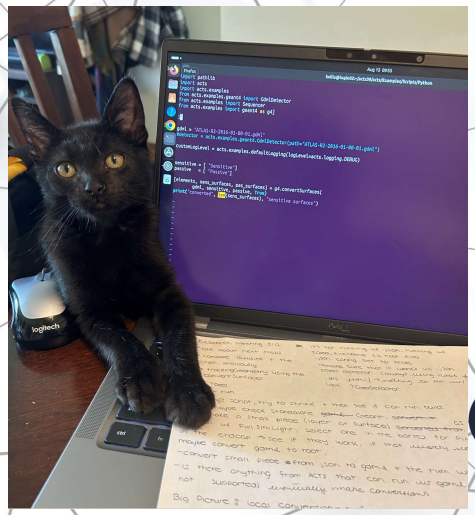
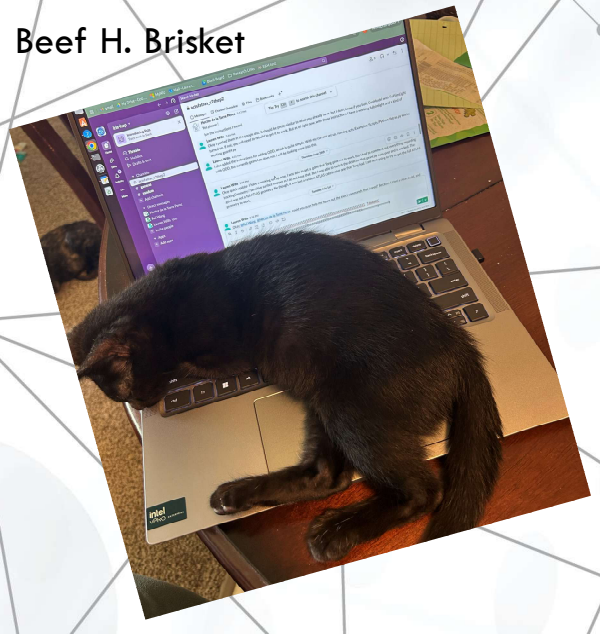
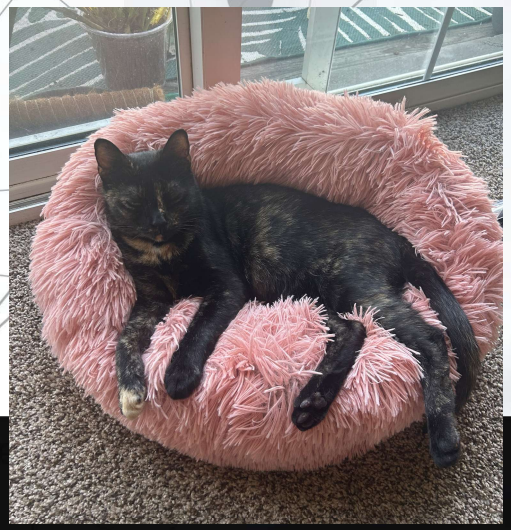


Beef H. Brisket



Ungoliant

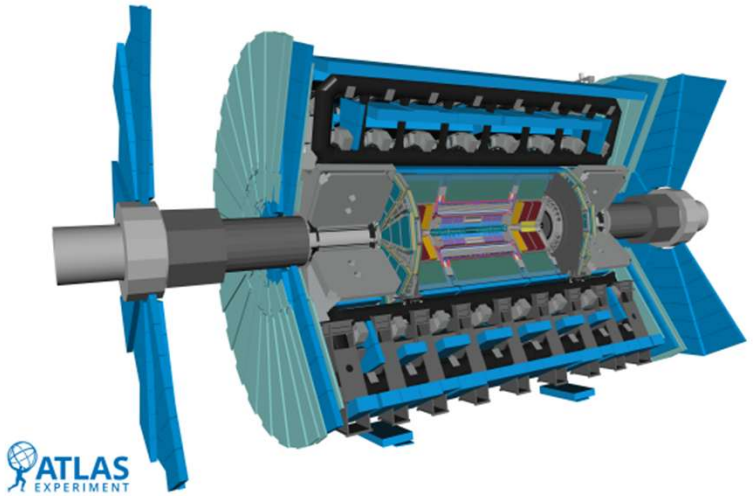


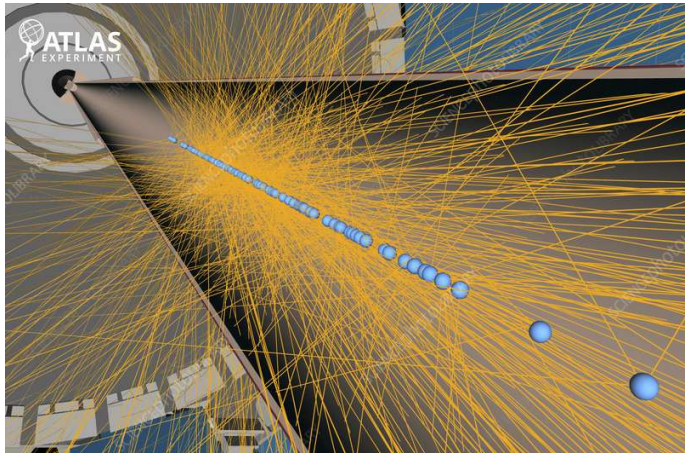
CP^2 UPDATE

Lauren Wills

TODAY'S PRESENTATION

- Introduction to Simulations in ATLAS
- Two types of Simulations- Fatras versus Geant4
- ACTs
- My Project





