FGT Simulation & Reconstruction Group

Interested persons		Locations
Tyler Alion	USC/FNAL	
Xinchun Tian	USC	
Tom Junk	FNAL	
Soumya Das	FNAL	
Nitali Dash	HRI (India)	
Anwindar Kaur	Panjab (India)	
……more……		

Listed are just the people who I have discussed actual projects with

22 October 2015

Repository Page: https://cdcvs.fnal.gov/redmine/projects/dunefgt/repository

Meeting Category Page: https://indico.fnal.gov/categoryDisplay.py?categId=515

This Talk

- Current FGT Simulation

- Bare minimum to be done for a full run-through
  - — By Jan 2016 Collaboration Meeting
  - — Putting machinery in place

- What needs to be done before the initial NDTF report in August 2016
  - — Putting physics in place

Current

- **Geometry**
  - — STT & ECAL gdml (seperate)
  - — Beginnings of unified geometry generation
  - — C++ interface following closely behind, test functions

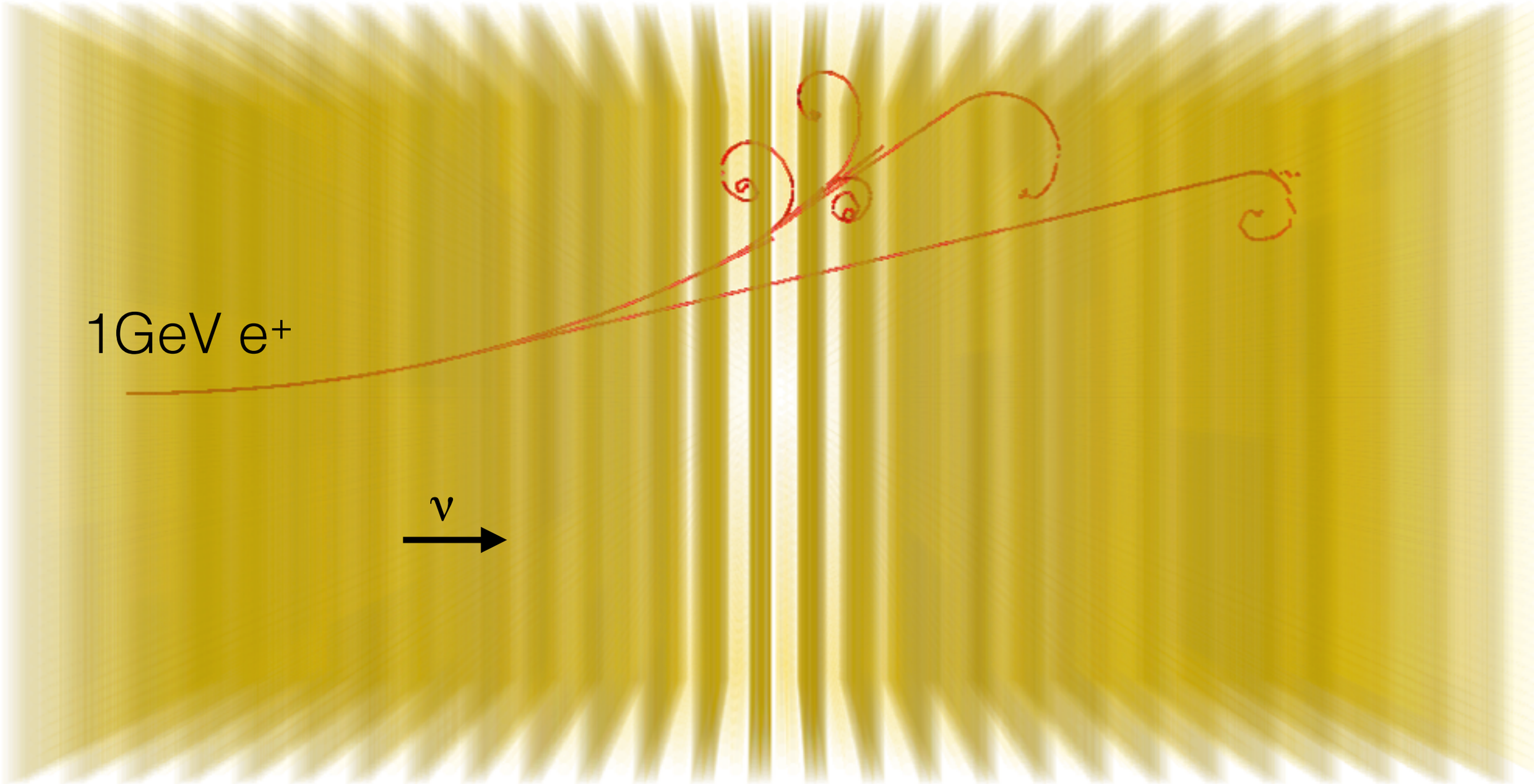
- **Event Generators** — Single Particle, GENIE, etc.

