

Norman Graf (SLAC National Accelerator Laboratory)
on behalf of the ALCPG Simulation and Reconstruction Working Group

Goal

As the complexity and resolution of imaging detectors increases, the need for detailed simulation of the experimental setup increases. Designing the detectors requires efficient tools to simulate the detector response and reconstruct the events. We have developed efficient and flexible tools for detailed physics and detector response simulation as well as event reconstruction and analysis. The primary goal has been to develop a software toolkit and computing infrastructure to allow physicists from universities and other labs to quickly and easily conduct physics analyses and contribute to detector R&D.

Detector Response Simulation Using Geant4

The Geant4 toolkit is the de facto high-energy physics standard for simulating the interaction of particles with fields and materials. However, the end user is normally required to write their own C++ program to access the libraries, and the learning curve for setting up the detector geometry and defining sensitive elements and readout can be quite daunting. We have developed the detector response package, slic, which is based on the Geant4 toolkit but defines generic input and output formats as well as a textual detector description. This allows the end user to fully describe the detector geometry and readout at runtime using a plain text file in an xml format. We provide executable programs for Windows, Mac OSX and several flavors of Linux.

Mission Statement & Architecture

- ❖ Provide full detector response simulation program and free the end user from having to write C++ code to study a detector.
- ❖ System should be powerful, yet simple to install and maintain.
- ❖ Need flexibility for new detector geometries and technologies.
- ❖ All of the detector properties should be definable at runtime with an easy to use format.
- ❖ Use Geant4 for interaction of particles with matter and fields.
- ❖ Thin layer of non-Geant4 C++ code provides access to:
 - Event Generator and Detector Geometry description input
 - Detector Hits output
- ❖ Selected xml, and extended the existing GDML format for pure geometry description, then extended it to incorporate missing detector elements.

Why XML?

Simplicity: Rigid set of rules
Extensibility: easily add custom features, data types
Interoperability: OS, languages, applications
Self-describing data, validate against schema
Hierarchical structure: OOP, detector/subdetector
Open W3 standard, lingua franca for B2B
Many tools for validating, parsing, translating
Automatic code-generation for data-binding
Plain text: easily edited, cvs versioning



LCDD

detector identifiers
sensitive detectors
regions, limits & cuts
physics lists
electromagnetic fields

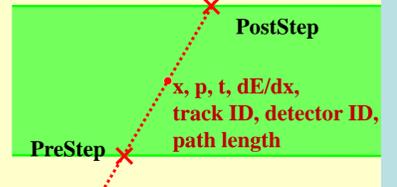
GDML

expressions
materials, solids
volume definitions
geometry hierarchy

I/O

- ❖ LCIO
 - Simple event data model and persistency framework used by LC community.
 - Generic Tracker and Calorimeter Hits & Monte Carlo particle hierarchy.
 - Performant, with on-the-fly data compression, random access.
 - Well documented, with C++, Java, python & FORTRAN bindings
- ❖ Generic TrackerHit
 - MC Track handle
 - Encoded detector ID
 - Hit position in sensitive volume.
 - Track momentum at hit position.
 - Energy deposited in volume.
 - Time of track's crossing.
 - Path length in sensitive volume.

SimTrackerHit



Event Reconstruction Using org.Icsim

org.Icsim is a Java toolkit for full event reconstruction and analysis. Java provides both a very powerful object-oriented language for development and transparent cross-platform portability. The components are fully modular and are available for tasks from digitization of tracking detector signals through to cluster finding, pattern recognition, track fitting, and analysis. Very detailed but flexible simulation of the response of position sensitive silicon detectors to charge deposition is possible, including variable readout dimensions (e.g. pixels or strips), various media, E & B field maps (including TCAD), detailed energy loss simulation using specialized code, electronics response, including electronics noise or inefficiencies, propagation of the signal to readout - e.g. simulation of CCD clocking and digitization of the signal. The code can be run standalone, for batch or Grid computing, or from within the JAS3 IDE, which then provides access to the WIRED event display and the AIDA-compliant analysis capabilities.

Digitization involves a detailed simulation of the ionization generation in the sensitive layer of the sensor, charge collection, signal formation and signal processing. This has two major goals: optimizing sensor parameters and comparing different sensors, and providing an estimation of the full detector performance..

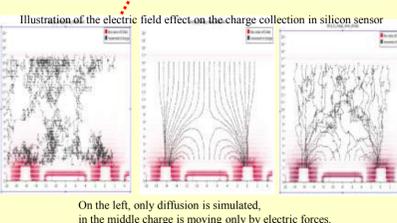
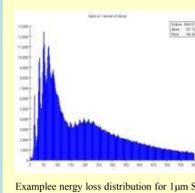
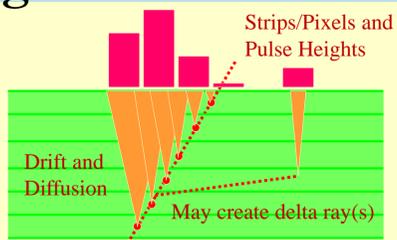
Overview

