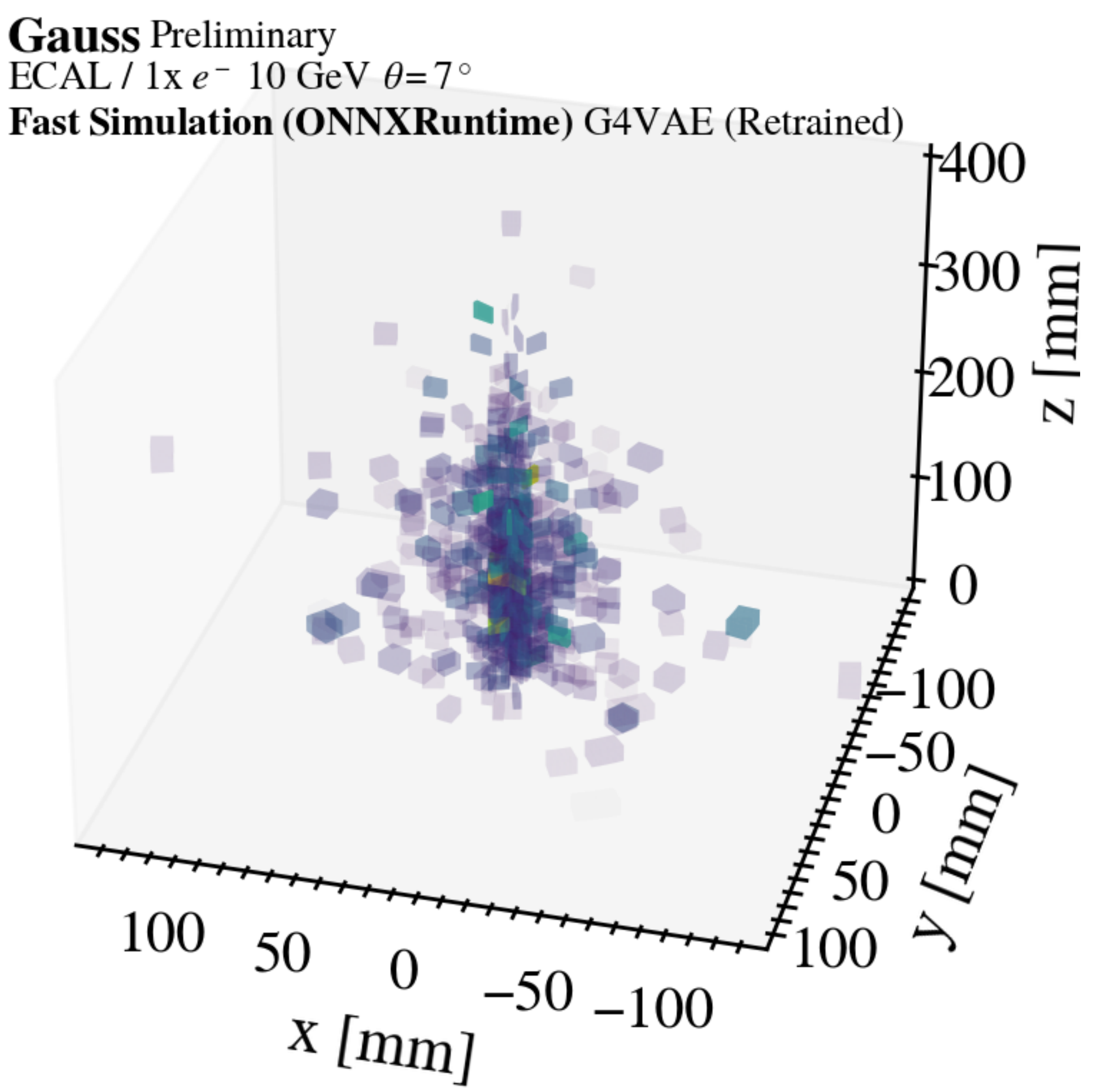
  - Training performed using **TensorFlow**
  - Ongoing detailed performance and timing studies
  - **Resources for model training were difficult to obtain**:
    Lack of adequate ressources @CERN (training over several days)
    ↳ Finally used ***Future SOC Lab*** cluster with docker containers for librairies

Machine Learning @ LHCb

Simulation

Computing resources too limited to use GEANT detailed simulation for the large simulated samples needed!

