Silicon Detector Tracking

Norman Graf
(for the sim/reco team)
LCWS 2010, Beijing
March 29, 2010

Defining the Detector

- lcdd xml file format defines detector at runtime.
  - Arbitrary layouts as it targets Geant4 primitives.
  - Verbose, prone to error and not connected to reco.
- Have a number of detector types defined in our “compact” format.
  - Allow sophisticated geometries to be defined with a minimum of effort.
- Lowest level is silicon wafer
  - Digitization is done at the reconstruction stage.
  - Allows study of strip and pixel layouts.

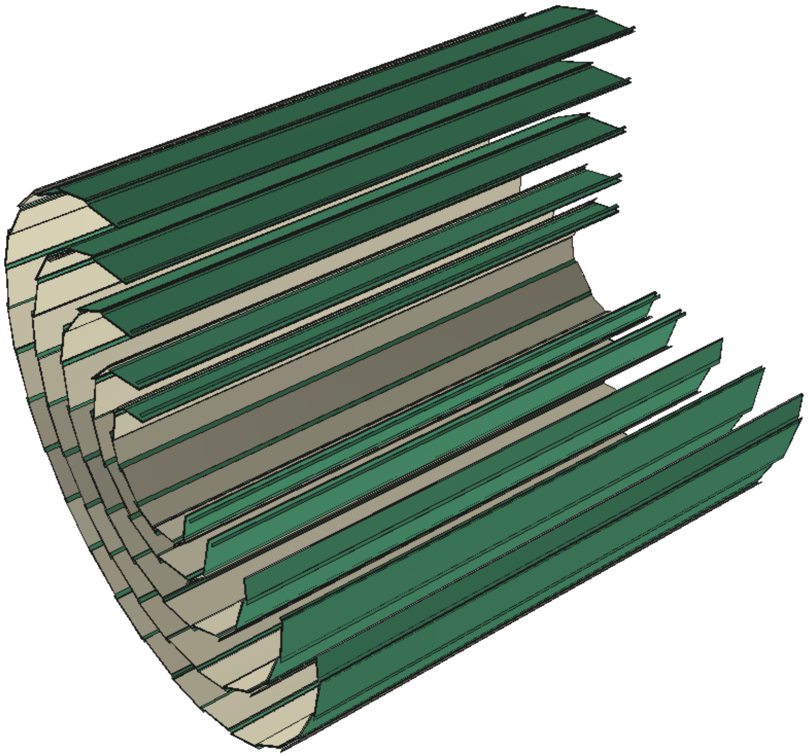
xml: Defining a Module

```
<module name="VtxBarrelModuleInner">
  <module_envelope width="9.8" length="63.0 * 2" thickness="0.6"/>
  <module_component width="7.6" length="125.0" thickness="0.26"
                    material="CarbonFiber" sensitive="false">
                  <position z="-0.08"/>
  </module_component>
  <module_component width="7.6" length="125.0" thickness="0.05"
                    material="Epoxy" sensitive="false">
                  <position z="0.075"/>
  </module_component>
  <module_component width="9.6" length="125.0" thickness="0.1"
                    material="Silicon" sensitive="true">
                  <position z="0.150"/>
  </module_component>
</module>
```

