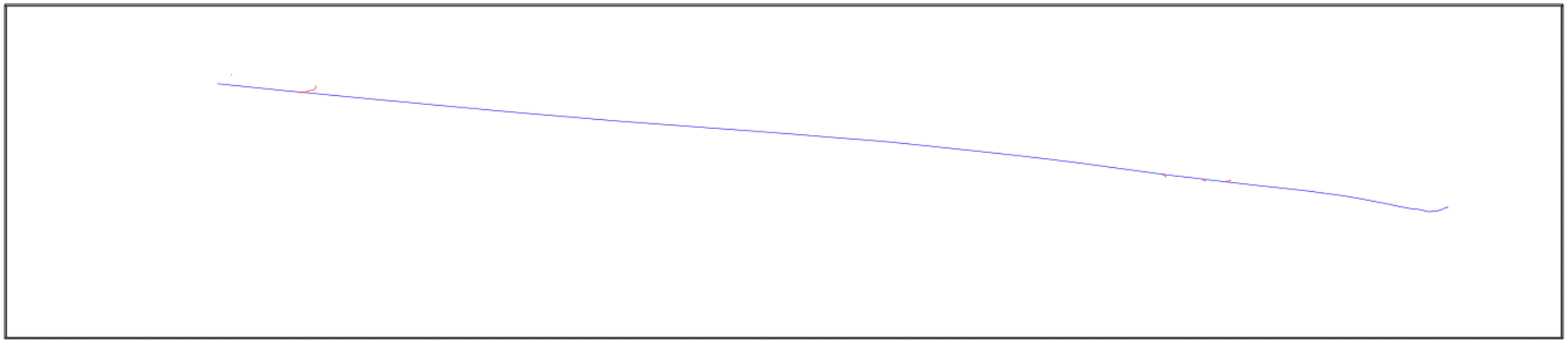
Voxels

- LArSoft simulation chain:
  - Flux (beam)
  - Event generation (GENIE, CRY, etc.)
  - Detector simulation ← particles, voxels
  - Electron clusters
  - Wire signals

Voxels: the definition

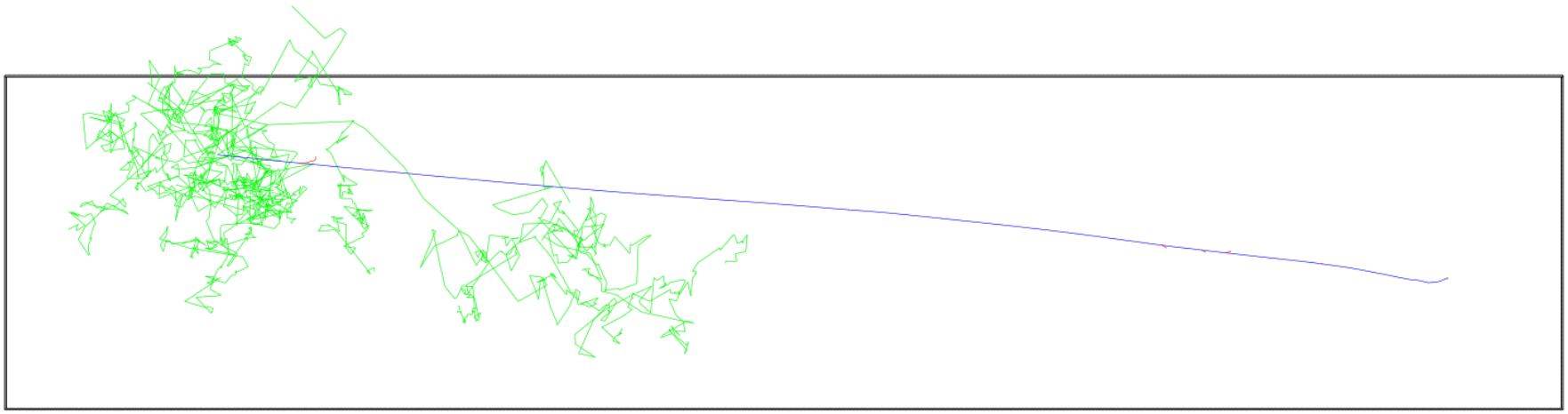
- Basic ideas:
  - The detector simulation is written in Geant4.
  - Divide the active LAr region into small three-dimensional volumes called “voxels” (volume elements, in analogy with “pixels”).
    - Actually, since we need to have time divisions as well, perhaps we should call them “tesserels”… but I won’t!
  - Sum the total energy deposited by charged particles within each voxel.
  - Write out the voxels as part of the event.
  - Digitization (conversion of voxels to simulated readout) will be done post-Geant4.