- **Fast simulation:** Replace Geant4 simulation with ML-generated output in specific area of the detector
  - Typically calo shower generation with GANs [*talk@CHEP2019*]
    ↳ Energy deposit generation ⇒ same pipeline as data-taking
  - Similar efforts among HEP experiments [*CaloChallenge workshop*]
  - **Geant4/CERN-SFT initiative** to train on experiment-independent datasets to compare various models objectively: VAEs, GANs, Diffusion models, Normalizing Flows [*talk@EP-SFT meeting*]
  - LHCb/Gaussino add the infrastructure and retrain on the target geometry [*talk@CHEP2023*]
  - **Challenges**: maintainability, large models, complex inference & retraining infrastructure
  - **Desiderata**: training accessible from different institutes

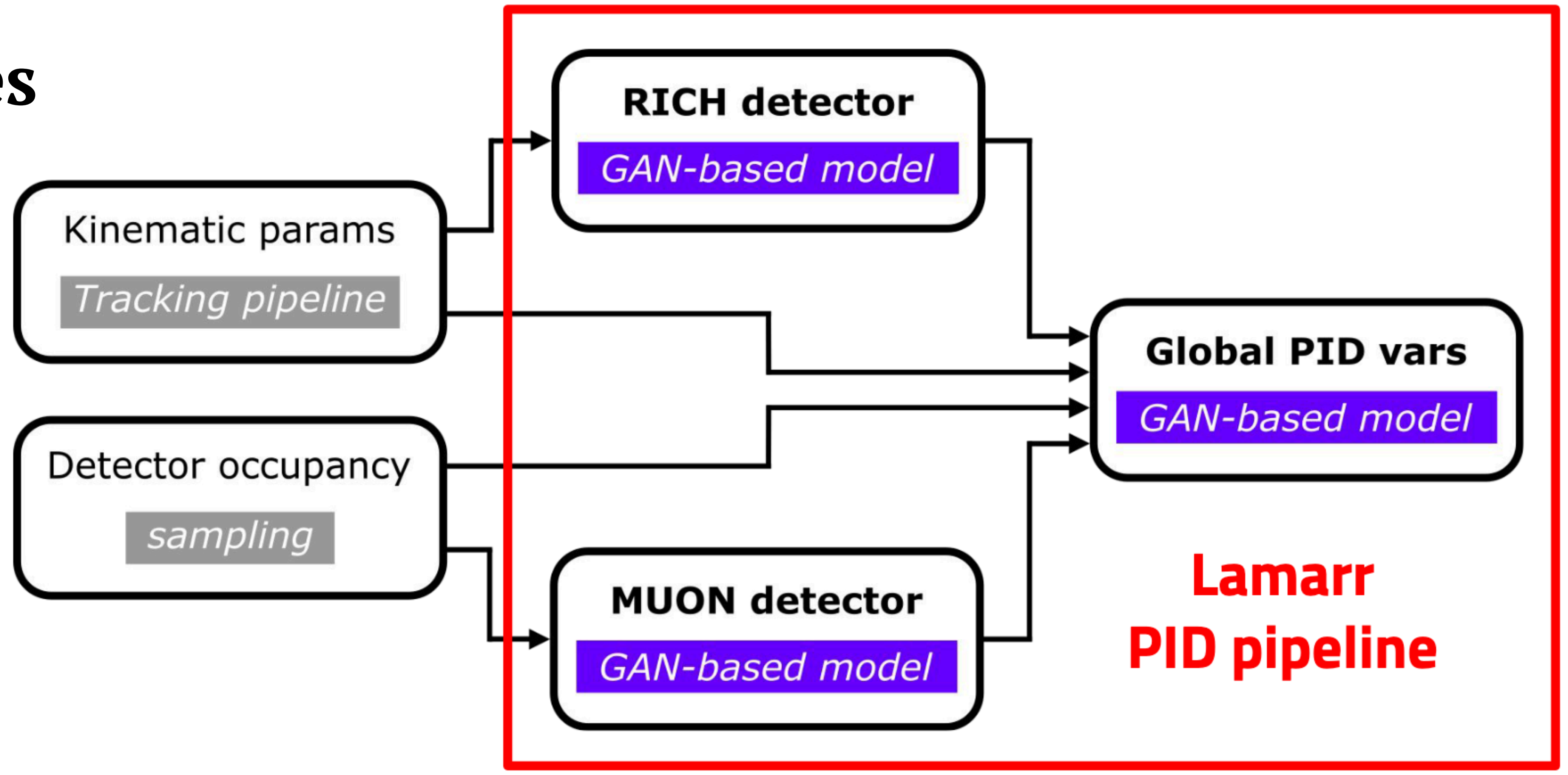


Machine Learning @ LHCb

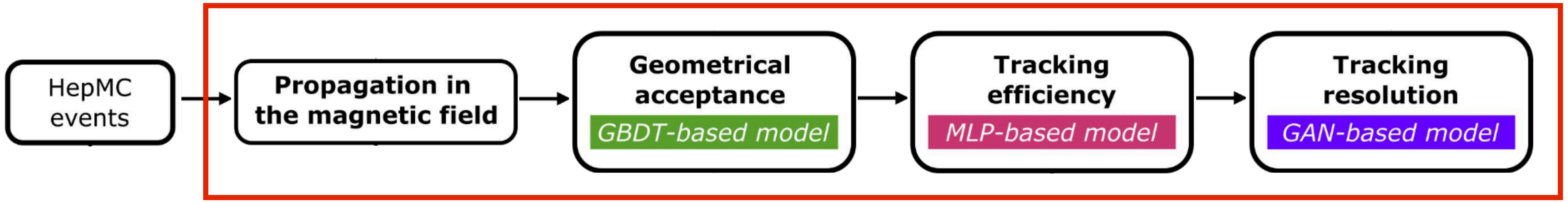
Simulation

Computing resources too limited to use GEANT detailed simulation for the large simulated samples needed!

- **Ultra-fast simulation: LAMARR** [*talk@CHEP2023*]
  - **ML models used to generate reconstructed quantities** (instead of energy deposit like for e.g. in GANs)
  - Scalar features ⇒ much smaller models
  - Inference on CPU possible, but using GPU would allow high parallelization!
  - **Pipeline of multiple ML models** is needed: GBDT, MLP, GAN, RNN, transformer…



