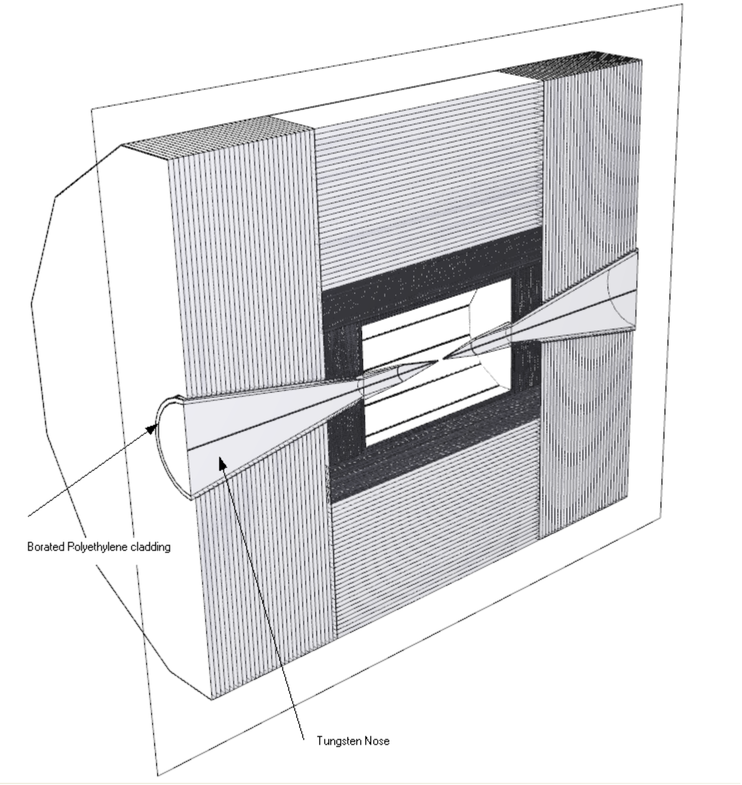
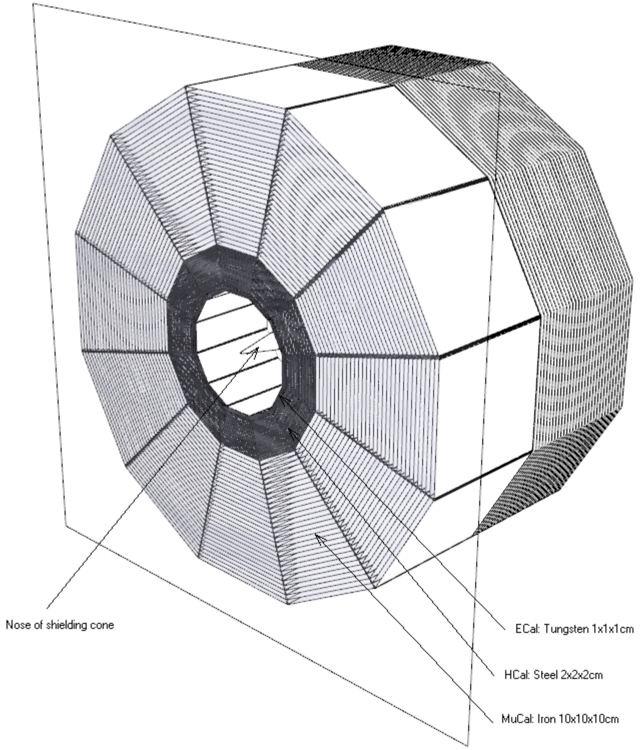
Getting Started with Muon Collider Detector Simulation

Norman Graf & Jeremy McCormick
SLAC





Status Report

- ✓ Background text files → stdhep
  - ✓ first pass done, creates weighted particles
  - ✓ Need to add code to generate additional particles to bring weights down to 1, ~factor of 28
- ✓ Detector model
  - ✓ First strawman detector mcd00 done
    - ✓ Silicon trackers, barrel plus end cap disks
    - ✓ Separate barrel and endcap calorimeters
    - ✓ Separate ECal, HCal, Muon systems
    - ✓ Reasonably segmented.
- ✓ Reconstruction
  - ✓ Where do people want to go?

LCIO Review

- SimCalorimeterHits
  - store energy depositions in ECal
  - summed for each cell → one hit / cell / event
  - list of particles that contributed w/ Edep + time
- SimTrackerHits
  - position, time, momentum, eDep, path length, and pointer to MCParticle
- MCParticle
  - energy, start point, end point, PDG, etc. of MC Particles in the detector simulation
  - Only primaries are persisted to LCIO because of high number of secondaries and background particles

Tools

Basics

- **Linux**, **OSX** or Windows (with cygwin) machine
- **cvs**
- **Maven 2**
- **Java** SDK 1.5 or greater
- **wget** (not necessary but very handy)
- **Netbeans** or **Eclipse** (optional Java IDEs)

ILC Software

- **LCSim** - HEP/ILC Java code
- **SLIC** - Simulator for the Linear Collider

Muon Collider

- Can set up separate cvs module if desired.

