INTEGRATING NOVEL GPU-ACCELERATED MONTE CARLO EVENT GENERATORS INTO THE HIGH ENERGY PHYSICS TOOLCHAIN

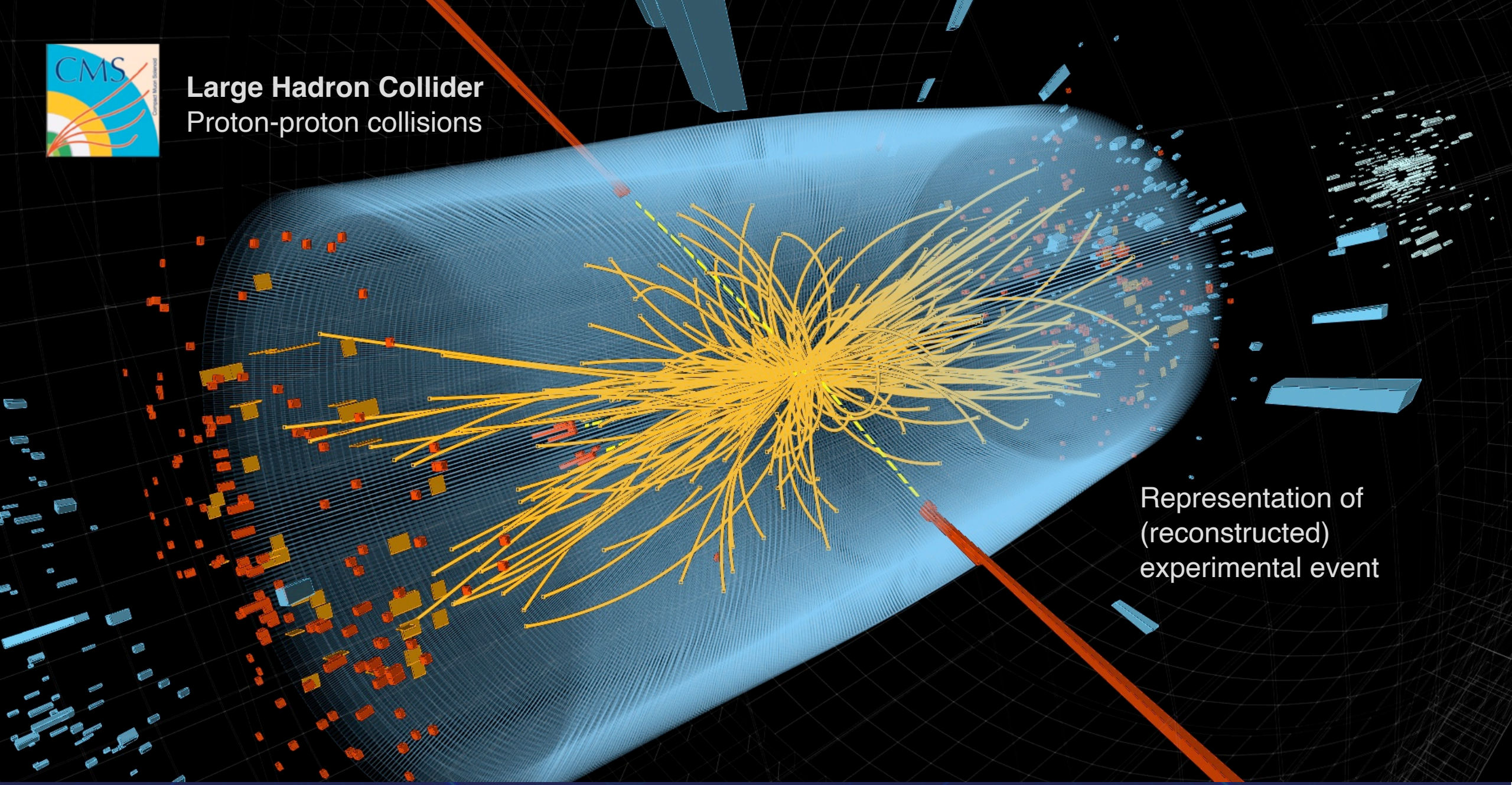
Carla Judith López Zurita

ETH Zürich

Supervisors

- Enrico Bothmann
- Stefan Roiser
- Zenny Wettersten

CERN openlab


CMS
Large Hadron Collider
Proton-proton collisions
Representation of
(reconstructed)
experimental event

