IntensityFrontier experiments: Geant4 performance

Hans Wenzel

Use of Geant4 in Iintensity Frontier experiments

- *In common:*

art: event-processing framework for particle physics experiments,

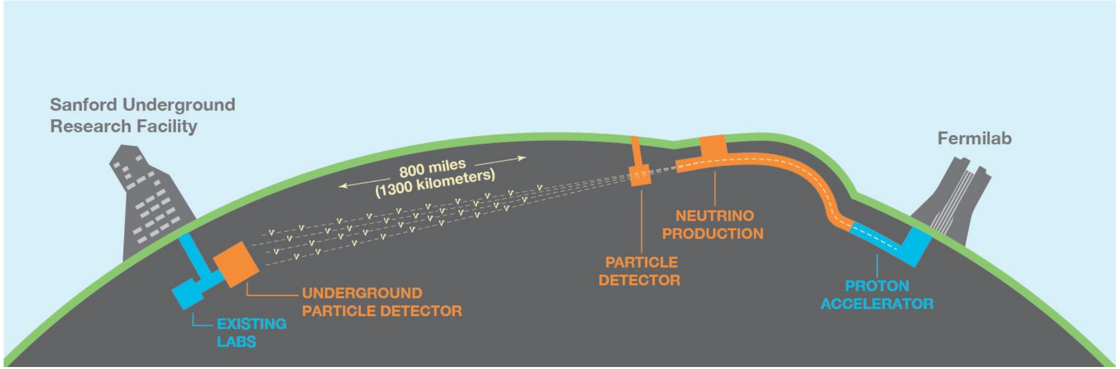
Currently: ups/upd for software distribution, Scientific Linux 7

Future: spack: package manager, alma Linux

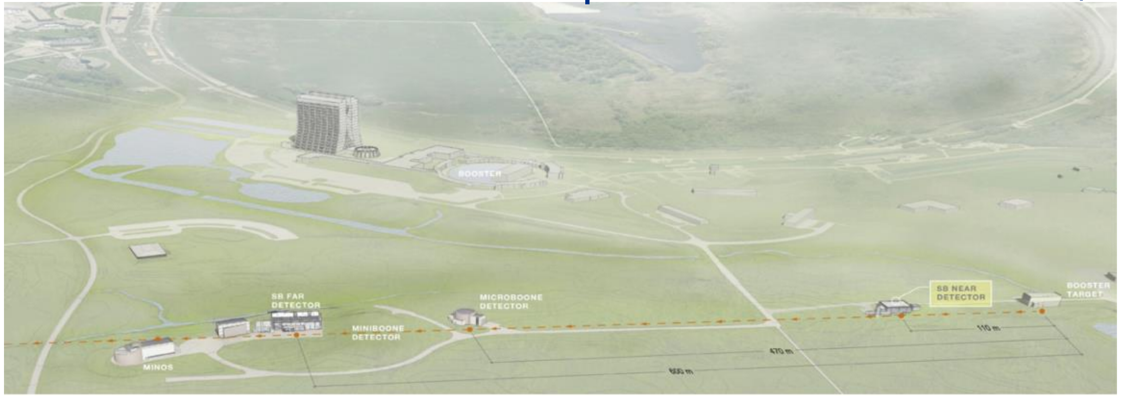
But different way to integrate Geant4

- μ->e:
- g-2: artg4 experiment specific, Geant4 UserActions are implemented as art services for Geant4, can be configured at runtime via fcl files
- Lar TPCs:
  - … all but Dune ND use Larsoft: The Liquid Argon Software (LArSoft) Collaboration:
    - Legacy larg4 Geant4 module
    - Refactored larg4 module
  - The plan is that, if possible, all will migrate to the refactored larg4.

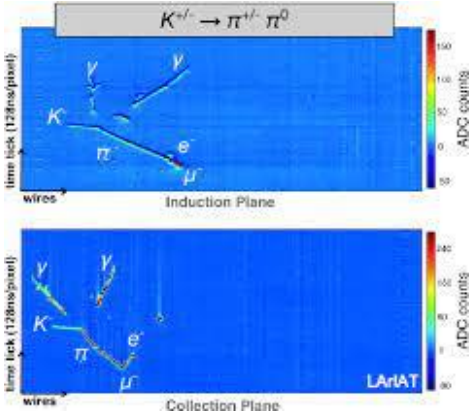
LArTPCs: Long baseline Neutrino experiments : Dune FD/ND