Lamarr
Tracking pipeline



Machine Learning @ LHCb

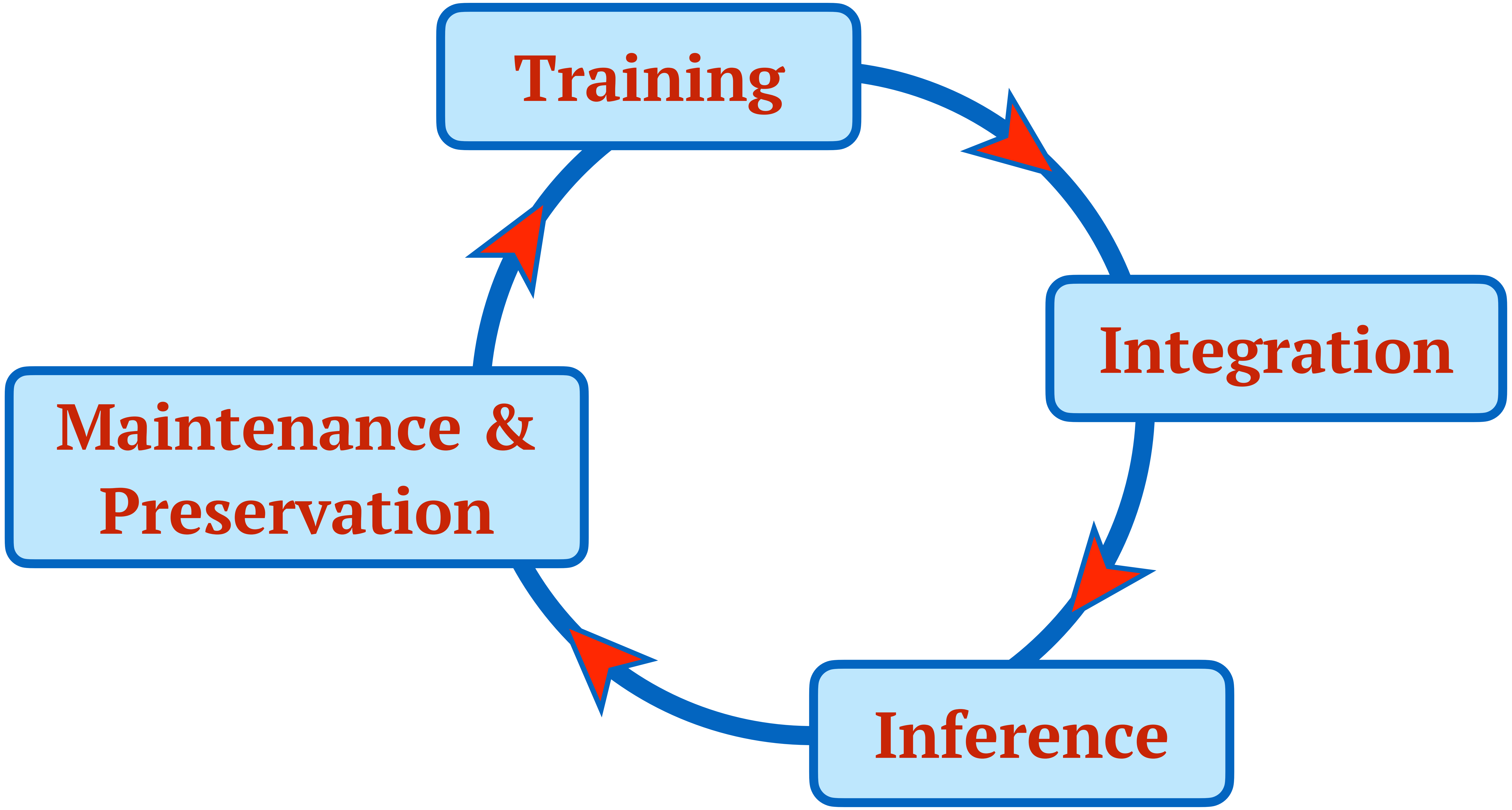
Particle ID **Flavour tagging**

- **ML-based algorithms used already for many years in general performance tools**
  - PID classification: [*Int. J. Mod. Phys. A 30, 1530022 (2015)*]
  - FT algorithm: **MLP** [*arxiv:1602.07252*], **BDT** [*Eur.Phys.J.C 77 (2017) 4, 238*]
- **Recent developments to improve robustness & performances**
  - PID classification: *Robust Neural Particle Identification Models* [*arxiv:2212.07274*]
  - FT algorithm: *Fast inclusive flavo(u)r tagging at LHCb* [*talk@CHEP2023*]

Analysis

- **Plethora of techniques and models applied throughout the LHCb analysis landscape**
  - Typically simple models (BDT) thanks to the inherent high signal/background ratio in LHCb's environment
  - Developed and using fitters on GPU to perform fits on millions of events (e.g. charm analyses)

Usage of ML: current & future challenges



Efficient and sustainable exploitation of ML/DL presents challenges at various steps

Common solutions among CERN collaborations is paramount!

Usage of ML: current & future challenges

Training

- **Currently most of the ML/DL trainings performed using resources available at local institutes/national facilities**
  - Models still relatively small (compared to industry)
  - SWAN system found to have some limitations (e.g. long trainings)
  - Need **robust & flexible pipelines for updated trainings, especially for production applications**

- **Multi-GPU batch system support would be beneficial**
  - Hyperparameters optimisation (lunch hundreds of trainings with different configuration)
  - Distributed computation (e.g. GAN shower generation for simulation)
  - Requires CUDA-enabled software packages available:
    ONNXRuntime, PyTorch, Tensorflow, VTK, ROOT, PyCUDA, cupy