Event generation

Expectation value

differential cross section

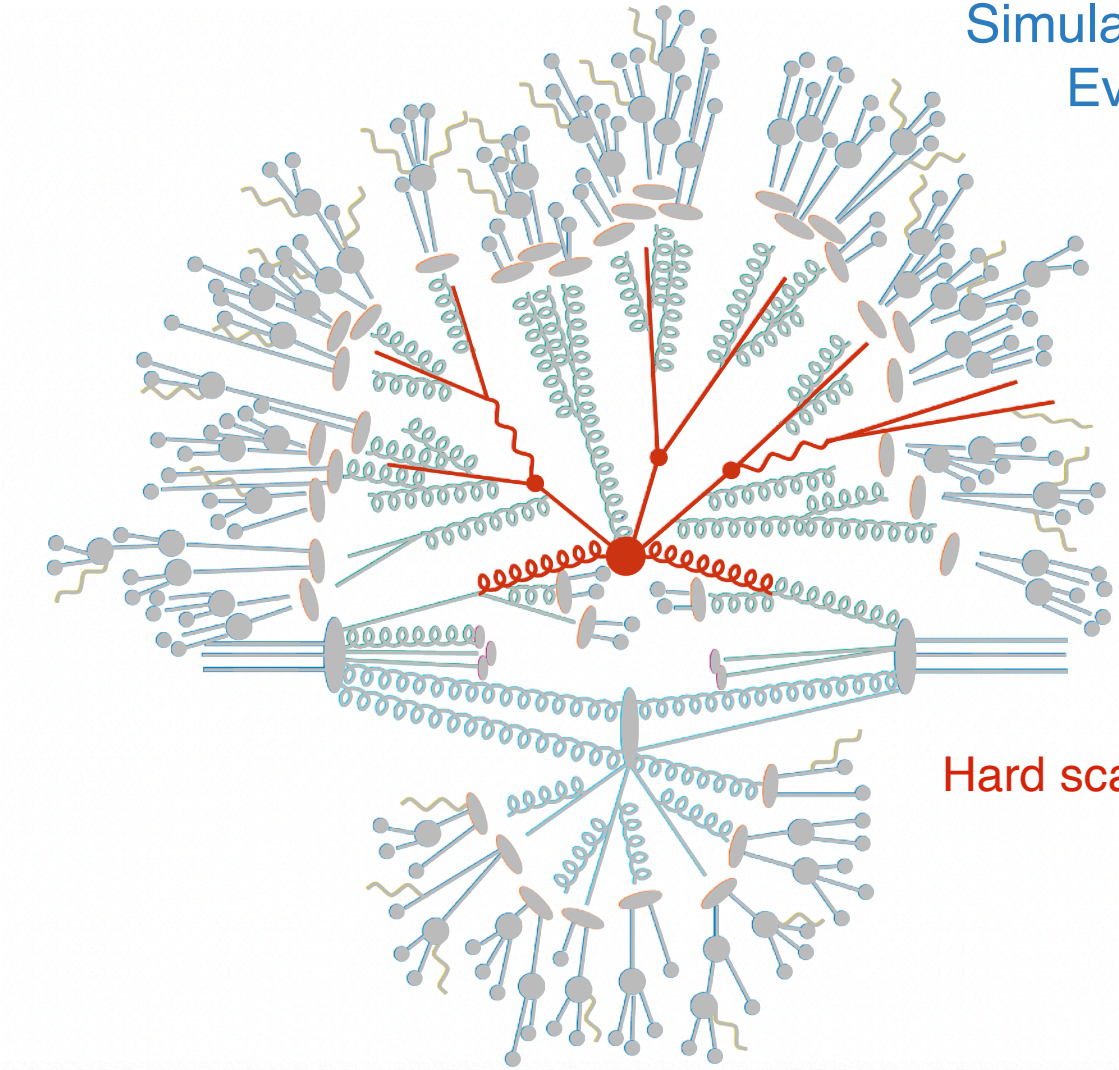
$$\langle O \rangle = \int \mathrm{d}\Phi_n \frac{\mathrm{d}\sigma(A, B \to n \text{ particles})}{\mathrm{d}\Phi_n} O(\Phi_n)$$