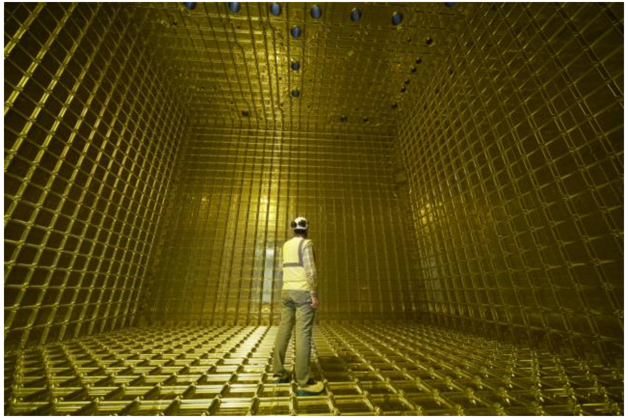

Sanford Underground Research Facility
800 miles (1300 kilometers)
Fermilab
NEUTRINO PRODUCTION
PARTICLE DETECTOR
UNDERGROUND PARTICLE DETECTOR
PROTON ACCELERATOR
EXISTING LABS

Short baseline neutrino experiments: microBoone, Icarus, SBND

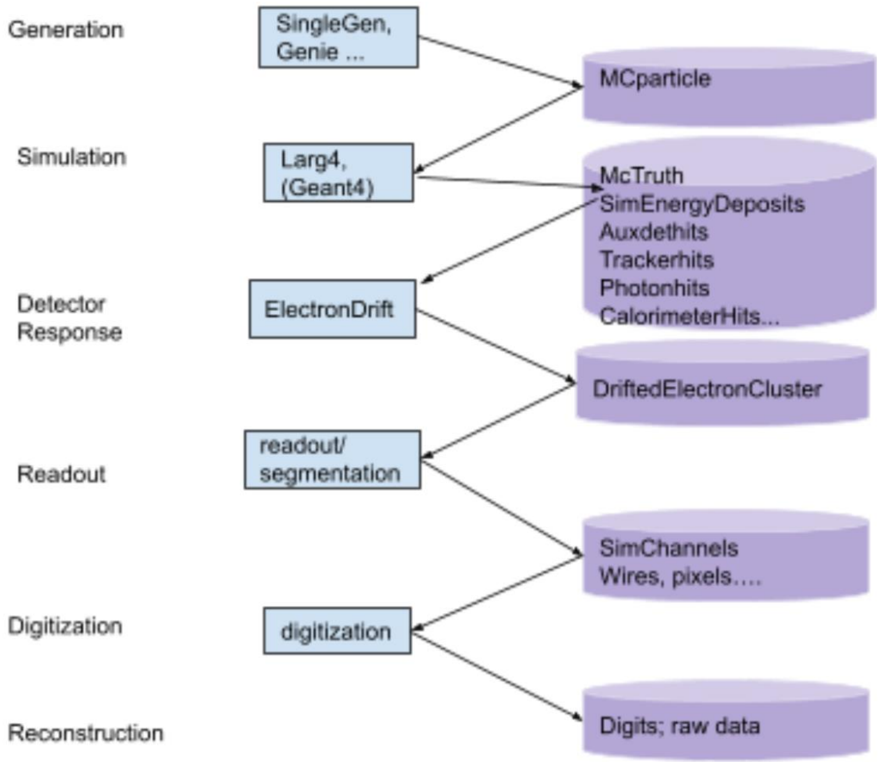


Test beam : protoDune (CERN),Lariat(Fermilab)