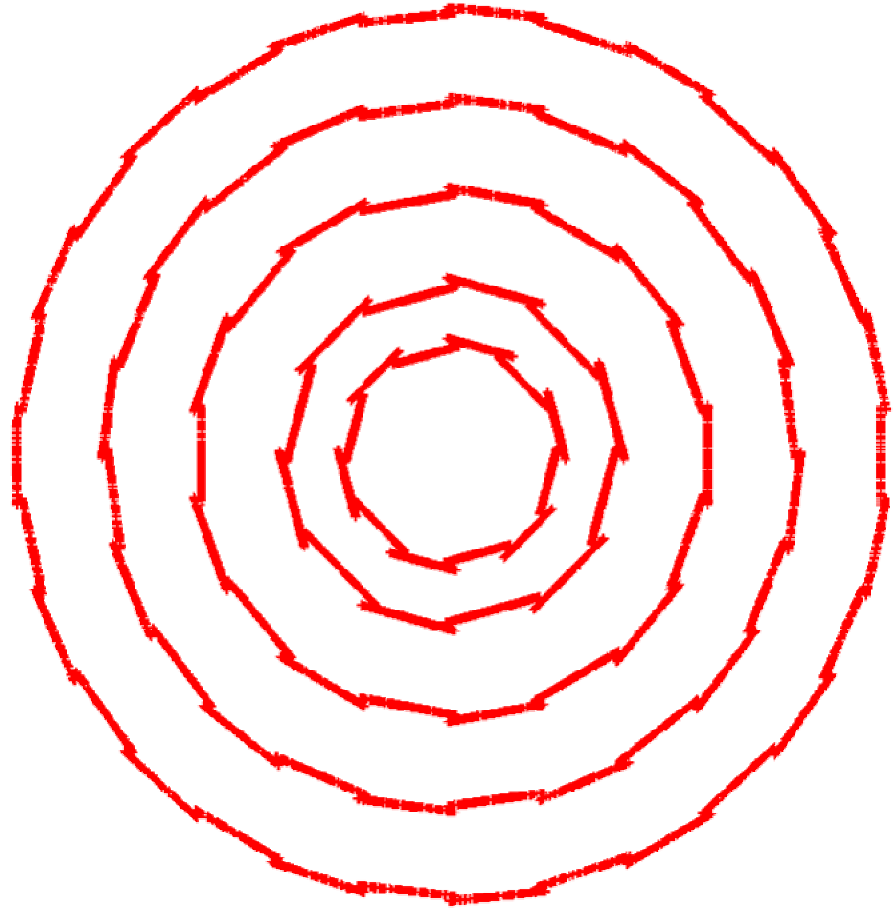
xml: Placing the modules

```
<layer module="VtxBarrelModuleInner" id="1">
        <barrel_envelope inner_r="13.0" outer_r="17.0" z_length="63 * 2"/>
        <rphi_layout phi_tilt="0.0" nphi="12" phi0="0.2618" rc="15.05" dr="-1.15"/>
        <z_layout dr="0.0" z0="0.0" nz="1"/>
</layer>
<layer module="VtxBarrelModuleOuter" id="2">
        <barrel_envelope inner_r="21.0" outer_r="25.0" z_length="63 * 2"/>
        <rphi_layout phi_tilt="0.0" nphi="12" phi0="0.2618" rc="23.03" dr="-1.13"/>
        <z_layout dr="0.0" z0="0.0" nz="1"/>
</layer>
<layer module="VtxBarrelModuleOuter" id="3">
        <barrel_envelope inner_r="34.0" outer_r="38.0" z_length="63 * 2"/>
        <rphi_layout phi_tilt="0.0" nphi="18" phi0="0.0" rc="35.79" dr="-0.89"/>
        <z_layout dr="0.0" z0="0.0" nz="1"/>
</layer>
<layer module="VtxBarrelModuleOuter" id="4">
        <barrel_envelope inner_r="46.6" outer_r="50.6" z_length="63 * 2"/>
        <rphi_layout phi_tilt="0.0" nphi="24" phi0="0.1309" rc="47.5" dr="0.81"/>
        <z_layout dr="0.0" z0="0.0" nz="1"/>
</layer>
<layer module="VtxBarrelModuleOuter" id="5">
        <barrel_envelope inner_r="59.0" outer_r="63.0" z_length="63 * 2"/>
        <rphi_layout phi_tilt="0.0" nphi="30" phi0="0.0" rc="59.9" dr="0.77"/>
        <z_layout dr="0.0" z0="0.0" nz="1"/>
</layer>
```