phase space

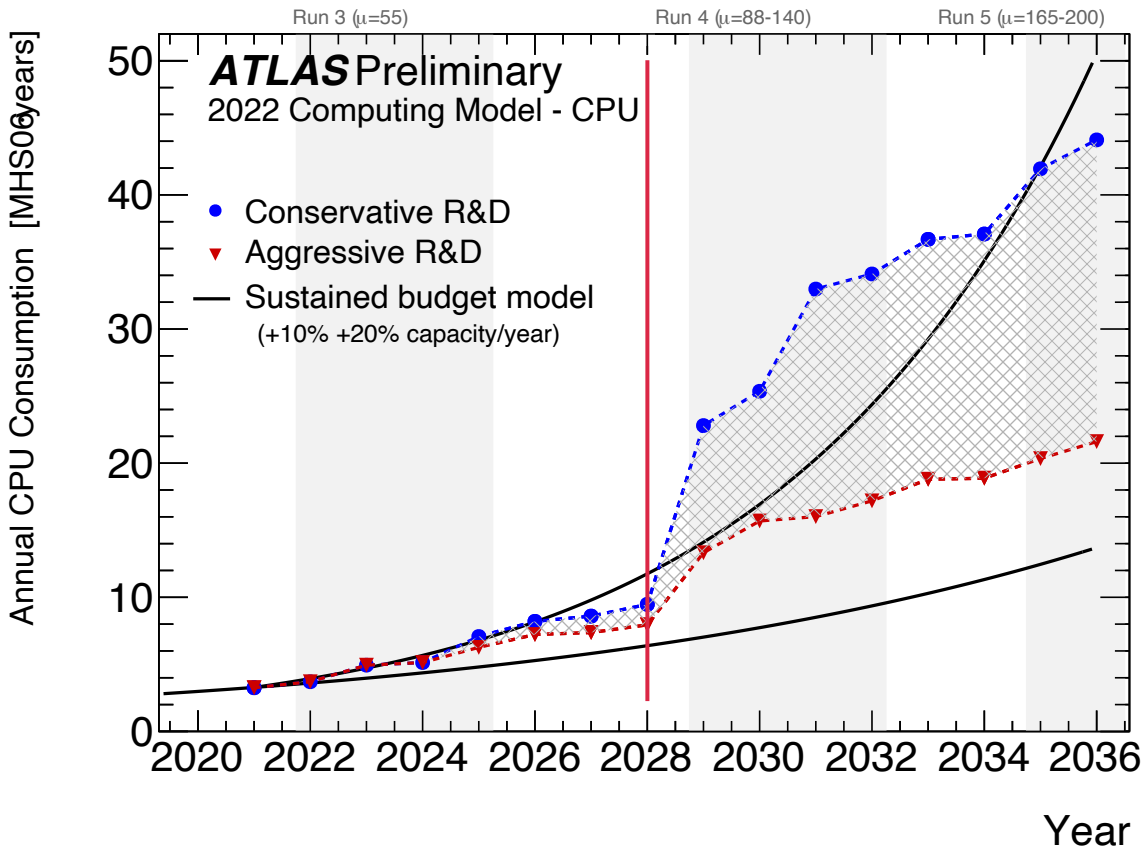
observable

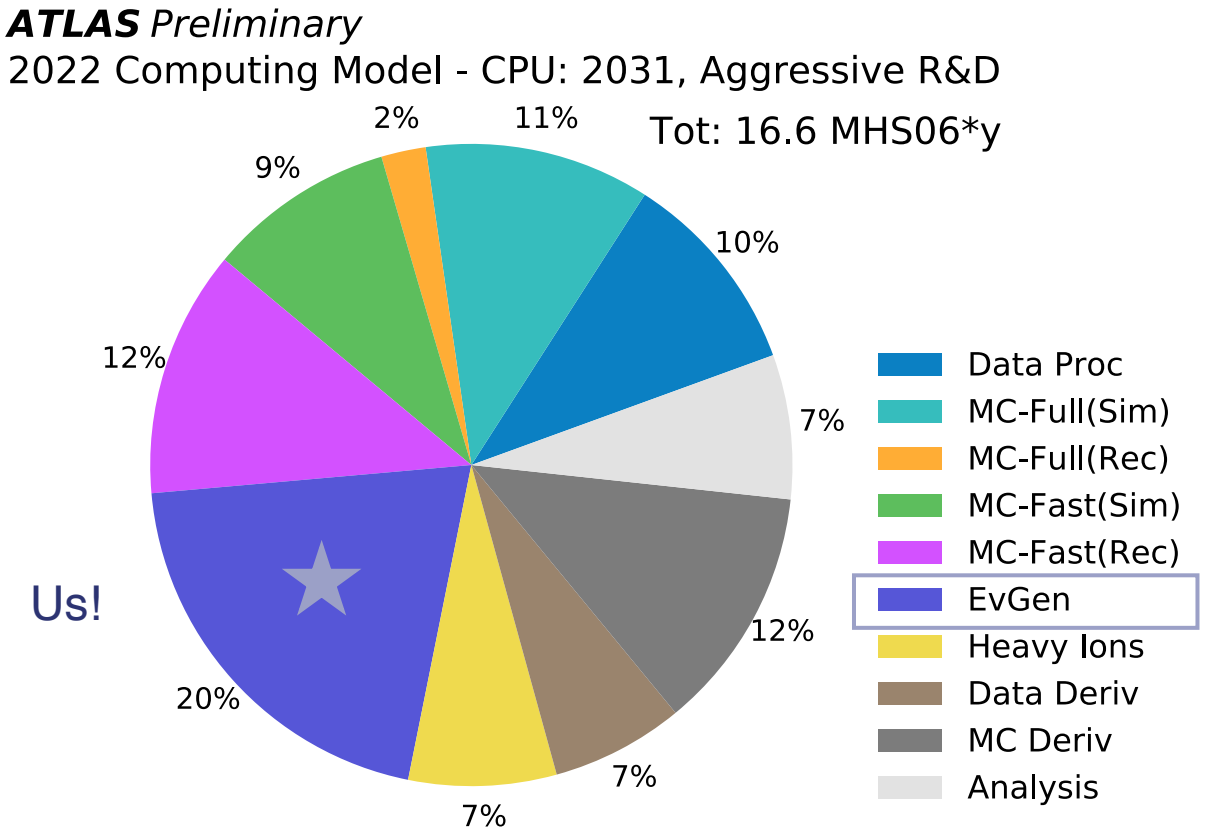
$$\langle g \rangle = \int dx f(x) g(x)$$

Solved using Monte Carlo Integration



High-Lumi LHC





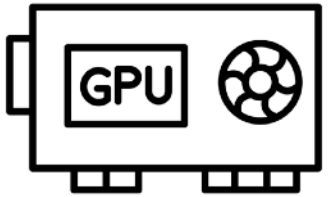
Idea: Put bottleneck on GPU

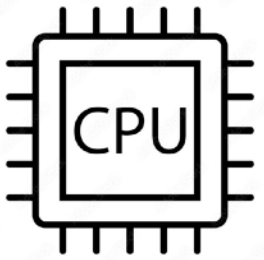
Spice

Introducing: Spice (Simulation of Particle Interactions in Collider Events)

- Chili (Common High-energy *Integration* LIbrary)
- Pepper (Portable Engine for the Production of Parton-level *Event* Records)