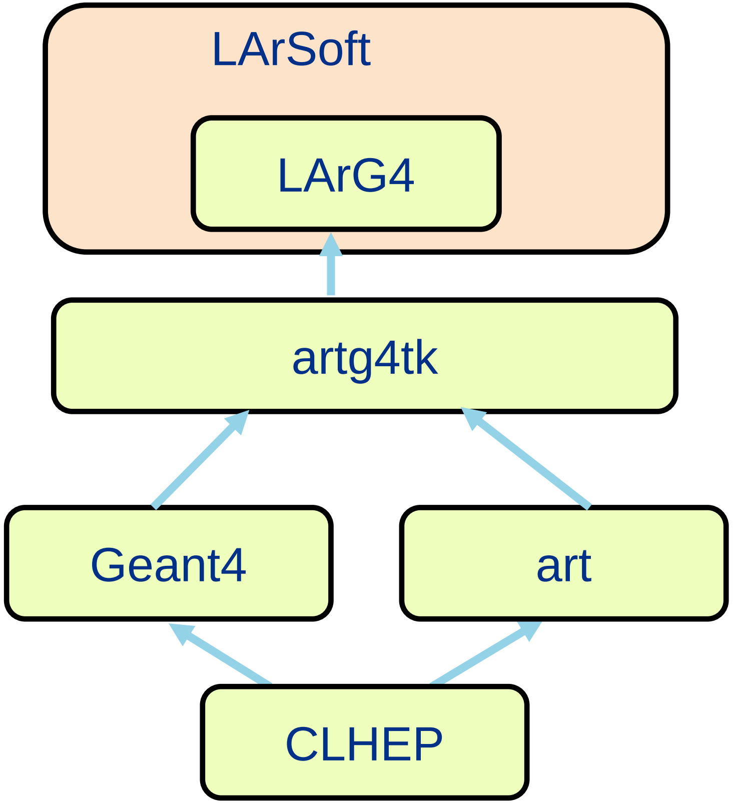
Experiments in test beams at CERN and Fermilab, Besides LArTPC R&D (readout PhotoDetectors…) Measurement of cross sections: total, inelastic, quasi-elastic, pi exchange..



Refactored larg4:

Separate out all functions that are not Geant4 specific, →

- more modular,
- easier to move to new versions of Geant4, (assuming no changes to Geant4 API),
- each of the components can be easily switched
- Easier to study the performance of each component



Artg4tk: derived from artg4 (g-2 specific), Flexible framework to integrate Geant4 and art:

- provides Geant 4 User action as art service
- Uses gdml + extensions (assign sensitive detectors to LV. Visualization attributes, step limits and cuts)
- Uses G4PhysListFactoryAlt to define and configure physics (reference physics list, em option, physics constructor) at runtime via command line option
- Provides library of predefined sensitive detector and associated hit collections (e.g. tracker, (DR)Calorimeter, Photodetector, LArTPC…..).
- Larg4: depends on artg4tk, all Larsoft dependencies are kept here, to preserve artg4tk as general Geant4 frame work.

Optical photons in liquid Argon

Liquid Argon Is a very efficient scintillator: (50000 VUV photons/Mev deposited) e.g. in a 2 GeV electron shower about ~7x10^7 VUV scintillation photons are produced/event in the absence of an electric field.

Using Geant4 (11.1.p02) to simulate photon generation and propagation on using a single core on an Intel® Core i9-10900k@ 3.7Ghz takes :

> 5 minutes minutes/event

(Compared to 0.034 seconds/event without optical photon simulation) → LArTPC-Experiments use look up tables and parameterizations instead of full simulation for photon response.

In the presence of an electric field there is competition between Ionization and Scintillation since electrons and Ions can be separated and the drift of electrons along the electric field is read out by wires in a stereo configuration or pixels. e.g. with an electric field of 500 V/cm the number of optical photons is reduced by a factor of two.

Geant4 performance

ProtoDune SP:

Single thread: G4 "stage 1": 25.6 seconds/event for the entire job. 15.5 seconds per event in larg4. almost all the rest is in the RootOutput module, probably compressing MCParticle trajectories. VMPeak: 5.1 GB, a large amount of the memory used here is to store MCParticle trajectories and energy deposits. Bottlenecks outside Geant4 (e.g. wircell. Geant4 represents only 1.3%   CPU in a ProtoDUNE-SP sim+reco workflow)

Dune:

Single thread

Ikarus:

single thread: Gent4 stage: 21 sec/event, Rootoutput 9sec/event. They attribute Peak resident size : 7 GB to detector simulation.

SBND:

single thread: 15 sec/event, Peak resident size: 6.2 GB.

MicroBooNE :

single thread: Geant4 resource usage is a sub-leading concern for them compared to detector electronics simulation. Performance is sufficient for current MicroBooNE needs.

Geant4 performance (cont.)

LArTPCs in summary Geant4 CPU resource usage is a sub-leading concern for them compared to detector electronics simulation and reconstruction stage. Memory is a concern.

μ2e:

most common configuration is 2 threads with about 3.4 GB of total memory.
less with 1 thread and ~3 GB of memory. Geant4 stage CPU fractions of a second to a few seconds.

Possible optimization

- Geometry avoid duplication of materials, logical volumes
- Currently none of the LArTPC experiments make use of multithreading
- No killing of particles that can’t produce a hit. E.g. simulation might involve lots of rock.
- Optimize the step limit to match the readout pitch.
- Optimize data layout avoid extensive use of compression/uncompression (CPU/Memory)