The Barrel Vertex Detector

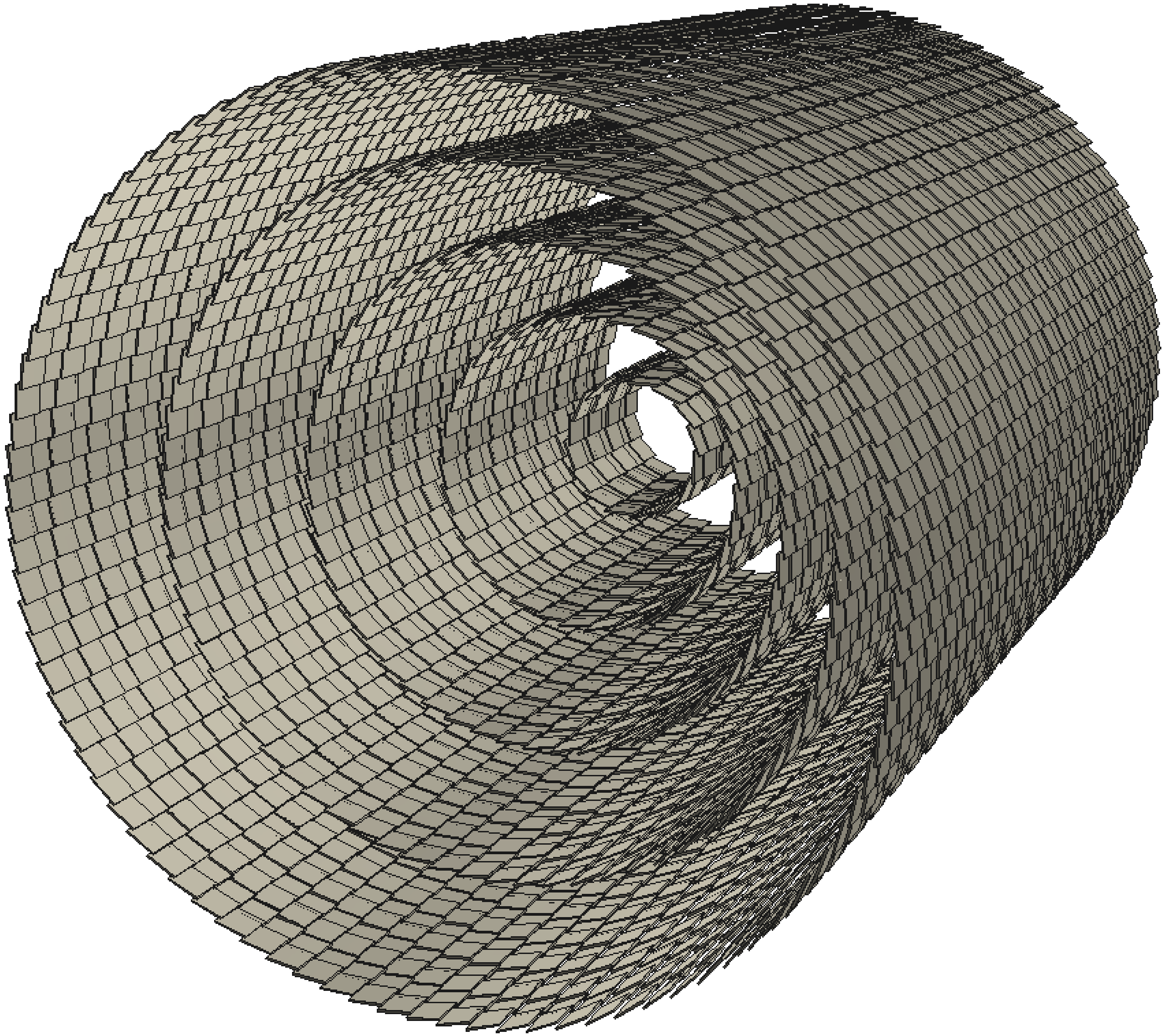


Vertex Barrel staves

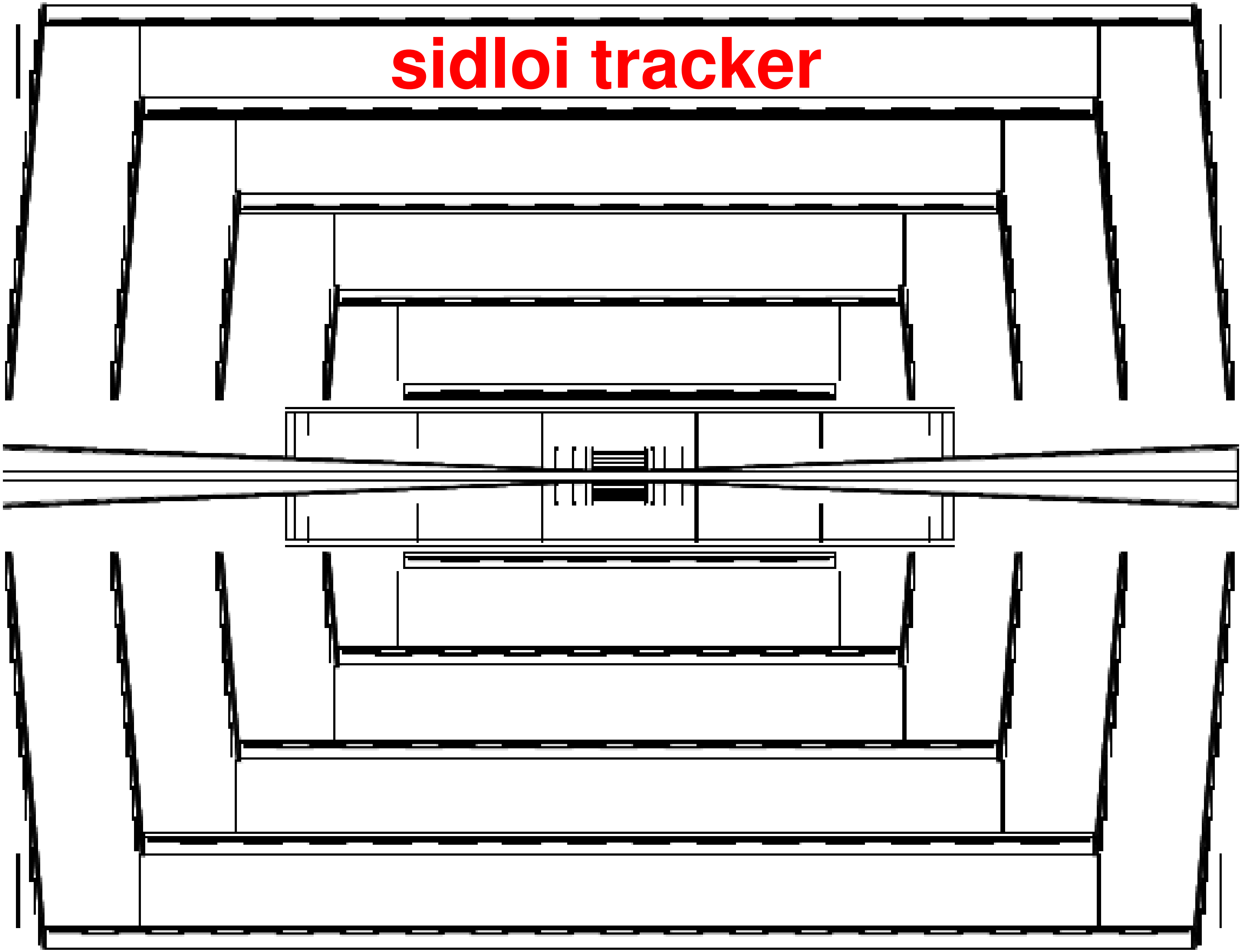


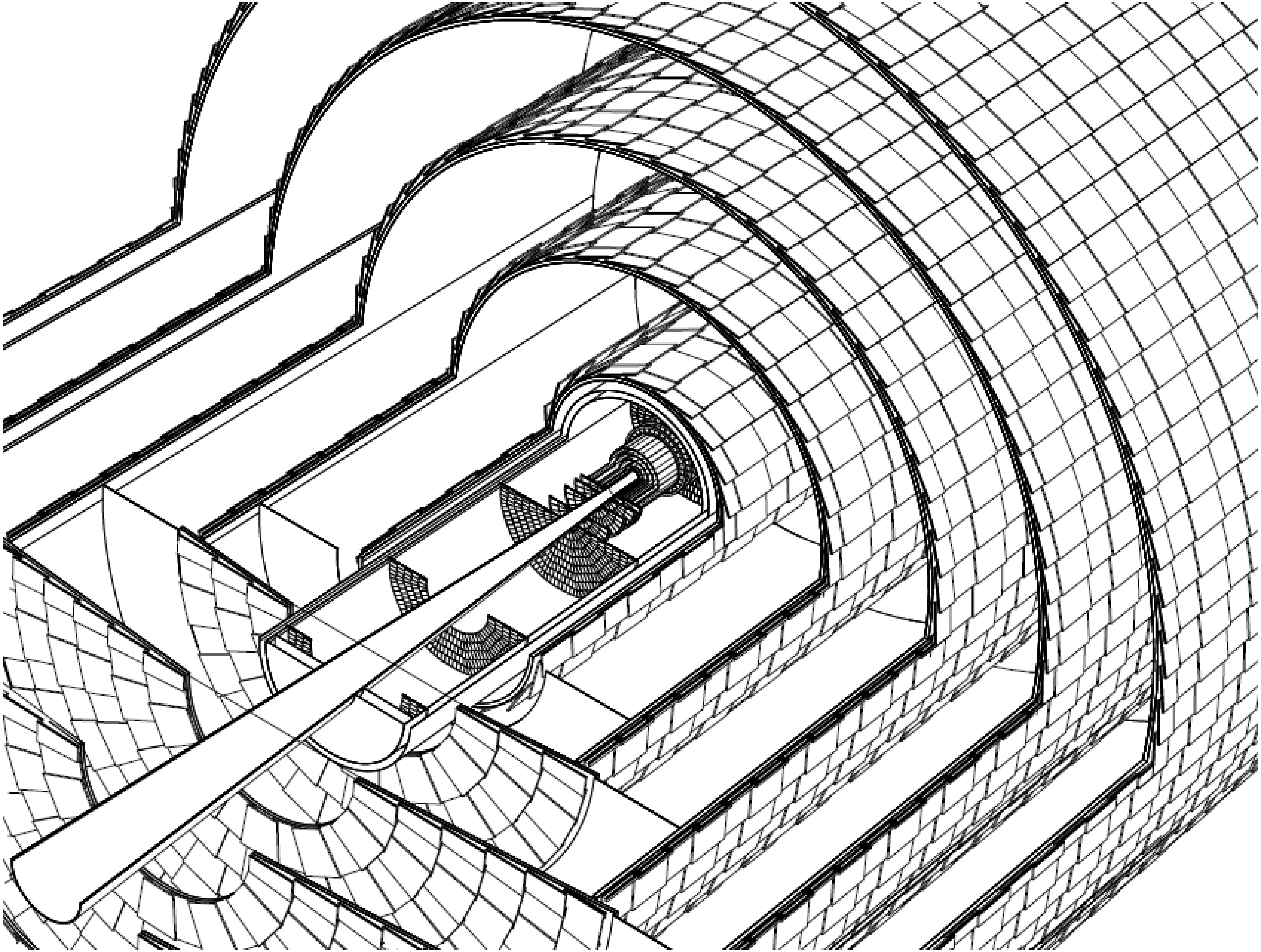
SimTrackerHits in Vertex Barrel

The Barrel Outer Tracker



sidloi tracker