- **Geant 4** — STT (parameterized drift) & ECAL

- **Simulated Data Products**
  - — NuTools products (MCFlux, MCTruth, MCParticles…)
  - — SimTube
  - — SimScintBar
  - — Depositions for each G4 step (xyz, energy)

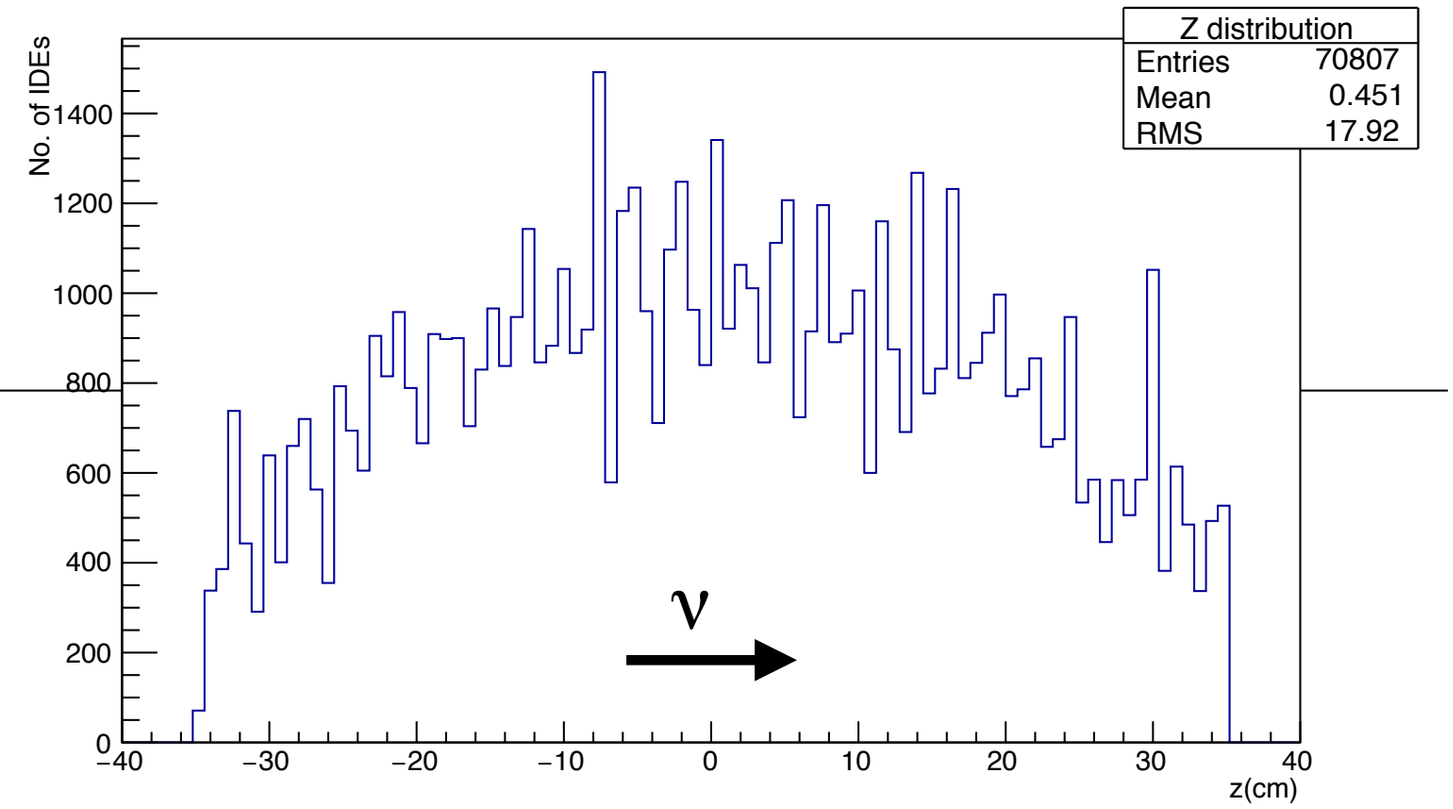
These are what is used for cheater modules.