SIMULATION IN ATLAS

- Allow us to predict how particles will traverse and interact with the detector.
- Importance:
  - Develop and validate reconstruction methods
  - Provide estimates for the background
  - Assessing systemic uncertainties
  - Detector optimization and resource management
- Bridges the gap between detector output and physics
  - Examples- inner detector output

GREAT- HOW DO WE ACTUALLY DO ALL OF THAT?

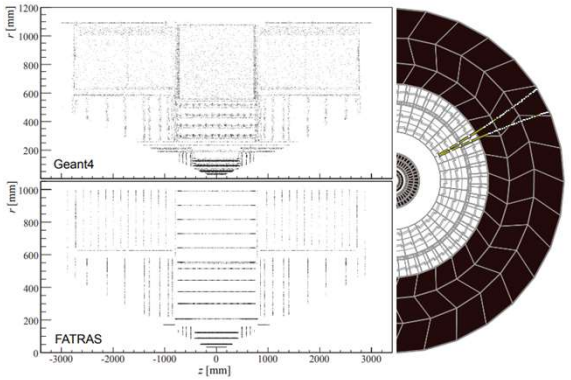
- Geant4
- FatRas
- ACTS

GEANT4

- Very detailed description of the geometry
  - Solid- size and shape
  - Logical- material, sensitivity, magnetic field, etc…
    - Can point to a sensitive volume, which is where the detector records signals
  - Physical- position and rotation in space
- Generation of events (Pythia)
- Properties of the particles + the material, what physical process will occur
  - Monte Carlo
- Physics lists, mathematical model for the selected interaction
- Interaction is executed
- All hits in the sensitive volume are recorded and digitized

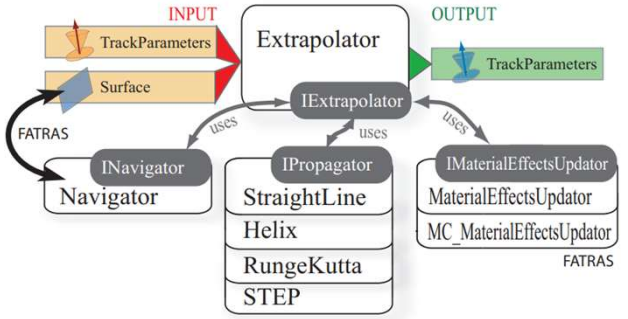
FATRAS- ATLAS FAST TRACK SIMULATION

- Simplified geometry (compared to Geant4)
  - Detector is represented by several volumes
  - Each Volume: collection of boundary surfaces
    - Contain pointer to attached volumes
    - Predict that path of particles as they intersect
  - Sensitive Elements: grouped into layers
    - Records the actual data- 'hits'
    - Example: silicon sensors
- Extrapolation Package
  - Track Parameterization rather than full calculation
  - Navigation through surfaces
  - Integrates the effects of material interactions (energy loss, scattering)
- Prepares the simulated track data to be outputted for reconstruction
  - Includes noise



10.1088/1742-6596/331/3/032046

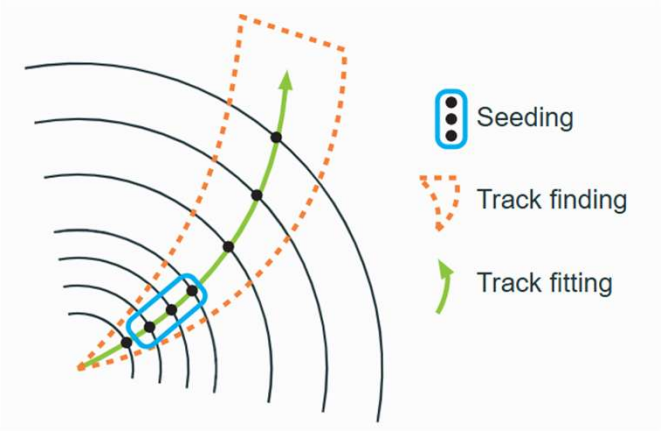
(b) Tracker geometries derived from photon conversions



https://indico.cern.ch/event/408139/contributions/979715/attachments/815586/1117531/CHEP06_Salzburger.pdf

ACTS- A COMMON TRACKING SOFTWARE

- Raw data outputted from simulation
- Finds seeds and constructs track candidates
  - Groups of hits that likely came from the same particle
- Propagates the candidates through the geometry
  - Simplified geometry. Similar approach to that of Fatras but not an identical map
  - Finds compatible hits and associates them with candidate
- Refines the tracks parameters
  - Kalman Filter
- Outputs a collection of reconstructed tracks