kokkos



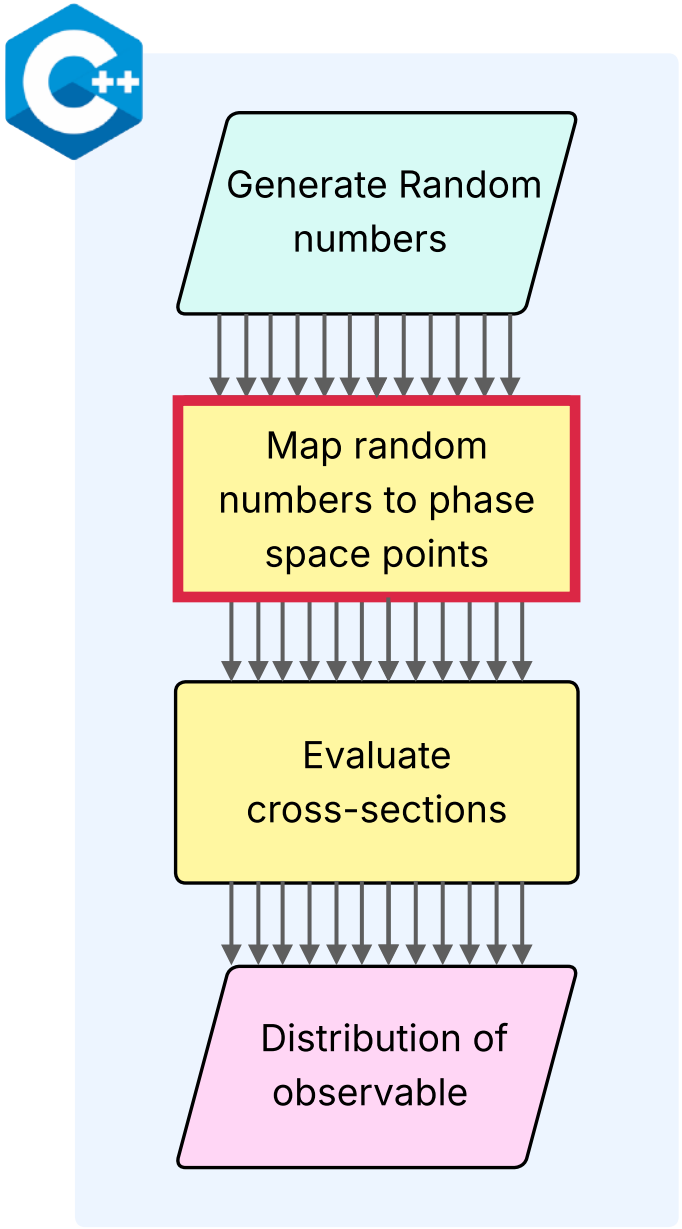


Hardware agnostic

Pepper

Done in batches:

- Multiple events at a time
- Takes advantage of GPU capabilities

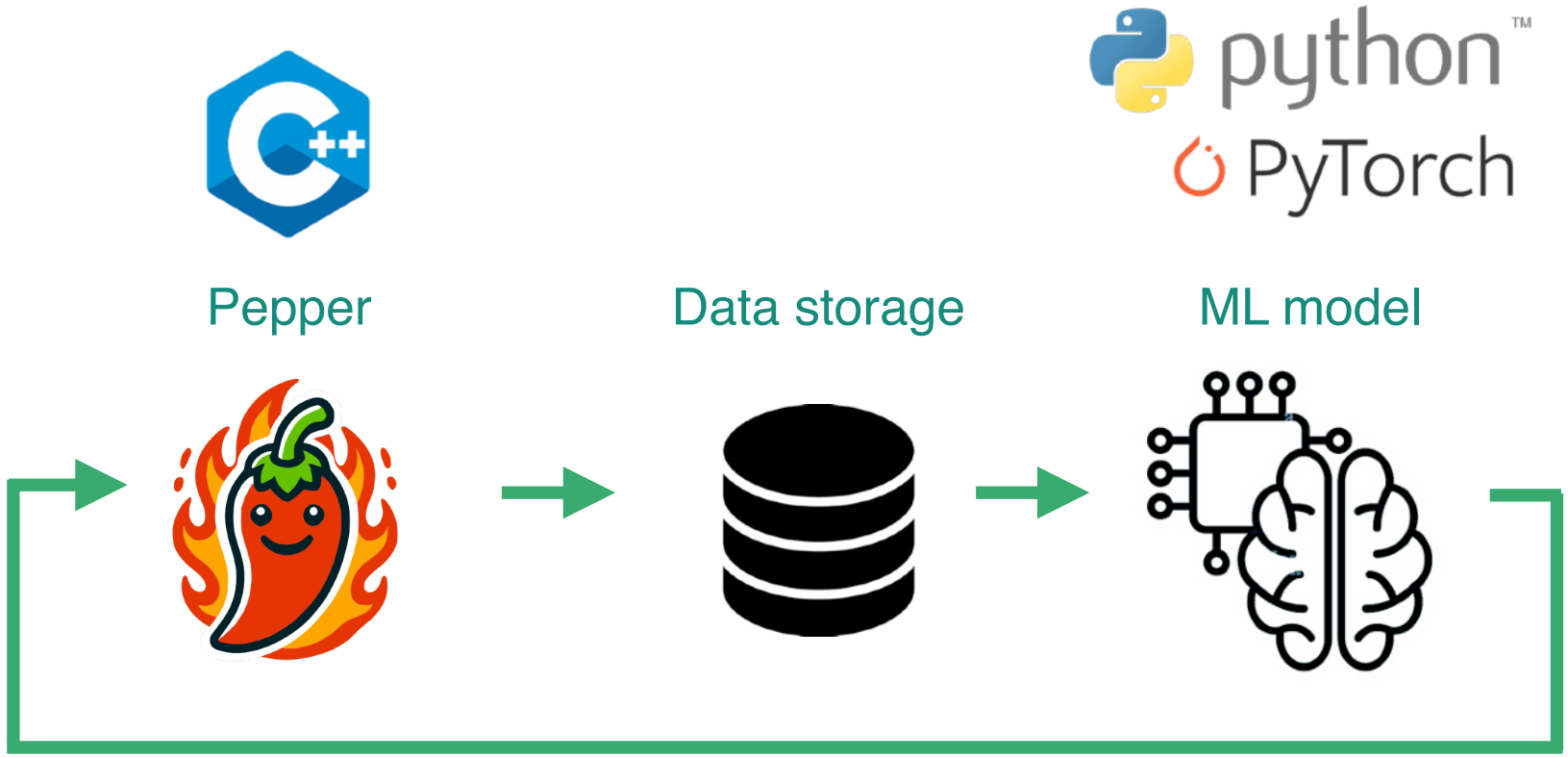


We can improve this step by learning a better distribution

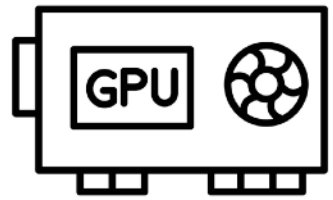