Usage of ML: current & future challenges

Integration

- **Integration of ML/DL models in software is not straightforward**
  - Most LCG stacks are not GPU-enabled
  - **Most libraries already available in LCG_10Ncuda stacks**
    recent need for additional packages for GNN and multi-GPU processing: **cugraph, nccl, cutlass**
  - LCG cuda-enabled stacks built with **gcc11**, LHCb production compiler uses **gcc12**
    [very responsive communication with LCG]

- **Expertise from IT (core software engineers with ML expertise)**
  - Understanding which models are particularly efficient on which architectures (CPU vs GPU) and why?
    ↳ Profiling computing resources to make efficient use of models processing data

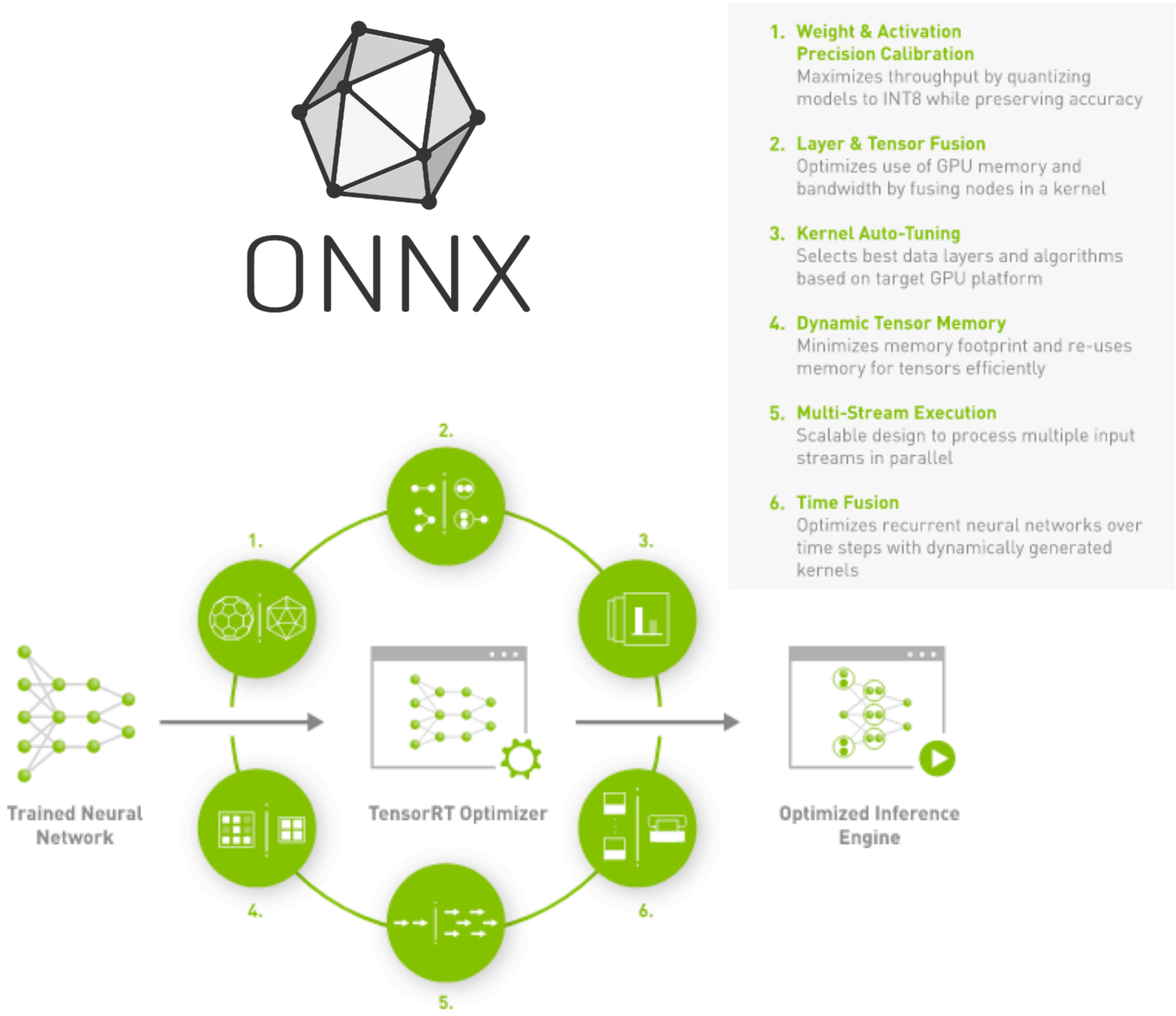
Usage of ML: current & future challenges