G4 deposition seen by the STT



We don’t expect so many interactions in a tracker with an average density of 0.1g/cm^3; I show this to emphasize that there are improvements to be made

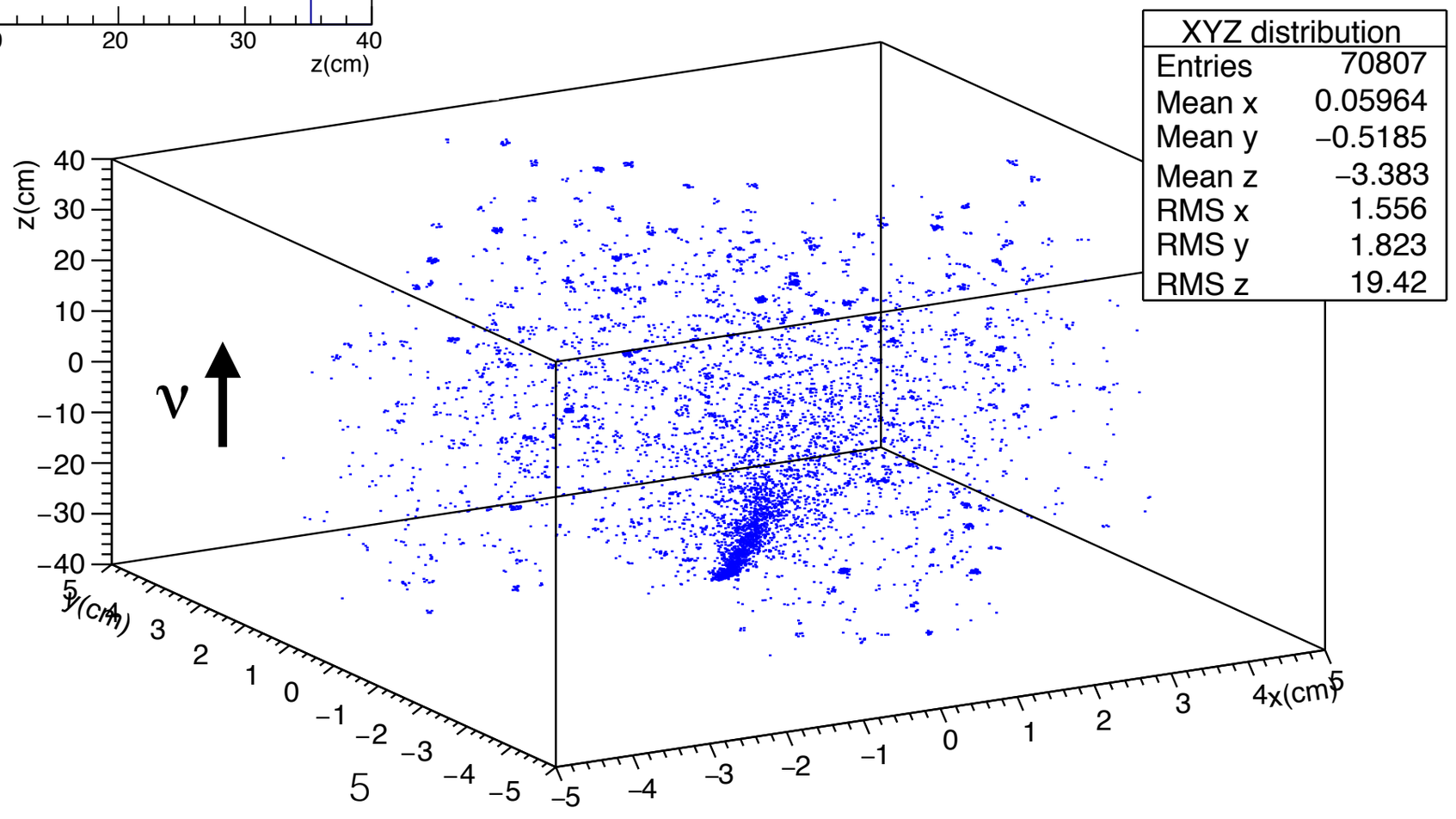
G4 deposition seen by the ECAL



There are 60 layers in the forward ECAL. Binning is not perfect, but you can see slabs of activity and non activity to the left

