LCD
Full Simulation & Reconstruction

Norman Graf
(SLAC)

Chicago January 7, 2002
Linear Collider Environment

- Detectors designed to exploit the physics discovery potential of $e^+e^-$ collisions at $\sqrt{s} \sim 1$TeV.

- Will perform precision measurements of complex final states.

- Require:
  - Exceptional momentum resolution
  - Excellent vertexing capabilities
  - “Energy Flow” calorimetry
  - Hermeticity
Mission Statement

- Provide full simulation capabilities for Linear Collider physics program:
  - Physics simulations
  - Detector designs
  - Reconstruction and analysis

- Need flexibility for:
  - New detector geometries/technologies
  - Different reconstruction algorithms

- Limited resources demand efficient solutions, focused effort.
Recent Focus

- Efforts devoted to providing physics analysis package for Snowmass meeting.
- Emphasis on integrated environment for event generation and fast detector simulation.
- Provided CD-ROM with precompiled executables (Windows and Linux), API, tutorials and example code.
- Less work on full reconstruction.
Plans

- Concentrate on GEANT4 full detector simulation.
- Emphasize results from full reconstruction.
- Include machine backgrounds.
- Iterate detector design based on above.
GEANT 4

- We have begun the transition to GEANT4 by incorporating the existing XML geometry-parsing code.
- Have defined generic hit classes for sensitive tracker and calorimeter hits.
- Prefer a common G4 XML-based solution.
Towards Internationalization

- Suggest that Tesla, NLC and JLC full simulation groups could run a single GEANT4 executable.
- Geometry determined at run-time (XML).
- Write out common “ideal” hits (~flat-file).
- Digitize as appropriate with plug-ins.
- Enormous savings in effort.
- Makes comparisons easy.
- Prefer common GEANT4 solution.
Full Simulations

LCD Full Sim
GISMO
C++

BRAHMS
GEANT3
FORTRAN

JIM
GEANT3
FORTRAN

Common GEANT4 executable
XML-based geometry
Generic Hit output
Analysis Frameworks

- Inclusive environment; support both ROOT/C++ and JAS/Java frameworks.
- Jet-Finding algorithms, invariant mass calculations, vertexing (based on SLD ZVTOP), flavor tagging, etc. are all available.
- Displays for event visualization.
- Histograms, fitting, etc.
LCDROOT Event Display
Wired LCD Event Display

[Image of a computer screen showing a visualization of a physics experiment with various data selection options on the left and a graphical representation of event data in the center.]
Topological Vertex Finder

- SLD’s unique topological vertex finder + mass tag
  - D.J. Jackson NIM A388, 247 (1997)
  - SLD collaboration PRL 80, 660 (1998)

\[ c \rightarrow D \]

\[ b \rightarrow B \rightarrow D \]
Encourage participation

- Tools are in place for many analyses.
- Time may be right for a mock data challenge.
- Will stress current reconstruction, but spurs creativity.

Upcoming Meetings

- ECFA/DESY Meeting, St. Malo, April.
- N. American meeting May/June?
- LCWS 2002 Chengdu Korea, August.
Data Generation and Analysis

- Generate MC events with full complement of backgrounds.
- Include beam structure, overlap events.
- Include machine backgrounds.
- Write out only detector hits; no MC info.
- Force development of reconstruction algorithms.
- Uncover “signal”. 
JAS Remote Data Access

- GUI
- TCP/IP Network
- Data Analysis Engine
- Experiment Extensions (Event Display)
- Java Compiler + Debugger
- User’s Java Code
- Padded Cell
- DATA
Distributed Analysis

- Catalog Server
- Control Server
- Client

Server to Server connections.
Summary

- The LCD group has developed a very dynamic detector design, data reconstruction and physics analysis simulation environment.

- This talk has only scratched the surface on a large body of work conducted by a remarkably small community of dedicated individuals.

- Much work remains to be done and we could all benefit from collaborative effort.

- Need to get serious about detector design based on realistic backgrounds and analysis.
American Linear Collider Detector simulation efforts are documented at:

www-sldnt.slac.stanford.edu/nld

Binaries, full source, API, tutorials, etc.

Thanks to:

A. Johnson, G. Bower, R. Cassell, T. Abe,
M. Iwasaki, W. Walkowiak, M. Ronan, B. Schumm,
et al.

Mail to: Norman.Graf@slac.stanford.edu