↓

Machine Learning (ML) model!

Pepper + ML



Problem!!



Spice

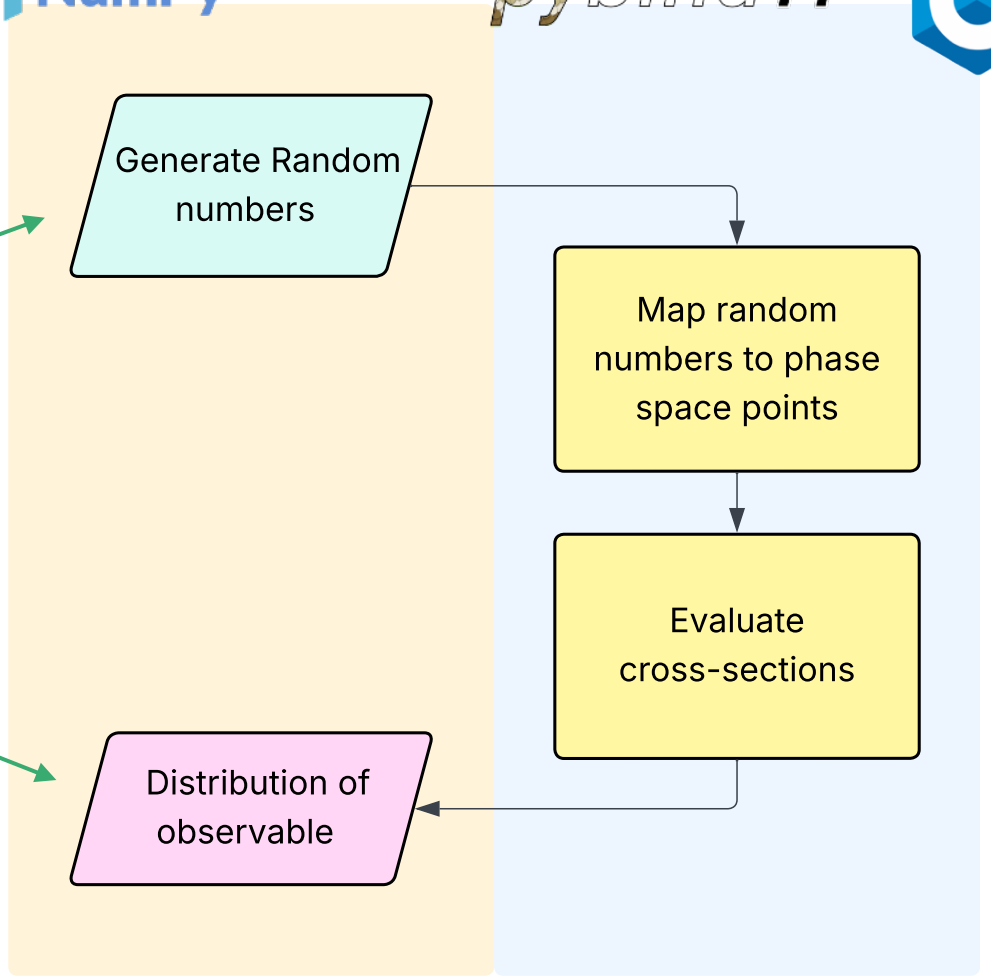
Spice (Simulation of Particle Interactions in Collider Events)

