Simulation and Reconstruction Software at FNAL

Rob Kutschke, FNAL

ILC Vertex Detector Review

October 25, 2007

Outline and Preliminary Comments

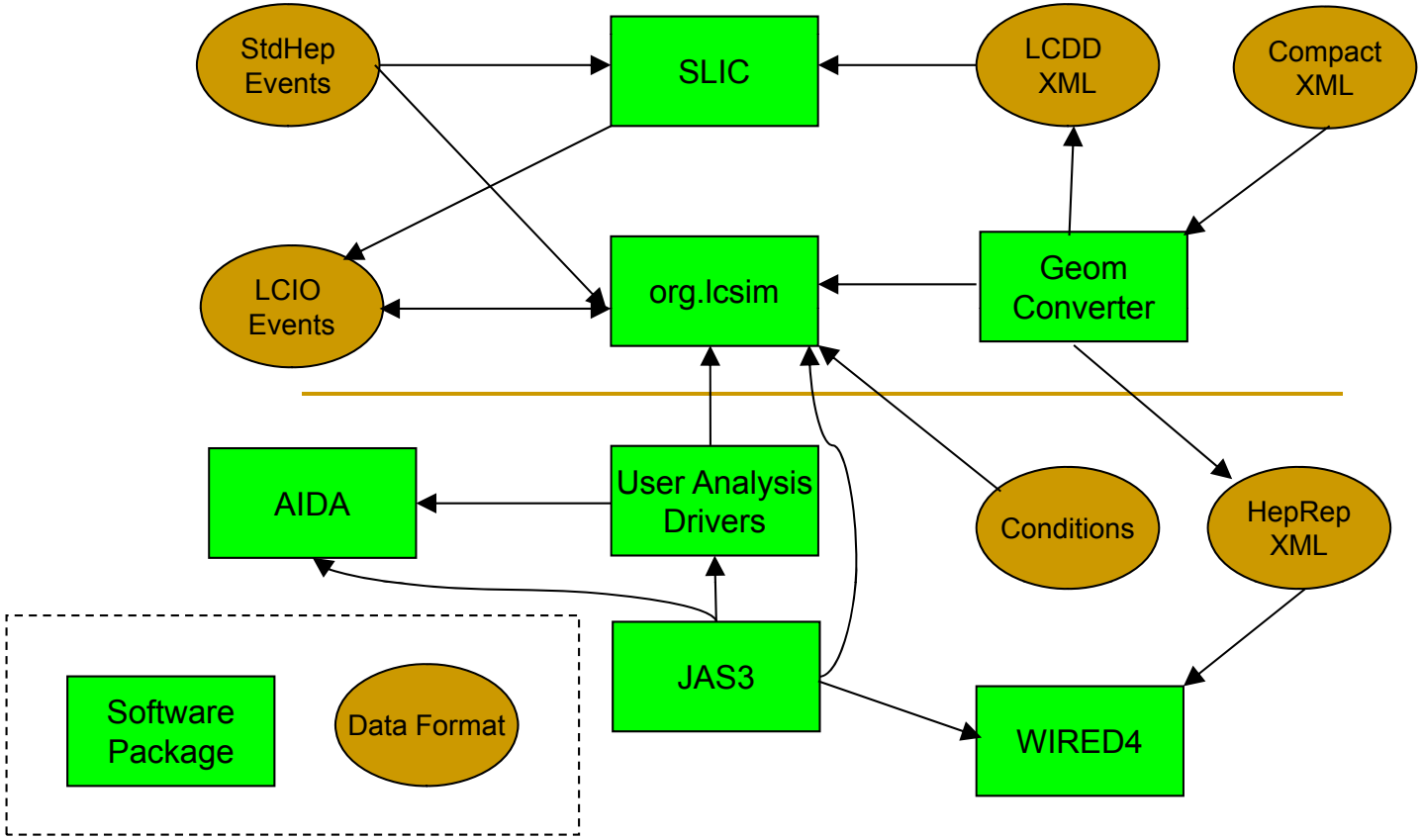
- People
- Background information
- Work plan on one page
- Work plan details
- Schedule drivers
- Computing Resources at Fermilab
- Summary

- All work reported here is SiD specific.
- Su Dong showed work already done.
  - I will discuss next steps.
- My focus is projects that involve the vertex detector and are worked on by FNAL staff.