Before You Start...

- Examples assume a bash shell in a Unix-like environment.
  - Linux, OSX, Cygwin
- Text in the light blue boxs should be typed or cut and pasted into a command terminal from your work directory, unless otherwise noted.

```
echo “Hello Muon Collider World!.”
```

- The “\” character is used to fit long single commands on one line and should work in your shell.

```
slic.sh –g myDetector.lcdd \
-r 99
```

- If you have any questions, **PLEASE ASK**.

Maven

- Project management, dependency management, automated build system, and other good stuff.
- Installation (the quick version).

```
cd /my/app/dir # specific to your machine
wget http://opensource.become.com/apache//maven/binaries/apache-maven-2.2.1-bin.tar.gz
tar –zxvf apache-maven-2.2.1-bin.tar.gz
cd apache-maven-2.2.1
export MAVEN_HOME=`pwd`
export PATH=$MAVEN_HOME/bin:$PATH
mvn
```

- Now go get a cup of coffee (and drink it) while Maven bootstraps.
- Add $MAVEN_HOME/bin to your PATH in your .bash_profile if you want it to be setup when you login.

Getting Started

- Create a working directory.

```
cd /someplace
mkdir work
cd work
```

- Set the CVS location.

```
export CVSROOT=:pserver:anonymous@cvs.freehep.org:/cvs/lcd
```

- Is Java installed? (not covered here)

```
javac -version
```

should be 1.5 or greater

- Is Maven installed?

```
mvn -version
```

should be 2.2.1 NOT 3

Installing Simulation Tools

- Follow the SimDist Build Instructions to create a binary on your platform.
- SLAC users with NFS access can use this script.

```
/nfs/slac/g/lcd/mc/prj/sw/dist/SimDist/pro/scripts/slic.sh
```

- Platform binaries can be found in the SLIC Dist Directory. (Sometimes these are even up-to-date.)
- If you want to change various options (the Geant4 version comes to mind), please talk to me.

(It will take a long time to build this, so should we do something else now?)

LCD Detectors Project

- Need to checkout and build LCDetectors to get an example ECal detector setup.

```
cvs co LCDetectors
```

- Technically, building this package is not required to run the simulation, but a local installation will be needed later for new detector development, recon and analysis work.

Simulating Events

- Get some events first (this is all one line).

```
wget \
ftp://ftp-lcd.slac.stanford.edu/ilc3/MUC/backgrounds/stdhep/excl-1to25m-mumi-0-100.stdhep
```

- This example command will simulate events in SLIC (Don't be scared...all will be explained).

```
1  ./SimDist/scripts/slic.sh \
2  -g ./LCDetectors/detectors/mcd00/mcd00.lcdd \
3  -i excl-1to25m-mumi-0-100.stdhep \
4  -x \
5  -o muCollTest \
6  -r 10000
```

1. Path to SLIC run script in your SimDist installation
2. LCDD geometry file from LCDetectors
3. StdHep file downloaded from FTP
4. Will clobber existing output file if it already exists
5. Name of output file (.slcio extension added automatically)
6. Number of events to run

Installing LCSim

- LCSim is a large collection of Java code that we will use as the reconstruction & analysis framework.
- To create a local installation of LCSim from scratch, you can follow the Detailed LCSim Build Instructions.
- You may also download a released jar.

```
wget http://www.lcsim.org/maven2/org/lcsim/lcsim/1.18-SNAPSHOT/lcsim-1.18-SNAPSHOT-bin.jar
```

- Usually, you would install LCSim locally by building it yourself, but this will make sure we're all using the exact same jar file. And additions/corrections can also be quickly pushed out to the web location as we work.

Installing JAS

- Download JAS3 from:

  http://jas.freehep.org/jas3/download.html

- Install Plugins, following instructions at:

  https://confluence.slac.stanford.edu/display/ilc/Installing+JAS3

First look at Events

- Fetch some single particle events:

```
wget \
ftp://ftp-lcd.slac.stanford.edu/ilc3/MUC/backgrounds/slcio/slic/muon_Theta90_10GeV_SLIC-v2r9p8_geant4-v9r3p2_QGSP_BERT_mcd00.slcio
```

```
wget \
ftp://ftp-lcd.slac.stanford.edu/ilc3/MUC/backgrounds/slcio/slic/pi_Theta90_10GeV_SLIC-v2r9p8_geant4-v9r3p2_QGSP_BERT_mcd00.slcio
```

Or some background events:

```
wget \
ftp://ftp-lcd.slac.stanford.edu/ilc3/MUC/backgrounds/slcio/slic/excl-1to25m-mumi_0_10000_SLIC-v2r9p8_geant4-v9r3p2_QGSP_BERT_mcd00.slcio
```