Software Tools for LC Simulation Studies

-- Recent Developments --

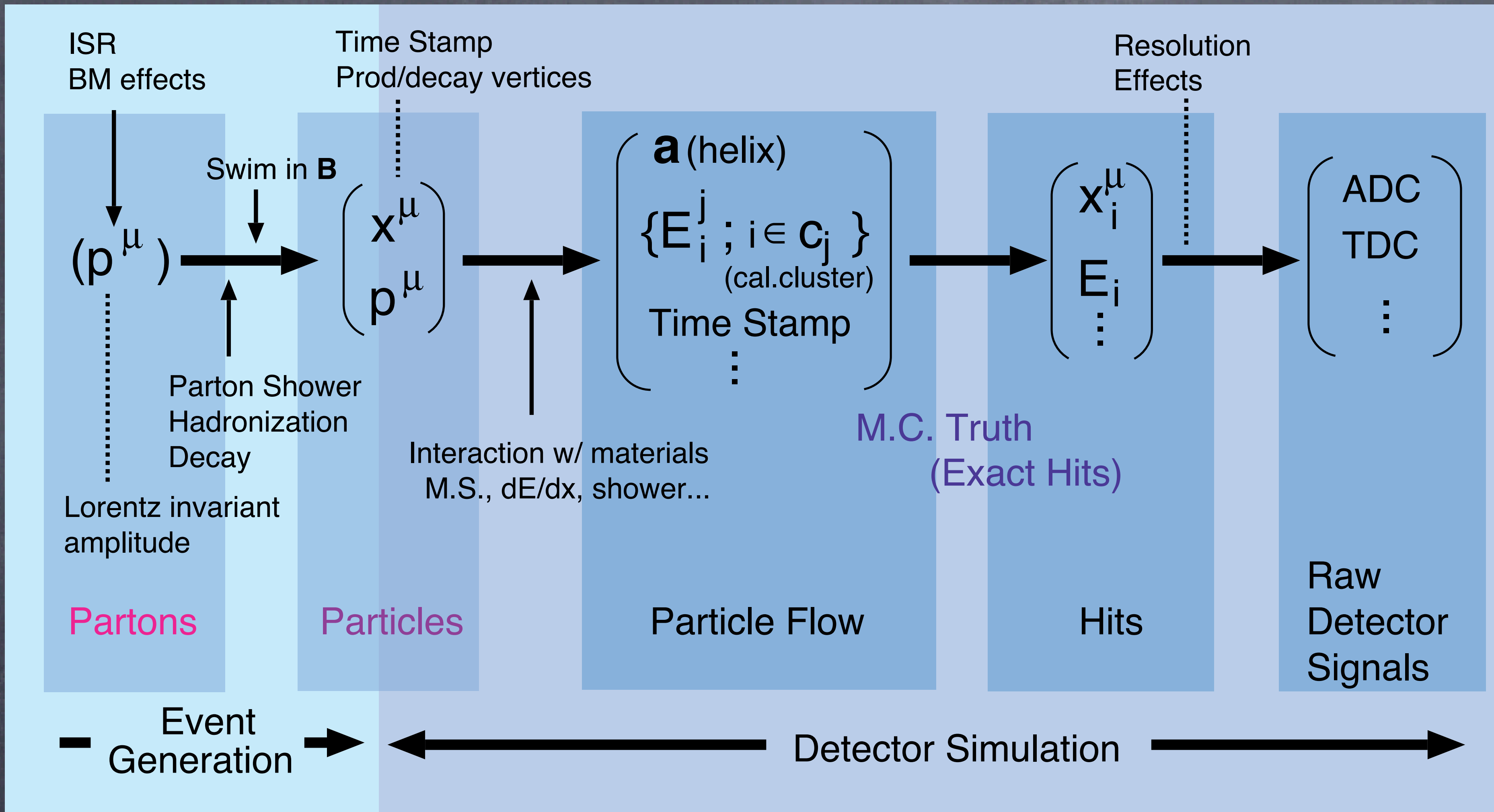
Keisuke Fujii, KEK

Representing ACFA-Sim Group
http://acfahep.kek.jp/subg/sim/

Overview

Flow Chart of Simulation

K.Fujii @ ACFA LCWS2003



Physsim
(HELAS/BASES/SPRING)