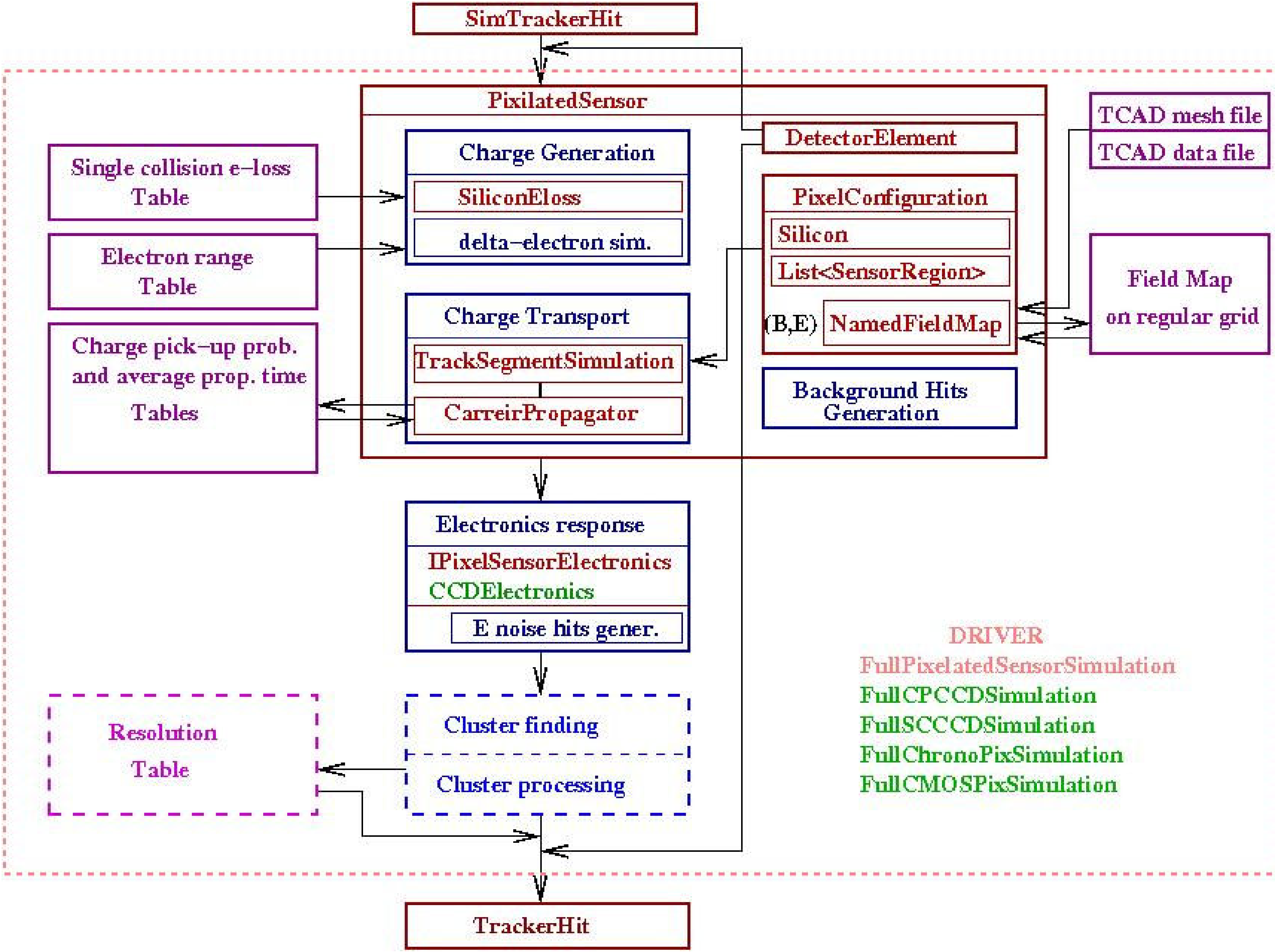
Hit Digitization

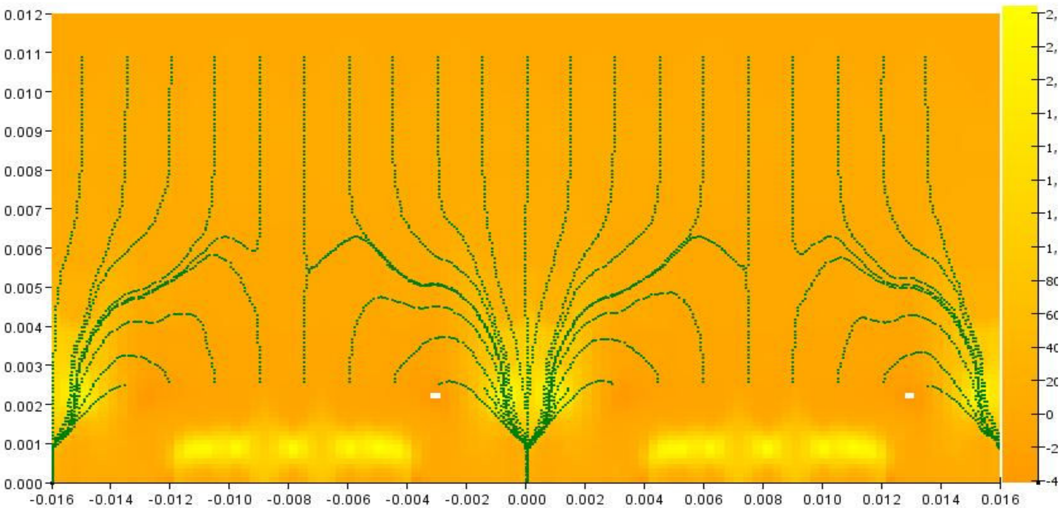
MC Hits $\rightarrow$ Readout ID, ADC, t $\rightarrow$ Clusters $\rightarrow$ Hits ($x \pm \delta x$)

- Charge deposition modeled by drifting and diffusing the charge to strips or pixels.
- Readout electronics modeled to study effects of noise, cross-talk and inefficiencies.
- Strip detectors based on CDF Si sensor simulation algorithm implemented by Tim Nelson, extended by Rich Partridge.
- Pixel detectors use either the strip simulation algorithm or a detailed modeling using electric field maps developed by Nick Sinev.
- Strip/pixel charges clustered by a nearest neighbor algorithm.
- Tracker hits formed from clusters with expected hit errors.