- ❖ Java based reconstruction and analysis package
 - Runs standalone or inside Java Analysis Studio (JAS)
 - Fast MC → Smear tracks and calorimetry clusters
 - Full Event Reconstruction
 - Beam background overlays at detector hit level, including time offsets.
 - detector readout digitization (CCD pixels, Si μ -strips, TPC pad hits)
 - *ab initio* track finding and fitting for ~arbitrary geometries
 - multiple calorimeter clustering algorithms
 - Individual Particle reconstruction (cluster-track association)
 - Analysis Tools (including WIRED event display)
 - Physics Tools (Jet Finding, Vertex Finding, Flavor Tagging)
- ❖ Write once run, run anywhere
 - Exact same libraries run on all platforms (Windows, Mac, Linux(es), Grid) using the Java Virtual Machine.

Digitization

PixSim (N. Sinev, UO)

Allows very detailed descriptions of charge carrier movement, e.g. list of collecting, absorbing and reflecting regions, properties of silicon (doping, mobility, diffusion length and so on), electric and magnetic fields (including TCAD maps). Use of lookup tables leads to faster simulation.



On the left, only diffusion is simulated, in the middle charge is moving only by electric forces, and the right picture shows how it moved in our simulations

Track Reconstruction

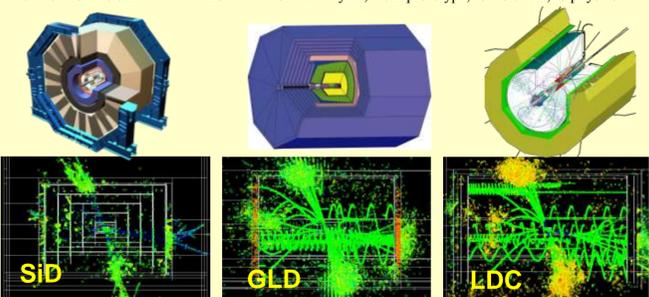
- ❖ Analytic covariance matrices available for fast MC smearing for each detector.
- ❖ Track "cheater" available for studies of full detector simulation events. Assigns hits on basis of MC parentage.
- ❖ Ab initio track finding packages.
- ❖ Standalone pattern recognition code for 1D (e.g. Si μ strip), 2D (e.g. Si pixel) and 3D (e.g. TPC) hits.
 - High efficiency, even in presence of backgrounds.
 - Efficient at low momentum.
- ❖ Conformal-mapping pattern recognition also available.
- ❖ MIP stubs in highly segmented calorimeters also provide track candidates, propagate inwards to pick up tracker hits.
- ❖ Fitting code incorporating multiple scattering and energy loss via weight matrix or Kalman Filter available.

Full Detector Simulations

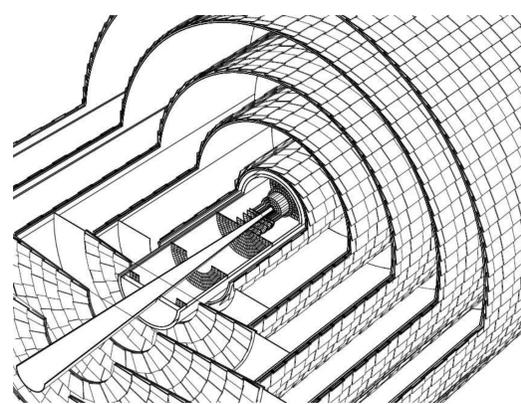
This package is capable of simulating a large variety of detectors and is being used to characterize and optimize the performance of the Silicon Detector (SiD), being proposed for the International Linear Collider.

ILC Full Detector Concepts

- Runtime XML format allows variations in detector geometries to be easily set up and studied.
- Sampling calorimeters:
 - absorber materials, dimensions
 - Readout technologies, e.g. RPC, scintillator
 - Layering (radii, number, composition)
 - Readout segmentation (size, projective vs. nonprojective)
- Total absorption crystal calorimeters:
 - Optical properties
 - Tracking detector technologies & topologies
 - TPC, silicon microstrip, silicon pixels
- The same executable is used to define all of the detectors, simply by reading in different xml files.
- Can be used to simulate the whole detector R&D cycle, from prototype, to testbeam, to physics.



SiD Pixel & Microstrip Tracker



Simulation Summary

- ❖ ALCPG sim/reco supports an ambitious international detector simulation effort. Goal is flexibility and interoperability.
- ❖ Provides a complete and flexible detector simulation package capable of simulating arbitrarily complex detectors with runtime detector description.
- ❖ Reconstruction & analysis framework is being used to characterize and optimize the Silicon Detector physics performance at the ILC & CLIC.
- ❖ Also being used for Muon Collider detector studies.
- ❖ Used to design HPS detector and analyze data for test run at Jlab.
- ❖ LCIO provides interoperability with tools developed in other regions (e.g. jet flavor tagging (LCFI), particle flow (Pandora)), other languages (FORTRAN, java, C++, python) and other analysis frameworks (e.g. Marlin, root).