LCGrace
(HEPRUP/HEPEUP interface)

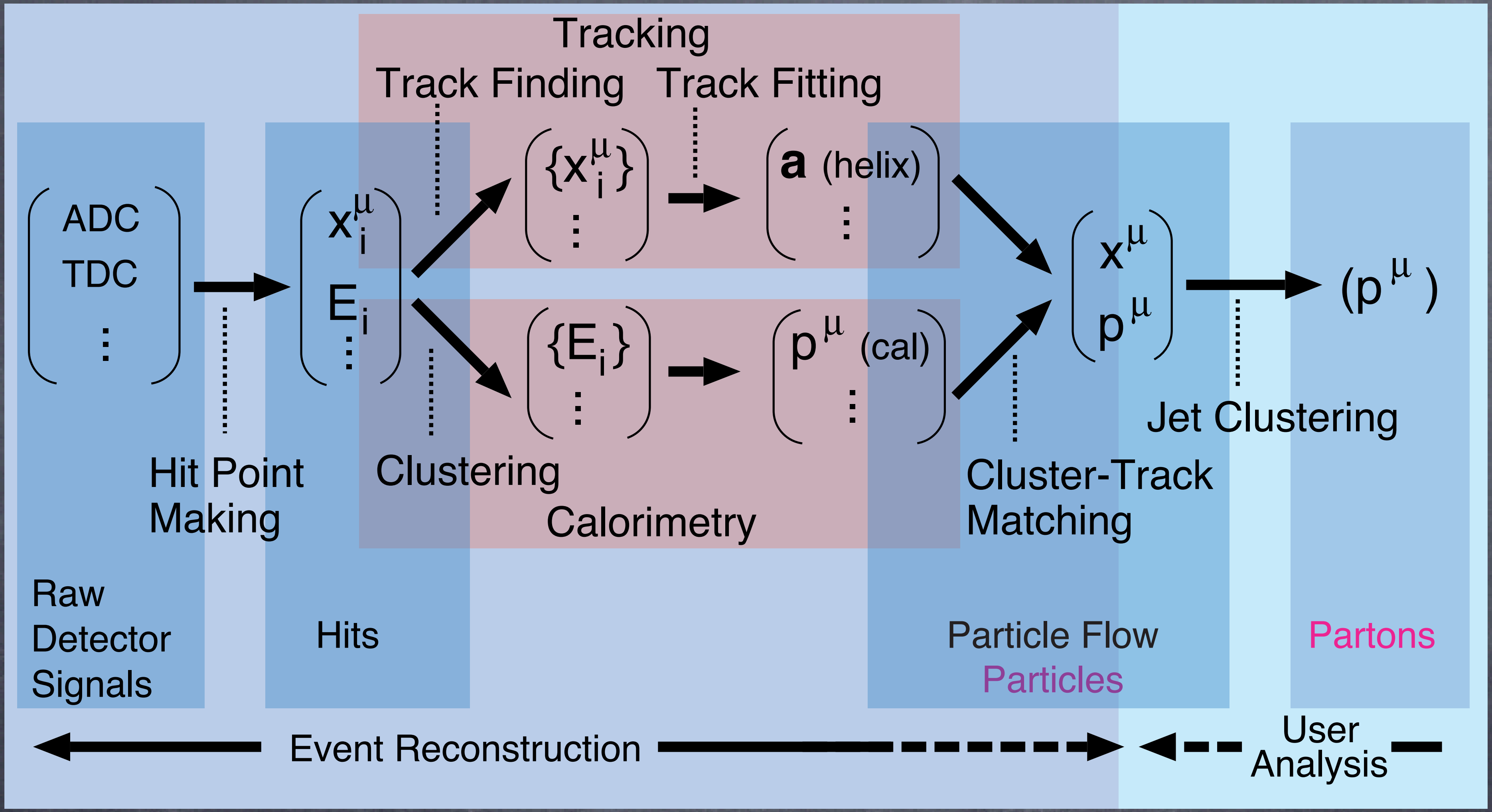
Pythia / Herwig

— QuickSim (LCLIB) ——……………………

—— JLCSIM (Geant3) ——……………

—— Jupiter/Satellites (Geant4) ——

Flow Chart of Analysis



QuickSim (LCLIB)

JLCSIM (Geant3)

Satellites

Physsim/Anlib

(Jet Finder/4-vector Manip.)

ZVTOP, JETNET

(Flavor Tagger)

JSF: the Flow Controller

All the packages work under the JSF environment

- Based on ROOT: C++
- Modular --> Unified Framework for
  - Event generation
  - Detector Simulation
  - Event Reconstruction
  - Physics Analysis
  - Beam Test (Online/Offline)
- Object I/O
  - Each module's data in a ROOT tree
  - User data (hists., ntuples, job params.) also in the same tree
- Unified User Interface (Batch/Interactive)
