Calling Pythia8 to handle an interaction during a Geant4 simulation

27th Geant4 Collaboration Meeting

Einar Elén, L.G Sarmiento

September 30, 2022

LUND UNIVERSITY

1. Background & Context

- LHC style experiments *typically* have clear separation between event generation and detector simulation
  - This is less true for other HEP applications
  - Signal process often happens in the middle of the simulation
- Context for our work is Light Dark Matter eXperiment (LDMX) and KAW-LDM project
- Two goals
  - Flexible signal simulation for LDMX using Pythia8 in Geant4
  - Making what we learn useful outside of our particular use-case
- Two reference applications for studying this
  - Carried out in collaboration with colleagues from the Lund Pythia group

1.1. KAW-LDM

- Project aiming at lasting impact on wider Swedish DM landscape
  - LDMX participation
  - DM Simulation aspects
  - Statistical inference & global data interpretation
  - Detector material evaluation for direct detection
- Use LDMX challenges and results to drive research and collaboration between theory and experiment

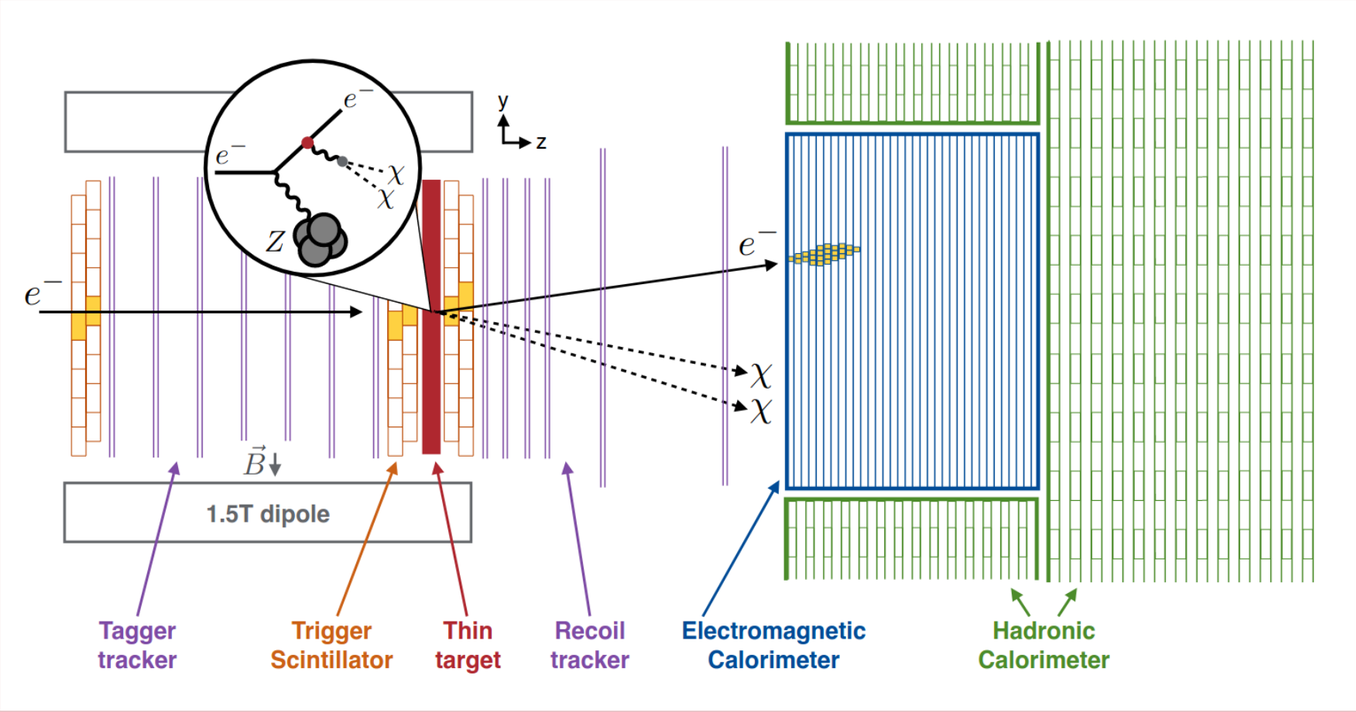
Knut and Alice Wallenberg Foundation

Stockholm University

LUND UNIVERSITY

CHALMERS UNIVERSITY OF TECHNOLOGY

1.2. LDMX



From LDMX Current status/prospects, https://arxiv.org/abs/2203.08192, Fig 1

2. Pythia8 as an primary generator

- Wanted a simple Geant4 component but with full-featured Pythia events
  - Picked a Pythia8-based primary generator action combined with a version of **J. Yarba**'s Py8Decayer example
  - Can also take decays directly from Pythia event
- In principle not different from pre-generating events and reading from disk
  - Similar tools are used by some experiments already
  - Offers a different UX
  - Allowed us to write a large portion of the interface code we need for more complicated projects and discovering pitfalls
- Unit testing was crucial
- Currently relying on system installation of Pythia8, could be relevant to write CMake code to (*optionally*) build

3. Hadronic interactions: Cosmic rays

- Using Pythia *applications* as models for hadronic interactions
- From discussions with Lund Pythia developers where they saw interest in adding Geant4 features to a Pythia application
  - Recent work by **T.Sjöstrand** and **M. Utheim**, including inelastic hadron-nucleus interaction at arbitrary energies for simplified nuclear matter
  - Each interaction is treated as a series of Pythia collisions between hadrons
  - For the physics: See T. Sjöstrand & M. Utheim Eur. Phys. J. C 82, 21 (2022)
- Doesn’t rely on any hard processes in Pythia
  - Only Soft QCD and Low-energy QCD processes
  - Allows for using a single Pythia object for each sub-collision and every event
    - Beam species and energy can be changed without setting up a fresh Pythia object
- However, embedding a full application is more complicated than typical event generation
  - Want to establish a corresponding interface to wrap this
- Extended/Hadr02 for CRMC was extremely useful here

LUND UNIVERSITY

4. Flexible signal simulation

- Hard process simulation for experiments like LDMX
- LDMX has a simulation/reconstruction framework including Geant4 simulation with a model for dark bremmstrahlung in target (LDMX-sw)
  - Signal simulation relies on library of pre-generated events from MadGraph for various electron momenta
  - Works for a given model, but changing models is non-trivial and not feasible for people outside of the LDMX software group
  - On-going work in the KAW-LDM project by **L. Gellersen**, **T. Gray**, **R. Pasechnik**, and **R. Catena** to use Pythia8 to explore further DM models
- Instead, use exact kinematics of the event
  - Requires a Pythia object set up for specific beam type and kinematics
  - Strategy for handling creation and lifetime of such objects will be a crucial challenge
  - As will interactions with multiple threads