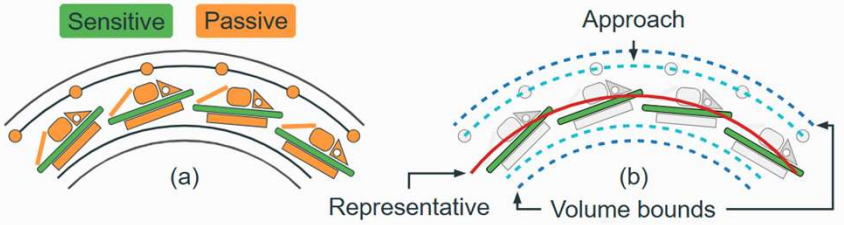
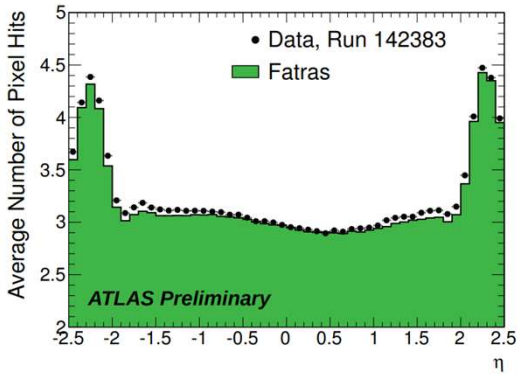


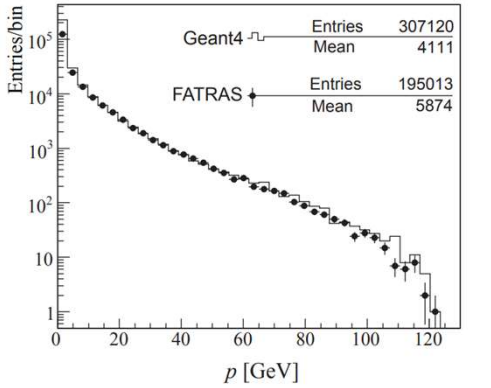
Fig. 7 *Sketch of the way a fully detailed simulation geometry (a) models passive elements, in addition to the sensitive elements shown in green. (b) shows a simplified version, where all non-sensitive elements are approximated.*

FATRAS VS. GEANT4

- Fatras reduces CPU time by a factor of 10 compared to Geant4
- Reproduces Geant4 within ~10% accuracy
- Pretty good at electromagnetic interactions and low energy interactions!
- Problems?? FatRas doesn't handle rare hadronic interactions very well.



(a) Number of pixels hits versus η



(a) Momentum spectra of Bremsstrahlung photons

doi:10.1088/1742-6596/331/3/032046

MY PROJECT

- Hadronic interactions simulated by Geant4, non hadronic interactions simulated by FatRas!
- Geant4 output- volumes
- Fatras output- boundary surfaces and sensitive elements
- Goal: Convert hadronic interactions, represented by volumes into a simplified output that Fatras can use
  - Identify which boundary surfaces correspond to the volume
  - Grouping the hits into layers of sensitive elements
- Propagating both hadronic and non-hadronic interactions by FatRas for a uniform result
- Running within ACTs framework

WORK SINCE APRIL

- Working examples of GeoModel, Geant4, and Fatras with Open Data Detector
- Working Full Sim Light example with SQLite files and gdml (full ATLAS geometry)
- .gdml's inside of ACTS?
  - Geomodel.py takes a produced .gdml file and should be able to convert it to an ACTS understandable object, never got that to work
  - A postdoc that I work with wrote a .json to .gdml conversion file, trying to get that to work anywhere inside of ACTS
  - Tried making geant4 (inside of ACTS) work with this converted .gdml (or any .gdml really) -> able to get it work when trackingGeoemtry is turned off (basically not utilizing the ACTS part), not super useful
  - When trying to get tg = acts.example.detectors.trackingGeometry, the trackingGeometry function uses .geant4convertSurfaces, which doesn't convert to python object correctly
    - Contacted experts, recommended that I update to the latest version of ACTS, did that, re-ran, still doesn't work
    - Unable to get anything with FatRas to work until I have a trackingGeometry
  - Basically: Have not had a single example of a working .gdml in ACTS
- Ran a BuildITK script, that works with .json and root files

CURRENT / FUTURE WORK:

- Trying to get a .gdml to run anywhere inside of ACTs
- Still need to figure out the best formats for conversion and what is even feasible
  - .json to .gdml and back?
  - Could utilize root, .db, .csv, others. Already have a root to .json, a .json to .gdml (maybe), a .gdml to a .csv, a .db to a .gdml
- Will need to validate whatever conversion we do choose, make sure that the map created from the converted file is the same as original
  - There was a ITK map validation script that I tried to get running a few months back, it never ended up working for me (also it currently does not exist right now), but will most likely contact the creators in the future
- For right now, whatever is converted will be saved for the validation, but eventually in the actual implementation the goal (I believe) will be an on-the-fly conversion