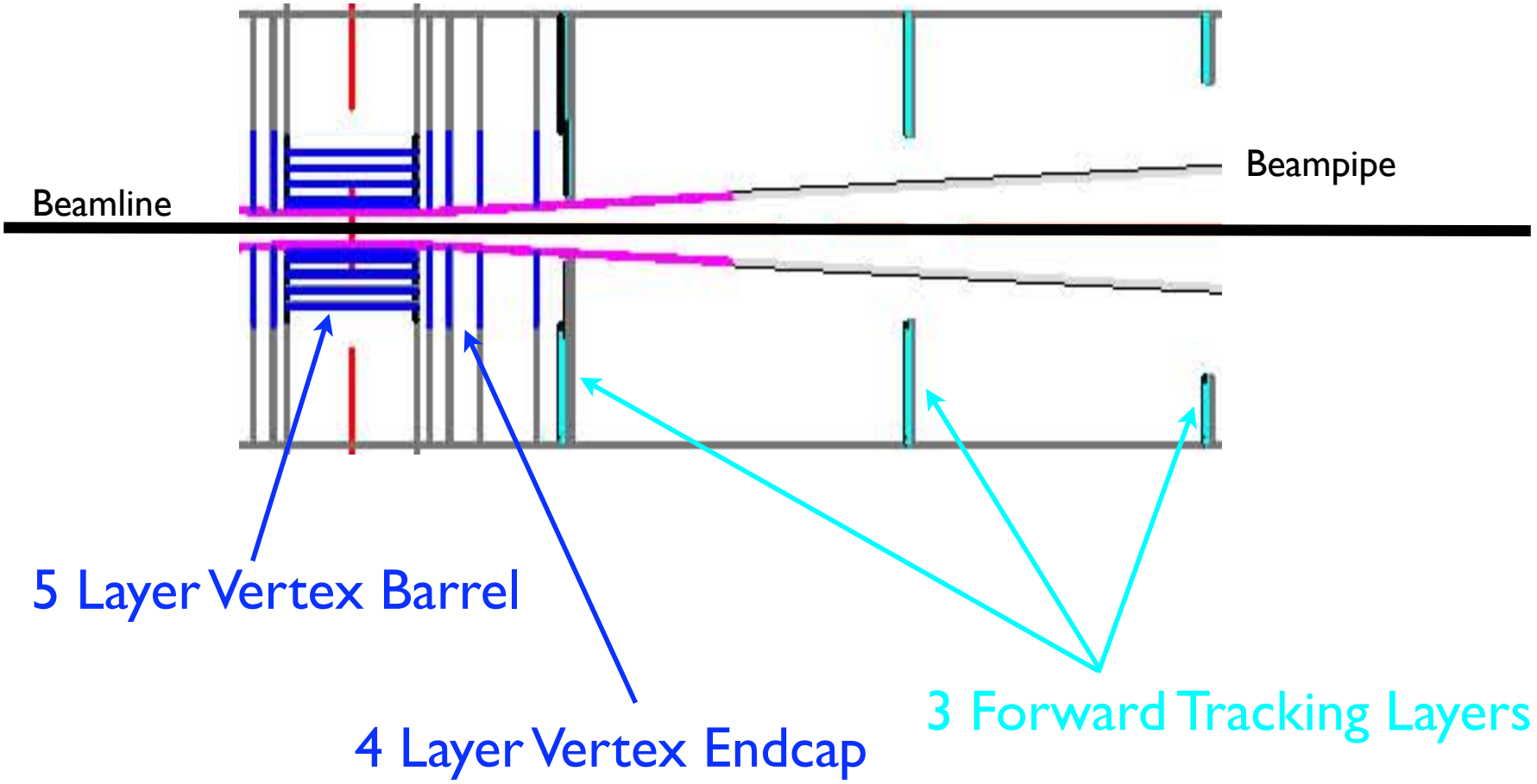
People

- Working on the main issues:
  - Hans Wenzel (100%), Rob Kutschke(~75%).
- Software distribution and www support:
  - Lynn Garren (small fraction).
- We are using work (still in progress) by:
  - Dima Onoprienko (Kansas State).
  - Tim Nelson and Jeremy McCormick (SLAC).
  - Both are working on making "hits".

Overview: "SiD/ALCPG" Framework



Endcap vs Forward



Goal: treat the Vertex Detector, the main Tracker and the Forward Tracker as an integrated system.

SiD Detector Models

- sid00
  - Complete but simplified sensitive volumes.
    - Barrel vertex detector and tracker are pure cylinders.
    - Endcaps are annulus of disk.
  - Lots of exisiting MC needed by PFA people.
- sid01
  - As above but more detail of dead material.
  - Added forward tracker.
  - Current official model: lots of existing MC.
- Wafer based model:
  - Under development:
  - Will need several variations, especially in forward region, including variations of dead material.

Work Plan Overview

1) Study occupancies in endcaps/forward region and define candidate tiling(s).
2) Interface TRF Kalman filter to org.lcsim for use as a standalone final fitter.
3) Learn how to use existing vertexing code, including but not limited to LCFI.
4) Pattern recognition code for forward region and integrate it with codes for barrel.
5) Join simulated analysis effort.
6) Infrastructure for the above.
7) Miscellany.

1) Occupancy Studies (Hans)

- Define candidate (virtual) tilings of forward region
  - Evaluate occupancy.
  - Start with sid01.
- Define options for wafer based versions.
  - In collaboration with rest of SiD.
- Currently two hit generation codes exist:
  - Tim Nelson and Jeremy McCormick:
    - Details of cluster shapes.
  - Dima Onoprienko:
    - Driven by fast startup for Calorimeter assisted tracking.
    - See talk in Thursday Simulation Session
  - Meet Thursday to understand how to move forward.
- Results in a few weeks.

2) Kalman Filter (Rob)

- Interface TRF to org.lcsim geometry and hit classes for use as a final fitter.
  - Support sid00, sid01 and wafer based version.
  - Initially use cheating pattern recognition.
- Produce track information valid at:
  - PCA to beamline; Innermost hit; Outermost hit.
- Support use by: pattern recognition, vertex finding/ fitting, track/shower matching codes.
- Status:
  - Internal self tests nearly complete (hardest part).
  - Complete the interface to the outside world.
  - Goal for a useable product: Christmas/2008.
  - See talk in Thursday Simulation Session.

3) Learn Vertexing Codes (Rob)

- Known options:
  - org.lcsim port of ZVTop?
  - LCFI
    - Either send our tracks to Marlin or learn to call C++ from java and use it in its native form.
    - I hear rumors that LCFI has hard ties to ILD geometry.
      - Need to work around this.
- Use this experience as feedback to evolve the output of the track fitter.
- Details will be driven by needs of benchmark analyses.
- Start once Kalman filter released.

4) Pattern Recognition Codes (Hans)

- Initial goal is to develop code that works in the endcap/forward region.
  - Evaluate track finding efficiency and give feedback about number and placement of layers.
- Either extend this to run in the barrel or integrate this with existing codes that run in the barrel.
  - Ultimately we want to view pattern recognition as an integrated task that considers all of Tracker+Vertex Detector + Forward Tracker.
- Include realistic backgrounds.
- Start once occupancy studies are complete.
  - Expect this to be ongoing for a long time.