100 5GeV γ

Directly into the forward ECAL



By: Soumya Das

Bare Minimum Run Through

Simulation: ToDo

- Unify geometry description and interface
  - — Use Bret Viren's Github repo (GGD) for description
  - — Generate GENIE geometry scan xml (>1hr)

- MuonID Geant4 simulation

- Mock digitization/calibration
  - — Assume perfect deconvolution
  - — Energy straight to ADC without noise / impulse response

- Final Result: Raw data object for each readout in each of the three subsystems

Bare Minimum Run Through

Reconstruction: start in STT

This looks like a lot of work, but everything will be cheated at first

- Hit Finder —> Hit (channel, time, pulse shape/height)
- Match Hits —> Some 3d object associated with hit
  — Straw tubes have readouts on both end, registering activity two times but with a small time shift, which can be used to localize activity longitudinally along tube
- Cluster Hits —> Track (trajectory points, assoc. hits, momentum)
  — Much different than LAr, maybe combine with “Match Hits”
- Particle Finder —> Particle (pdg, momentum, energy, assoc. track)
  — Start by filling in momentum measurement in B-field
- Vertex Finder —> Vertex (xyz, daughter tracks)
- Event Finder —> Event (vertex, interaction, energy, y)
  — Start by filling in vertex
- Project Tracks to ECAL and MuonID……

Blue: write a cheater art module
Red: write a product to event

Bare Minimum Run Through

Reconstruction: Look to ECAL & MuonID

- Hit Finder —> Hit (channel, time, pulse shape/height)

- Match ECAL Hits —> Some 3d object associated with hit
  — Scintillator bars have readouts on both end, registering activity two times with different attenuations, which can be used to localize activity longitudinally along tube. Perhaps mix this into “Cluster Hits”

- Cluster Hits
  —> ECAL Shower (total energy, momentum, assoc. hits?, time)
  —> MuonID Track (momentum, assoc. hits)

- Particle Finder —> Particle (pdg, momentum, energy, assoc. ECAL)
  — Match to Particle with Track and continue filling in energy, pdg
  — If no Track, new Particle

- Event Finder —> Event (vertex, interaction, energy, y)
  — Use PID to fill in interaction, energy, y…

Blue: write a cheater art module
Red: write a product to event

FGT Particle Reconstruction

Particle	Kinematics	PID
e	momentum in B, ECAL	Transition Radiation, ECAL
π	momentum in B, ECAL	ECAL shower
μ	momentum in B	lack of ECAL shower, MuonID
π	momentum in B	lack of Transition Radation
p	momentum in B	dEdx, range, lack of ECAL
n	late / displaced deposition	
K	momentum in B	lack of ECAL shower
Λ	decays	decays

Generally: Particles curve in the STT B-field, revealing their momentum to high a precision. These tracks are projected into the ECAL to select $e^{\pm}/\pi^0/\gamma$ from hadrons, and ECAL MIP hits are projected into the MuonID to select $\mu^{\pm}$ from hadrons.

Immediate Task List

- **Geometry**
  - — Write GGD files for World, Hall, MuonID, ECAL, STT
  - — Geometry C++ Interface for MuonID
- **Geant 4**
  - — MuonID simulation
  - — Limit step size (.2mm) in Straw Tubes
  - — Implement Transition Radiation
- **Detector Electronics**
  - — mock up digitization, leaving blank functions for the real deal
  - — Parameterize difference between waveforms on opposing readouts
- **Event Display**
  - — Currently just using histograms
- **Reconstruction**
  - — Consult Reco papers and experts from other STT/ECAL/MuonID systems
  - — Write data products, cheaters

Within 12 Months

- **Geometry**
  — Fill in all significant volumes and densities
- **Geant 4**
  — Study proper step size in Straw Tubes
  — Check G4 Transition Radiation against standalone simulations
  — Sets of verification plots for each subsystem
- **Detector Electronics**
  — Parameterize impulse response and study requirements based off of physics requirements
  — Borrow response from model detectors (ATLAS TRT, T2K ECAL?)
  — Add Noise
  — Write signal service to convolute/deconvolute signal
- **Event Display**
  — 3D and 2D displays showing all data types in all subsystems
- **Reconstruction**
  — Assess what should not be cheated for sake of LAr/HPTPC comparison
  — Write those algorithms if possible
  — otherwise parameterize the cheated at all appropriate points