  - GUI/CUI (user definable command line args. & default vals.)
  - Simple Built-in Event Display

Package Availability

General Information from ACFA-Sim Home Page
http://acfahep.kek.jp/subg/sim/softs.html
Latest Packages from CVS on jlccvs.kek.jp
http://jlccvs.kek.jp/

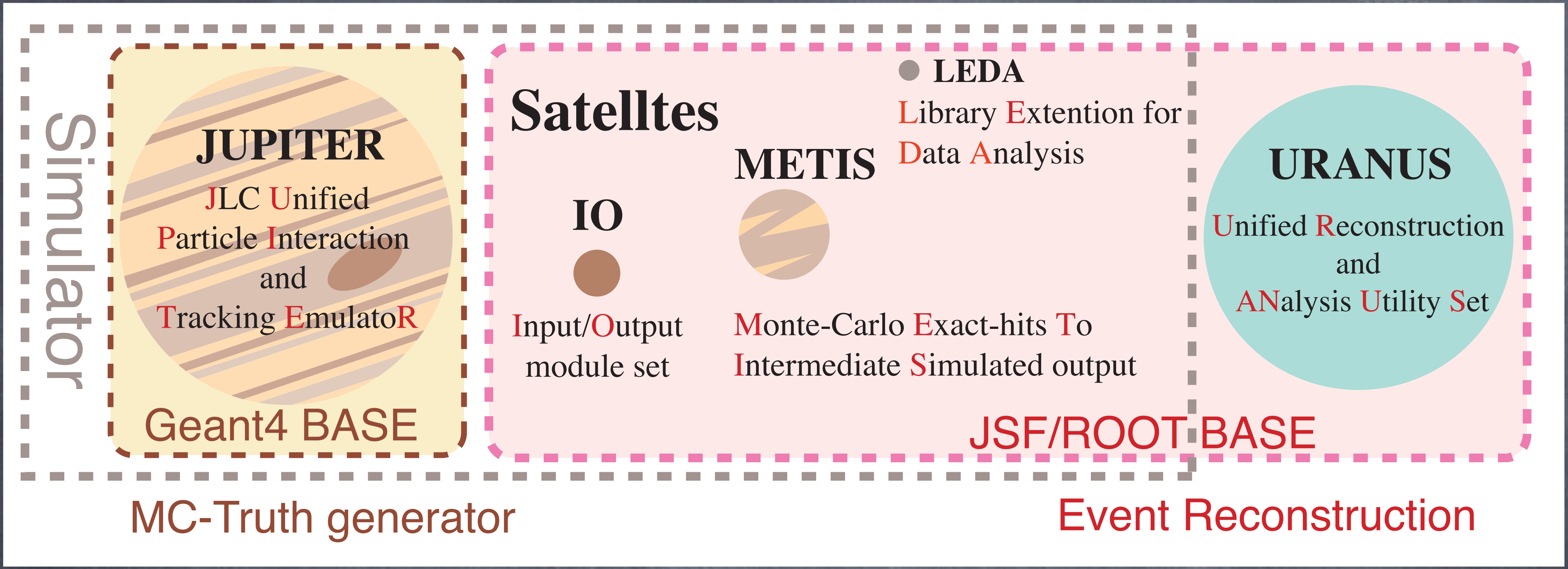
- LCLIB (Old Fortran Library for QuickSim)
- JLCSIM (Geant3-based Full Detector Simulator)
- JSF includes the following subpackages
  - Generator Interface to Pythia, Physsim, LCGrace, ...
  - Hadronizer (Pythia)
  - QuickSim (C++ wrapper for LCLIB)
  - C++ Version of Bases/Spring
  - GUI, Event Display, ZVTOP, JETNET, examples ....
- Physsim (Event Generator, Anlib, examples)
- Jupiter (Geant4-based Full Detector Simulator)
- Satellites (in preparation)