Voxels: why we use them



Geant4 event display (**not** the LArSoft event display!)

We'd like to think of events and tracks in the LAr TPC as simple lines



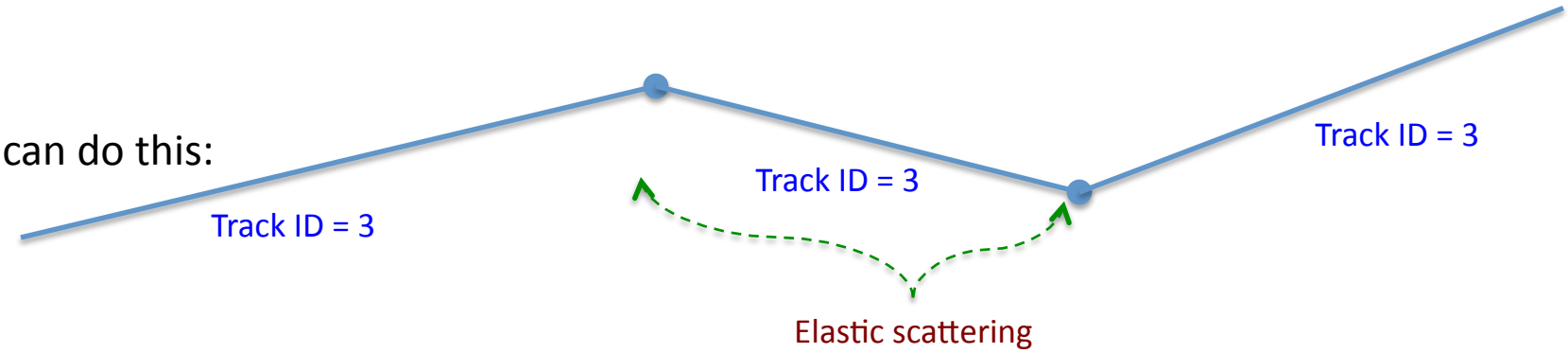
Geant4 event display (**not** the LArSoft event display!)

But that's not how Geant4 sees them. Even relatively low-energy events are modeled in G4 by hundreds or thousands of even lower-energy particles.

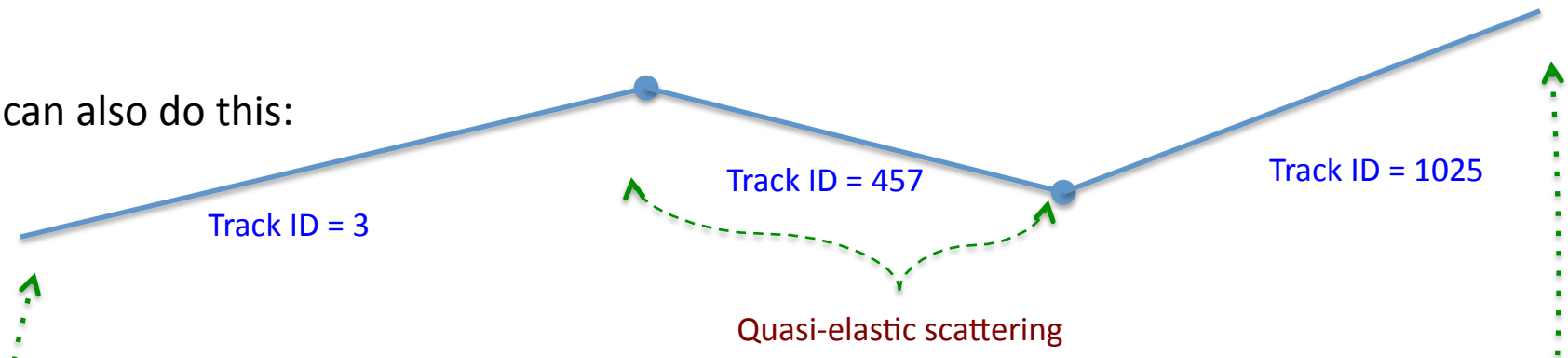
(I'm cheating a little here; we'll see better examples when I show the Truth View in the event-display talk.)

If we only wrote particle truth information from Geant4, we'd have difficulties.