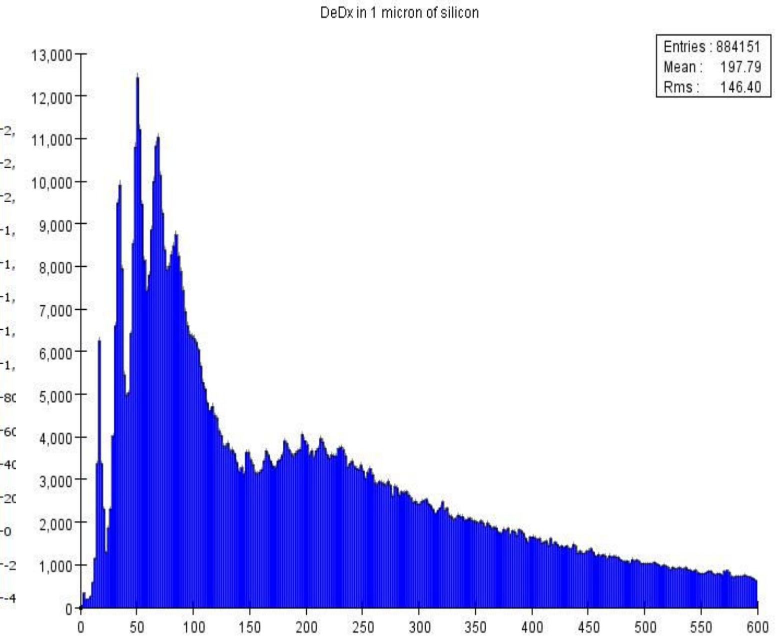
PixSim

- Very detailed but flexible simulation of pixel detector response to charge deposition
  - Dimensions, media, E & B field maps (incl. TCAD)
  - Energy loss simulation using Bichsel code.
  - Electronics response (e.g. CCD, chronopixel, …)
    - Addition of electronics noise to collected charge
    - Propagation of the signal to readout – simulation of CCD clocking, and combining of signals from different charge trains
    - Determination if signal exceeds threshold
    - Assigning signals to particular bunch crossing
    - Digitization of the signal
    - Simulation of fake hits due to electronics noise
  - Fixed configurations can be preprocessed into tables of response, allowing use in full event simulation.


SimTrackerHit
PixilatedSensor
Charge Generation
SiliconEloss
delta−electron sim.
Charge Transport
TrackSegmentSimulation
CarreirPropagator
DetectorElement
PixelConfiguration
Silicon
List<SensorRegion>
(B,E)
NamedFieldMap
Background Hits Generation
TCAD mesh file
TCAD data file
Field Map on regular grid
Single collision e−loss Table
Electron range Table
Charge pick−up prob. and average prop. time Tables
Electronics response
IPixelSensorElectronics
CCDElectronics
E noise hits gener.
Cluster finding
Cluster processing
Resolution Table
DRIVER
FullPixelatedSensorSimulation
FullCPCCDSimulation
FullSCCCDSimulation
FullChronoPixSimulation
FullCMOSPixSimulation
TrackerHit



Example TCAD map for Chronopixel sensor



Example of energy loss distribution for 1 micron thick silicon layer (x axis in eV), calculated by SiliconEloss class.