Inference

- **Inference (production systems) is currently the main challenge:**
  - Needs to be **fast, flexible & easily maintainable**
  - Standardized ML-model data format: e.g. **ONNX**
  - WIP on generic interface allowing access to desired backends: PyTorch C++ API, ONNXRunTime, TMVA::SOFIE… [*talk@EP-SFT meeting*]

ONNX

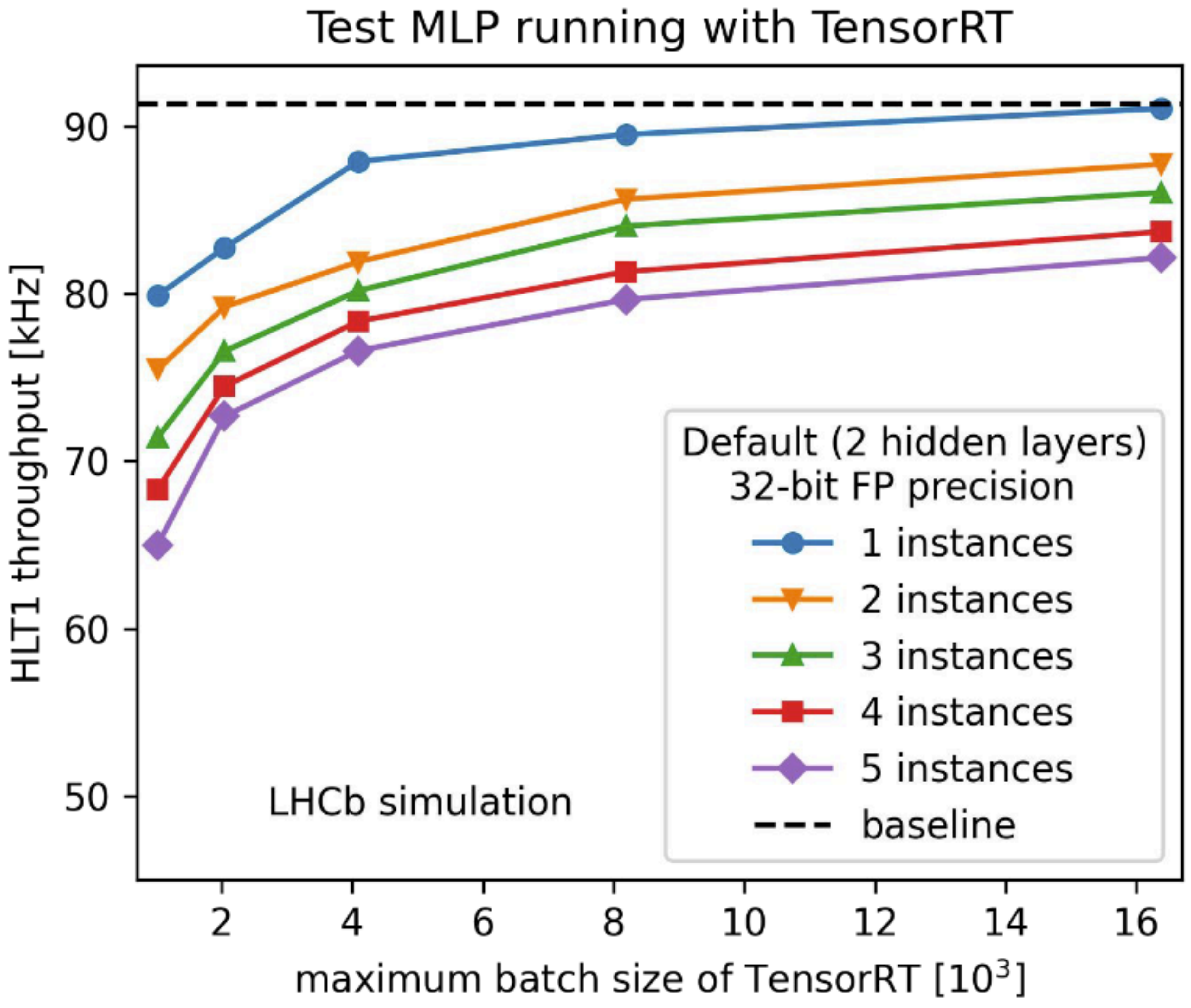
- **Ongoing developments / studies inside LHCb:** [*talk@CHEP2023*]
  - Inference on **GPUs** (NVIDIA A5000) using **TensorRT**
  - Benchmark for a **simple MLP** for ghost track probability
    17 features — 2 hidden layers (25x20) — 1 output