Jupiter and Its Satellites

Design Concepts

K.Fujii @ ACFA LCWS2003

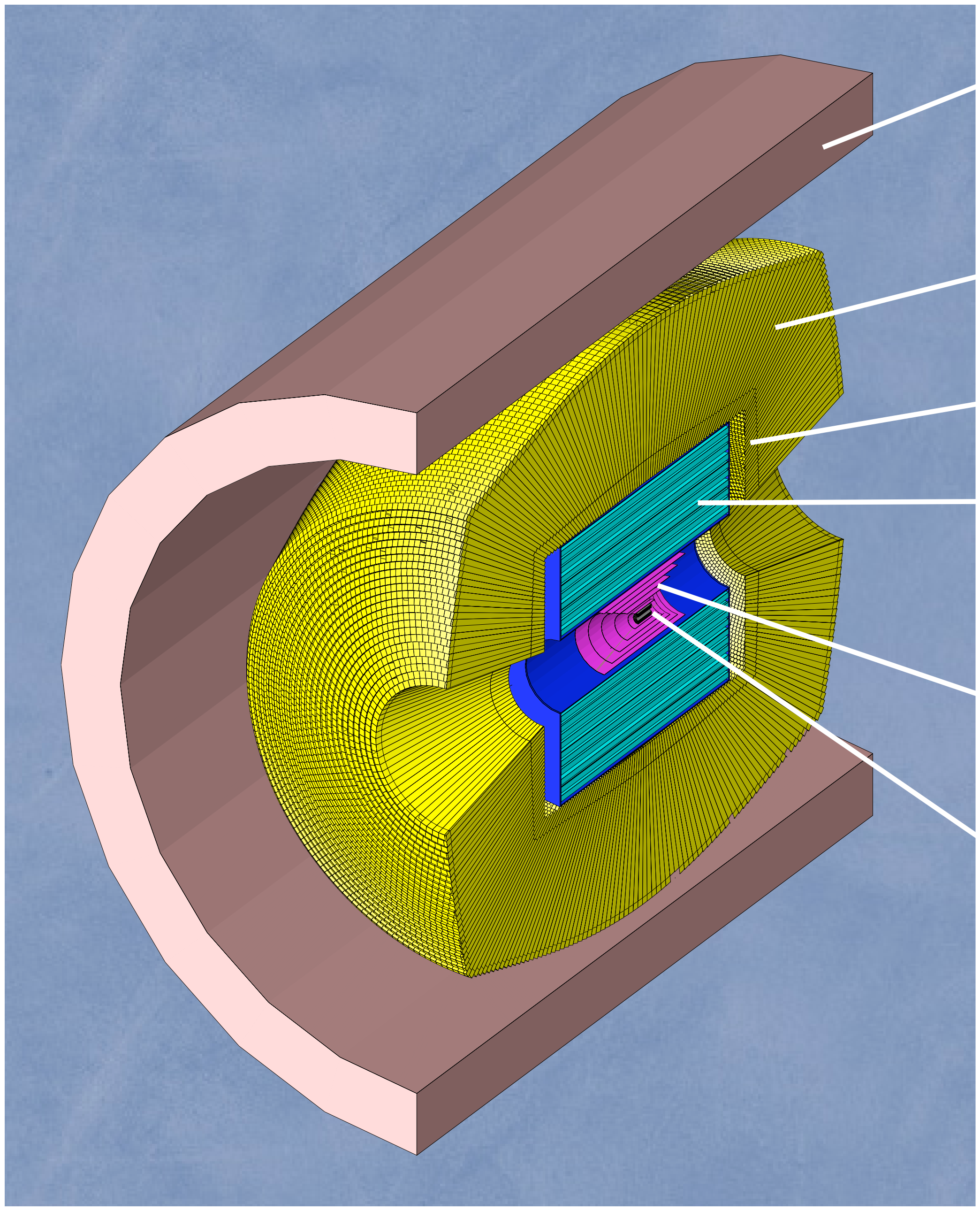
Core Developers: K.Hoshina & K.F.