Geant4 can do this:



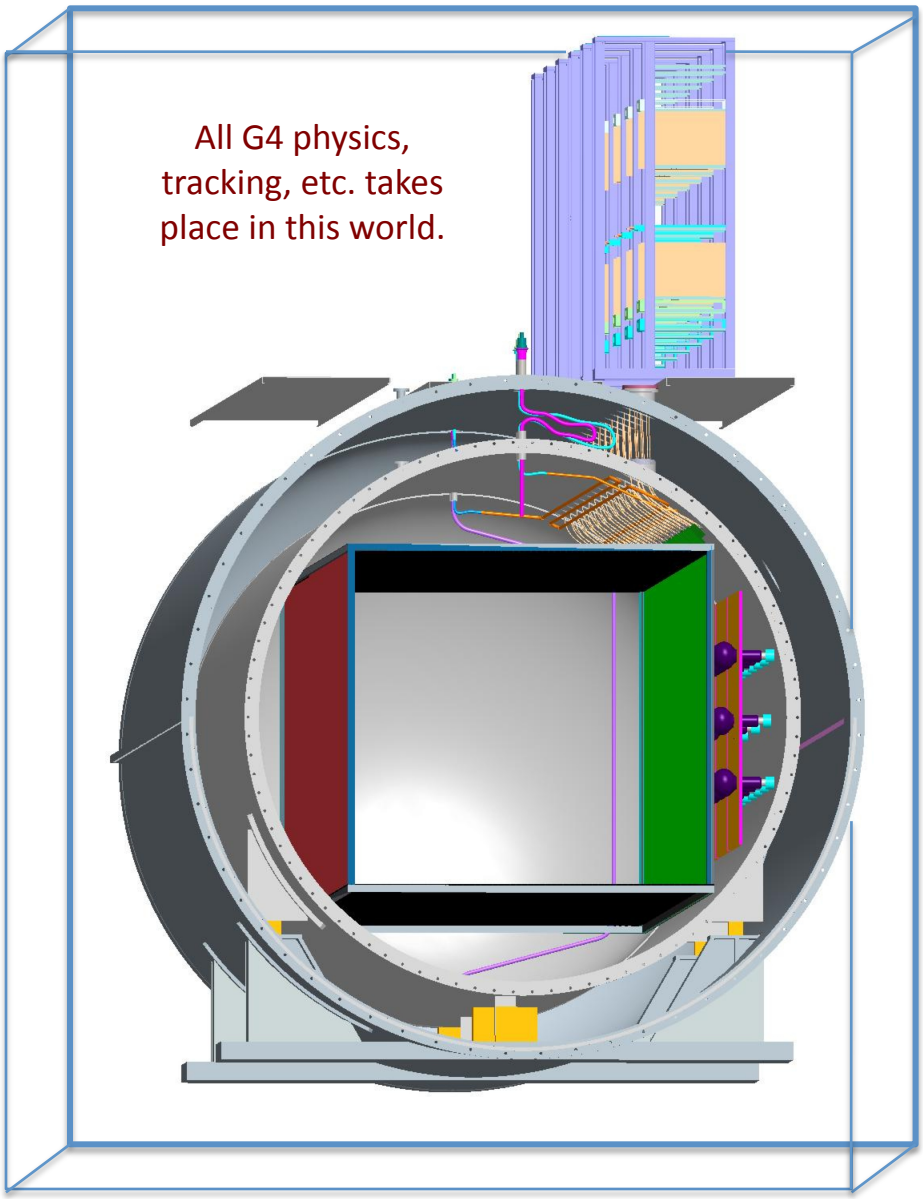
Geant4 can also do this:



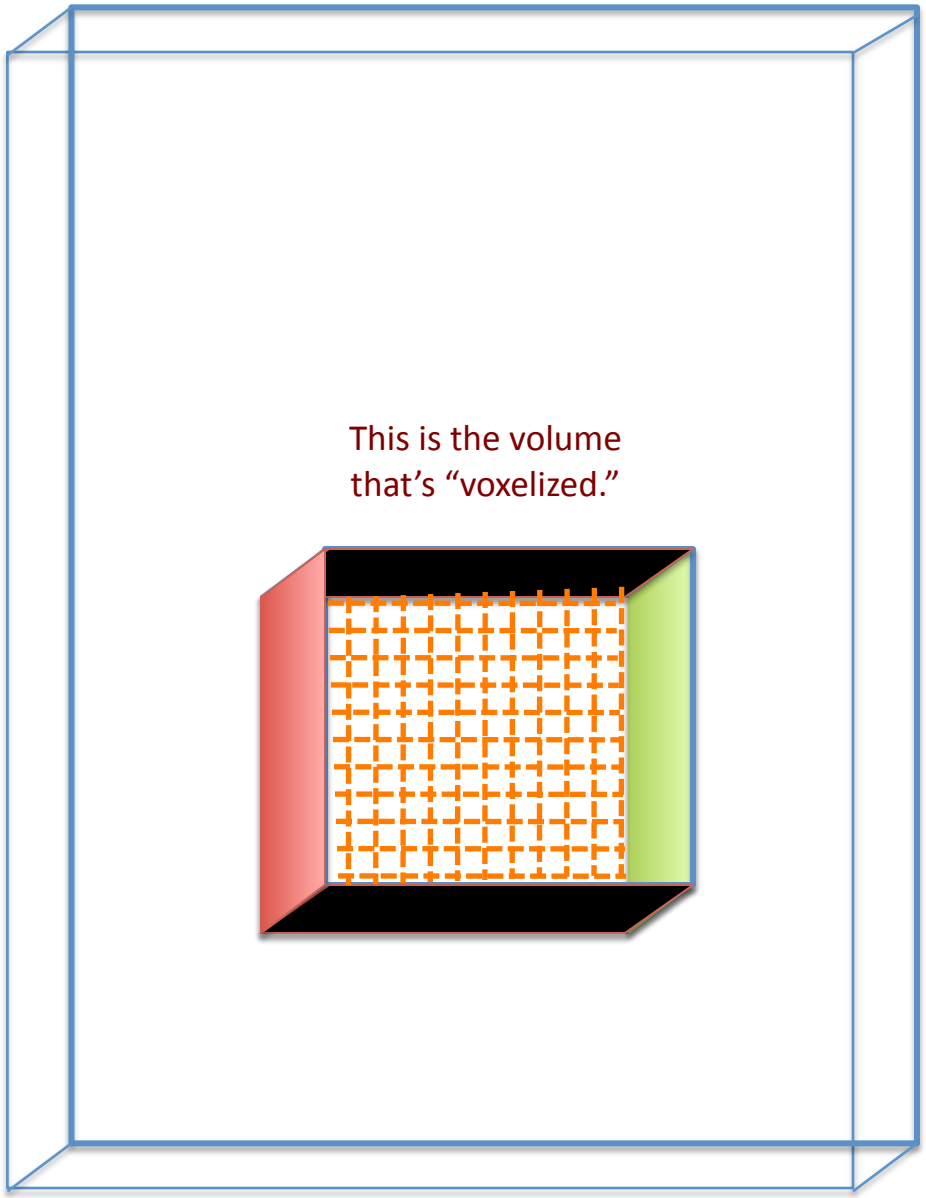
Either way, the energy loss per unit length would not be quite the same here versus here

Simplest answer: let G4 determine the energy deposited in each small voxel.

"Mass world" – complete geometry



"Parallel world" – readout geometry



Voxel Parameters

- Defined in LArG4/SimulationParameters.xml
  - The values in this file are **guesses** on my part. Some work should be done to refine them.
- Four size parameters: the voxel sizes in x, y, z, and t.
  - "t" doesn't have anything to do with the drift time. It's the size of the time window for the event. I anticipate its use in overlay studies.
- Four offset parameters: voxel offsets in x, y, z, and t.
  - These parameters set the origin of the voxel co-ordinate system w.r.t. the origin of the detector co-ordinate system.
  - Uses
    - If you want (0,0,0) to be in the middle of voxel, rather than its edge.
    - If you want the edges of the voxels to be aligned with a particular structure within the LAr TPC.
- Energy threshold
  - If a voxel contains less than this threshold, don't write it out.

Developer notes: Relevant packages in LArSoft

- `SimulationBase` (namespace simb::, in nutools)
  - Data classes that are input to the G4 simulation
- `Simulation` (namespace sim::)
  - This package defines data structures that are output from the LArSoft simulations, not algorithms
  - Voxel Classes
  - Detector Simulation Particle Truth Classes
- `LArG4` (namespace LArG4::)
  - The Geant4 implementation of the detector simulation
  - This package defines algorithms, not data structures.

If you're still confused about the difference between particle tracks and voxels, wait until my next talk. Hopefully the pictures will make it clear.