- Chili (Common High-energy *Integration* Library)
- Pepper (Portable Engine for the Production of Parton-level *Event* Records)
  - Pyper (*Python bindings* for Pepper)

Pyper

PyTorch
NumPy

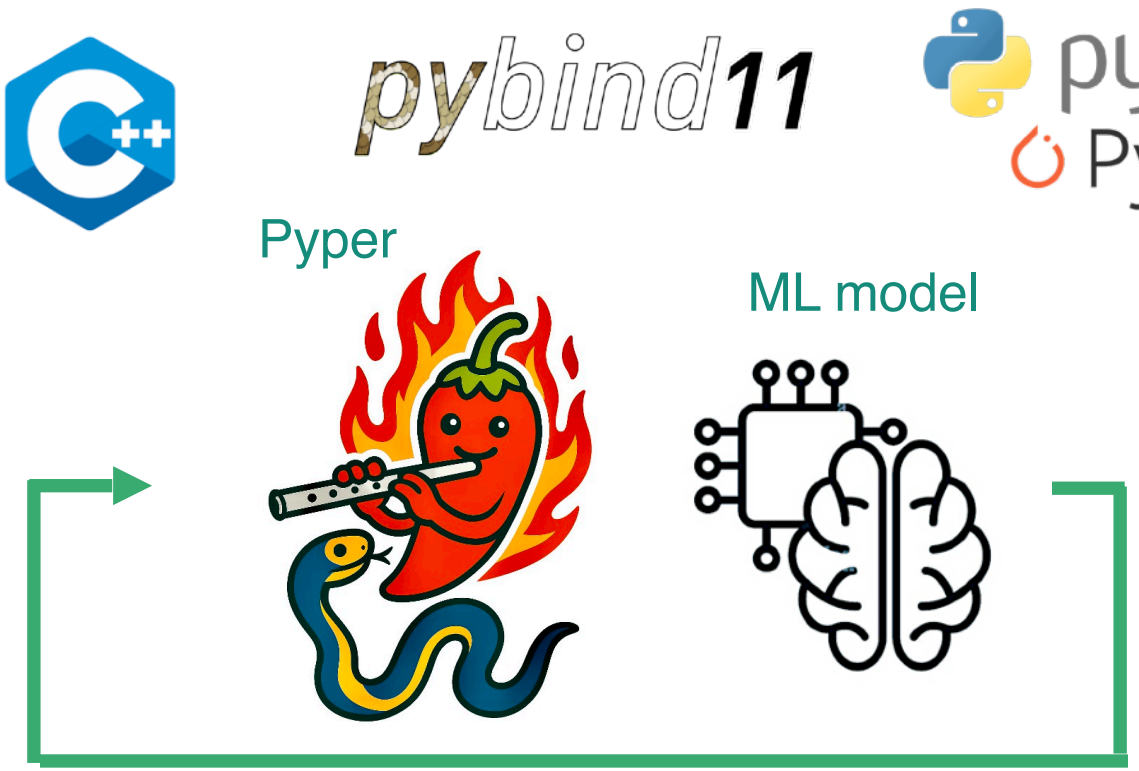
Python exposed
pybind11

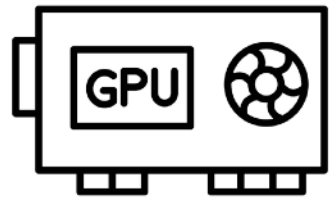


Passing data arrays by **reference**

avoids having to create a **copy / store** in intermediate file

Pepper + ML

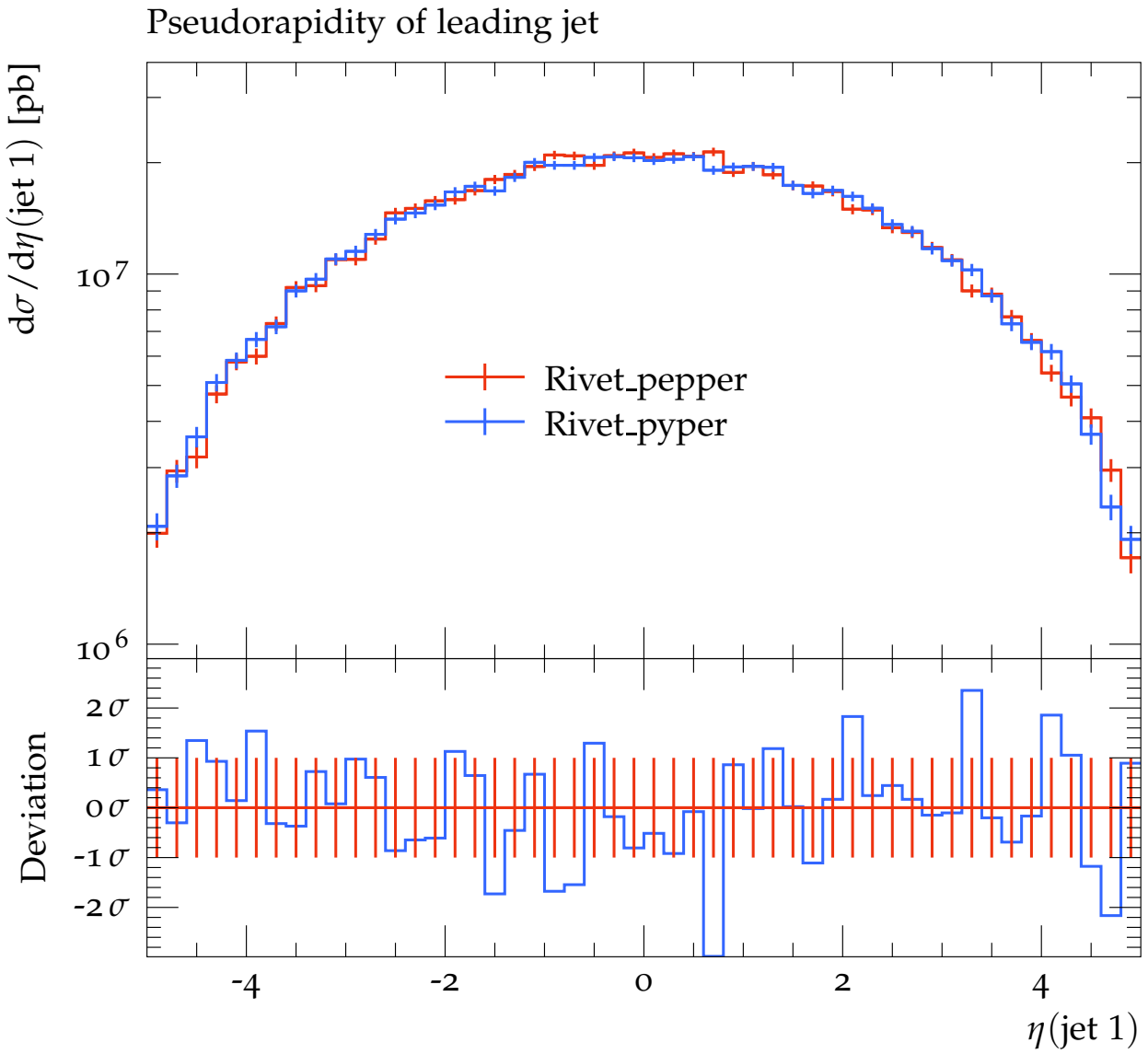




Results

- Process: pp -> 2 jets
- Number of batches: 500
- Batch size: 1000

*__Pseudorapidity__: spatial coordinate describing the angle of a particle relative to the beam axis



Kolmogorov-Smirnov p-value

0.9618 > 0.05

Compares the underlying distribution F(x) of a sample against a given distribution G(x).

Results come from the same distribution.

Future outlook

- Performance and timing evaluation within the full application
- Test machine learning applications, namely normalizing flows
- Integration into higher-order (N)NLO calculations

Thank you!

https://gitlab.com/spice-mc