5) Simulated Analyses (Rob)

- Choose simulated analyses that stress the full tracking/vertexing system.
  - b and charm tagging stress vertexing:
    - First suggestion is B(H-> b bar) and B(H -> c cbar)
      - May be oversubscribed.
      - If so we will discuss other options with benchmarking group.
- Will also pick analyses that stress forward resolution and forward pattern recognition.
  - Have not thought about this yet.
- Start concurrent with work on vertexing codes.

6) Infrastructure (Both)

- Some classes are not rich enough for our needs:
  - Hits
    - Several efforts underway: Onoprienko, Nelson.
  - Tracks
    - Started discussion in ILC Forum. Little response.
- Bookkeeping of used hits.
  - Needed to layer pattern recognition algorithms.
  - Several private codes already exist in org.lcism.
- Persistence:
  - Administrative barriers in changing persisted classes.
  - Will ignore persistency until we have working code that uses our new classes.

7) Miscellany

- Projects that we think are needed but will only do if they are in the direct path of our work:
  - Run-time configuration system for use with org.lcsim.
  - Improve release management.
    - Becomes important if we start supporting users as distinct from developers.

Schedule Drivers

- LOI: due Oct 1, 2008.
- June 1 2008:
  - Final simulation and reconstruction software ready.
  - Simulated analyses of benchmark physics processes developed and ready to run on final software.
- February 2008
  - FNAL software in good enough shape that people doing analyses can use it.
  - Some algorithms not final but interfaces will be.
  - Real pattern recognition will still be immature.
    - Can get started using cheating pattern recognition.

Computing Resources at FNAL

- Machines for small jobs and Grid access:
  - ilcsim(4x1cores); ilcsim2 (2x2 cores).
- Grid Resources
  - Fermigrid: O(1000) cores 2 to 3 GHz.
    - Authorized: 150 cores (300 when need demonstrated).
  - Spare cycles on other grid installations.
- Networked Attached Storage (BlueArc)
  - 15 TB shared accelerator modeling (small users).
  - Separate partition for code/constants
  - Mounted on user machines, ilcsim*, Fermigrid nodes.
- To get accounts on ilcsim* or to join VO
  - Contact Lynn Garren (garren@fnal.gov).

Reducing Barriers to Entry

- It can take a long time to learn simple things.
  - Software is not ready for users (vs developers).
  - Documentation is incomplete and scattered:
    - Answers that do exist are generally very good.
    - But you need to ask exactly the right question.
- Make the software ready for users.
  - Responsibility of all of us working on it.
- Create comprehensive tutorial/example suite.
- At FNAL software is all installed on ilcsim*.
  - Goal: to have first histograms in 5 minutes.
- If we could convert aida files to root, it would help recruiting at FNAL for simulated analyses.