SiSim

- Track sub-divided into segments
- Each segment drifted to sensor electrodes accounting for Lorentz angle
- Charge for segment divided among electrodes accounting for diffusion
- Charge transferred from sensor electrodes to readout electrodes as established by a charge transfer matrix – provides way to include capacitive coupling to intermediate and neighbor

SiSim Readout Chip Simulation

- Both GenericReadoutChip and KPix available
- Provides
  - Charge to ADC count conversion
  - Adds noise to channels with hits as well as adding random noise hits
  - Encodes / decodes raw data format
  - Settable hit threshold & neighbor threshold
  - Simple linear noise formula
    - Sig(Noise) = NoiseOffset + NoiseSlope * Capacitance

SiSim Additions

- Pixel simulation
  - Provides single algorithm for pixel and strip digitization, clustering, and hit making.
  - Mapping a 2D Gaussian charge distribution onto pixels using BivariateDistribution class
- Strip and Pixel Clustering
  - New nearest neighbor algorithm with settable cluster seed and neighbor thresholds.
  - Use HashMaps to achieve linear scaling with hit occupancy

Track Finding

- Some structural/implementation changes to improve tracking performance, especially with large numbers of hits.
  - Divided tracking volume and developed “quick check” algorithms to improve speed and memory.
  - Improved algorithm for forming HelicalTrackCross hits in stereo layers.
- StrategyBuilder automatically constructs strategies
  - Strategies used to guide track reconstruction
  - Specify p_T and impact parameter constraints and χ^2 cuts
  - Specify layers to be used for a given strategy and role of each layer (seed, confirm, or extend)

Running the code

- Attempt to expose all controls and settings to end user via xml file → runtime control.
- Systematically study detector response effects due to:
  - readout technology, size and layout
  - electronics response, crosstalk, efficiencies and noise

```
<driver name="TrackerBarrelSetup"
    type="org.lcsim.recon.tracking.digitization.sisim.config.SiTrackerBarrelSensorSetup">
        <subdetectorName>SiTrackerBarrel</subdetectorName>
        <readoutElectrodesPitch>0.050</readoutElectrodesPitch>
        <senseElectrodesPitch>0.025</senseElectrodesPitch>
        <transferEfficiencies>0.986 0.419</transferEfficiencies>
    </driver>
```

Tracker Digitization Control

```
<driver name="TrackerDigi"
        type="org.lcsim.recon.tracking.digitization.sisim.config.StripDigiSetupDriver">
      <subdetectorNames>SiTrackerBarrel SiTrackerEndcap</subdetectorNames>
      <rawHitsCollectionName>TKR_RawTrackerHits</rawHitsCollectionName>
      <trackerHitsCollectionName>TKR_TrackerHits</trackerHitsCollectionName>
      <maxClusterSize>10</maxClusterSize>
      <noiseIntercept>300.</noiseIntercept>
      <noiseSlope>30.</noiseSlope>
      <noiseThreshold>6000.</noiseThreshold>
      <readoutNeighborThreshold>6000.</readoutNeighborThreshold>
      <seedThreshold>6000.</seedThreshold>
      <neighborThreshold>6000.</neighborThreshold>
      <centralStripAveragingThreshold>4</centralStripAveragingThreshold>
      <oneClusterErr>0.288675135</oneClusterErr>
      <twoClusterErr>0.2</twoClusterErr>
      <threeClusterErr>0.333333333</threeClusterErr>
      <fourClusterErr>0.5</fourClusterErr>
      <fiveClusterErr>1.0</fiveClusterErr>
    </driver>
```

SiD Tracking Improvements post-LOI

LOI Implementation

- Include all active and dead material in tracker geometry
- Virtual segmentation divides cylinders / disks into sensors
- No overlapping sensors, gaps
- No charge deposition modeling
- Simple clustering to make hits
- 3D stereo hits formed for forward disks including ghost hits
- SeedTracker and Calorimeter Assisted Tracking algorithms
- Simple helix fitter used for pattern recognition and final track fit (no multiple scattering correlations, circle + s-z fits)

Post-LOI Improvements

- Update material as needed to reflect changes in tracker design
- Individual planar sensors to match detailed geometry
- Overlapping sensors, realistic gaps
- Realistic charge deposition
- Improved clustering / hit making
- 3D stereo hits formed for forward disks including ghost hits
- Incremental improvements in algorithms – no big changes
- Continue to use simple helix fitter for pattern recognition, long term goal is to implement a Kalman filter for final track fit

Tracking Summary

- Digitization improves from the virtual segmentation used in the LOI to full digitization of SimTrackerHits into ADC counts in pixels and strips, using detailed drift, diffusion, readout, …
- Full clustering of neighboring hits, giving cluster-dependent measurement position and uncertainties.
- Tracking works well out of the box.
  - Currently characterizing performance.
  - Testing and optimization remains to be done.