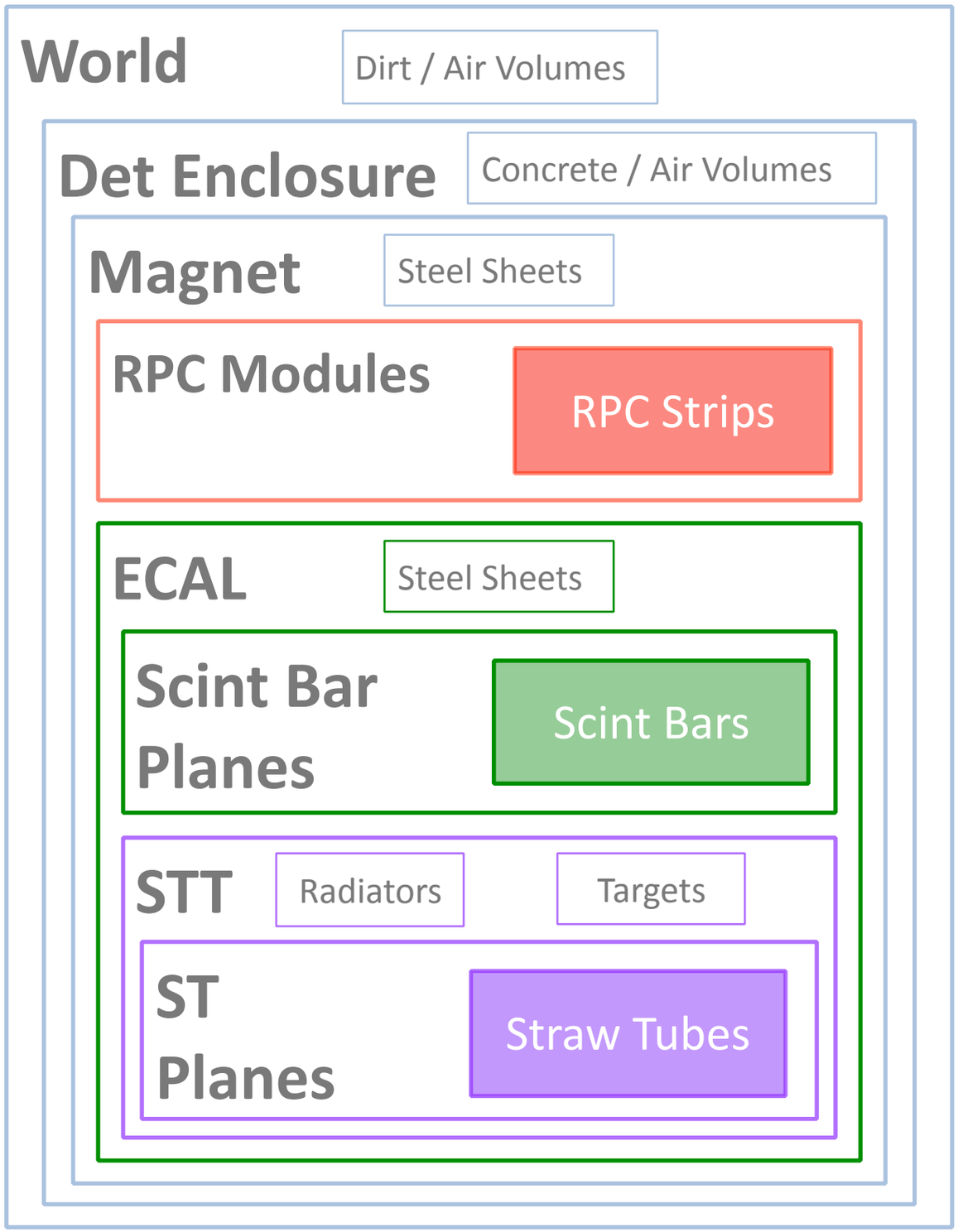
Questions?

Suggestions??

Volunteers???