Summary and Conclusions

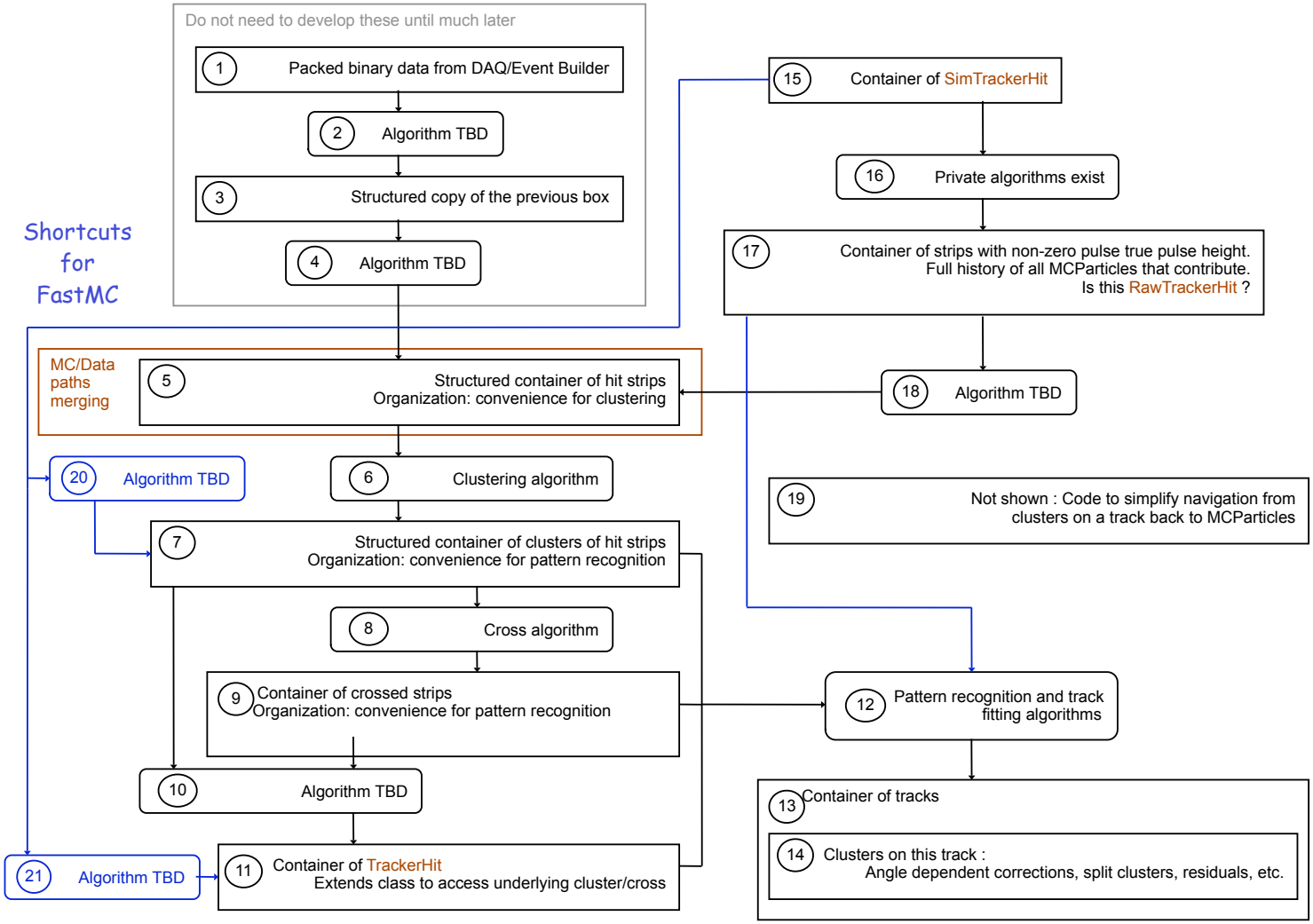
- Fermilab has two experienced people working on tracking in general and forward tracking specificially.
- Projects to be completed soon:
  - Occupancy studies and definition of candidate tilings.
  - Kalman filter on cheated pattern recognition.
- Projects to start after these:
  - Forward pattern recognition
  - Learning vertexing codes
  - Simulated analyses.
- There remain some infrastructure issues.
- There remain barriers to entry that must be a reduced soon.

Backup Slides

ALCPG Software Suite: Figure Page 4.

- SLIC:
  - G4 based simulation system.
- org.lcsim
  - Java based framework for reconstruction and analysis.
- Wired 4 based event display.
- GeomConverter:
  - Reads "Compact Detector Description" XML.
  - Native format for org.lcsim.
  - Can write:
    - HepRep XML for Wired-4
    - LCDD XML for SLIC
- Data formats: StdHep and LCIO.

My View of "Hits"



Evolving Tracks

- Wiki page at SLAC:
  - http://confluence.slac.stanford.edu/display/ilc/Evolving+the+LCIO+Track+Class
- Direct link to my list of desired features:
  - http://confluence.slac.stanford.edu/display/ilc/From+Rob+Kutschke
- Announcement in the Forum:
  - Analysis and Reconstruction
  - Tracking and Vertexing
  - http://forum.linearcollider.org/index.php?t=tree&goto=1092&rid=0&S=d7fbd8520f51b76a739748d70e9a781f#msg_1092