Usage of ML: current & future challenges

Inference

- **Inference (production systems) is currently the main challenge:**
  - Needs to be **fast, flexible & easily maintainable**
  - Standardized ML-model data format: e.g. **ONNX**
  - WIP on generic interface allowing access to desired backends: PyTorch C++ API, ONNXRunTime, TMVA::SOFIE… [*talk@EP-SFT meeting*]
- **Ongoing developments / studies inside LHCb:** [*talk@CHEP2023*]
  - Tested throughput impact of TensorRT inference
    - ↳ Multiple copies of typical sized MLPs seems to **effect throughput in an acceptable way**
    - ↳ Promising avenue of having flexible ML reconstruction and selection at the first trigger level



Usage of ML: current & future challenges

Maintenance & Preservation

- **Key aspect that needs to become standard practice**
  - Online ML applications need to be well structured with **flexible and robust retraining pipelines**
- **CI/CD system**
  - GitLab runners with GPU
  - Pipeline training to make optimal use of resources (e..g. for large models in simulation)
- **MLOps**
  - Necessity for a **powerful tool** allowing **versioning of model, data and hyperparameters**:
    e.g. *MLFlow & Dvc for versioning of Deep Learning datasets* $\Longrightarrow$ *<u>common storage space like EOS</u>?!*
  - **Models organisation** and fast retraining:
    e.g. *Snakemake used for the ultra-fast simulation project*

Summary

- **Multiple on-going developments of ML/DL in LHCb**
  - **Online** (tracks reconstruction, PV finding, trigger) **& Offline** (simulation, performance, analysis)
  - Majority built on **PyTorch** library (few TensorFlow) with **trainings** done using **local resources**
- **Currently the main challenges lie in the integration, inference & maintenance**
  - Working towards **standardization (ONNX)** for ML-model data format
  - Effort to enable a **generic backend inference interface in LHCb production systems**
- **Would benefit from collaboration with / support from experts at IT department**
  - **Large state-of-the-art GPU clusters available for the CERN community**
  - **Multi-GPU batch system** — hyperparameter optimisation
  - **Integrated MLOps tools** — model, data & hyperparameters versioning
  - **Profiling expertise** — efficient integration and use of models used to process data
  - **Maintenance expertise** — keeping updated versions of the different backend packages

Supplementary material

Supplementary material

- **Latest reports on ML applications in LHCb:**

Trigger & Online

- *Applications of Lipschitz neural networks to the Run 3 LHCb trigger* [talk@CHEP2023]
- *Graph Neural Networks for Full Event Interpretation at LHCb* [talk@CHEP2023, arxiv:2304.08610]
- *DNN for finding primary vertices in proton-proton collisions at the LHC* [talk@CHEP2023, arxiv:2309.12417]
- *High-throughput machine learning inference with NVIDIA TensorRT* [talk@CHEP2023]
- *High-throughout GNN track reconstruction at LHCb* [talk@CTD2023]

Simulation

- *The LHCb ultra-fast simulation option, LAMARR* [talk@CHEP2023]
- *From prototypes to large scale detectors: the Gaussino simulation framework* [talk@CHEP2023]

Flavour tagging

- *Fast inclusive flavo(u)r tagging at LHCb* [talk@CHEP2023]