Geometry

Volume Hierarchy:



Each bolded volume will have its own GGD file dedicated to building all of its sub volumes. This partition eases collaboration and multiple versions

Each colored volume will have its own array of C++ Geometry and Simulated objects

The design easily allows multiple people to be working on this at the same time

UTILITY PACKAGE
(ART SERVICE MODULE)

Utilities
- DetectorProperties
- TimeService
- other

Geometry

NuTools

Simulation Chain:
(Package of ART producer modules)

EventGenerator → **NDG4** --> **DetSim**

Products living in NuTools/ SimulationBase or HiSoft/Simulation

MCTruth

SimTube

SimScintBar

MCParticle

RawDigit

Input to Reconstruction

DATA PRODUCTS

NuTools G4Base Package

Mandatory G4 User Actions

ConvertMCTruthToG4

G4VUserPrimaryGeneratorAction

"MCTruth" is the output of our generators

DetectorConstruction

G4VUserDetectorConstruction

— Parse GDML with G4GDMLParser
— Set Detector Field (B field)

G4VUserPhysicsList

— Default to QGSP_BERT, can select from others in G4PhysListFactory
— Registers any other G4VPhysicsConstructors
— Subject to change with G4 v10

G4Helper

Much of the code you would write in a standalone simulation

- Instantiates Detector Construction
- Registers any parallel worlds
- Sets physics lists, potentially with extra physics constructors
- Resets ConvertMCTruthToG4 and applies command "/run/beamOn 1" every time *G4Run* is called

FGTG4_module

ART Producer Module

- Instantiates G4Helper
- Passes GDML to Detector Construction
- Passes parallel worlds
- Tells helper to initialize physics
- Manages any other additional User Actions
- For each event, tells helper to run the simulation. (*G4Run*)

UserActionManager

Singleton class used by both the helper and the user to set mandatory and additional user actions

STT G4 Simulation

G4Base (NuTools)
— Construct detector
— Register parallel world

Geometry Package:

StrawTubeGeo
One for each ST
boundaries in world coordinates

NDG4 Package:

StrawTubeReadoutGeometry
G4VUserParallelWorld

Uses G4SDManager to set a Sensitive Detector for each Straw Tube logical volume

★ **StrawTubeReadout**
G4VSensitiveDetector
Sensitive version for each ST

— Parameterize Drift of Ionization Electrons
★ • Drift Velocity
• Ionization Energy (W)
— Add Ionization Electrons onto simulation objects

Simulation Package:

SimTube One for each ST with hits
— Create IDEs (generally for each g4 step) mapped from time tick (tdc) since numElectrons will be the pulse

IDE (id and energy)
— x, y, z in world coordinates
— Energy — TrackID
— Number of ionization electrons

★ Digitization needs to be written

RawDigit **TWO** for each ST with hits
Vector of digitized pulse height (adc) indexed per time tick (tdc). One for each end of wire.

★ Nitali Dash wrote and pushed this a few days ago, still work to do

★ These need to be better estimated or calculated

ECAL G4 Simulation

G4Base (NuTools)
— Construct detector
— Register parallel world
— Set User Actions

Geometry Package:

ScintBarGeo
One for each SB
boundaries in world coordinates

NDG4 Package:

SimScintBarListAction
UserAction (NuTools class)

For each G4Step, check that it is inside a ScintBarGeo. If it is, then add the energy deposition to the corresponding SimScintBar object

Simulation Package:

SimScintBar One for each SB with hits
— Create IDEs (generally for each g4 step) mapped from time tick (tdc)

IDE (id and energy)
— x, y, z in world coordinates
— Energy — TrackID

★ Digitization needs to be written

RawDigit **TWO** for each SB with hits
Vector of digitized pulse height (adc) indexed per time tick (tdc). One for each end of wire.

Soumya is already working diligently on improving and verifying this code, and will transition into writing this digitization

MuonID G4 Simulation

G4Base (NuTools)
— Construct detector
— Register parallel world
— Set User Actions

Geometry Package:

RPCStripGeo
One for each strip
boundaries in world coordinates

NDG4 Package:

?

— Use parallel world or user action?
— Considering LArSoft AuxDet code
—> parallel world

Simulation Package:

SimRPCStrip One for each strip with hits

IDE (id and energy)
— x, y, z in world coordinates
— Energy — TrackID

★ Digitization needs to be written

RawDigit One for each SB with hits
Vector of digitized pulse height (adc) indexed per time tick (tdc).