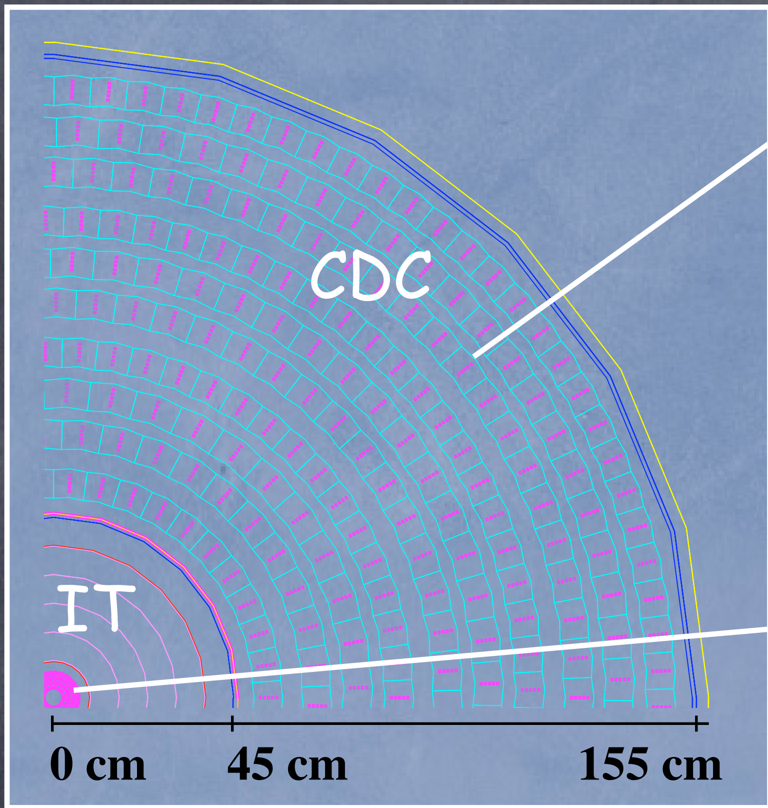
- Modular --> allows simultaneous code development by many detector subgroups
- Powerful Base Classes (that provide Unified Interface)
  - to facilitate easy (un)installation of components
  - to help implement detailed hierarchical structures
  - to minimize user-written source code by
    - automatic naming system / material management
    - B-field compositions for accelerator components

Current Status of Jupiter (3T)

K.Fujii @ ACFA LCWS2003

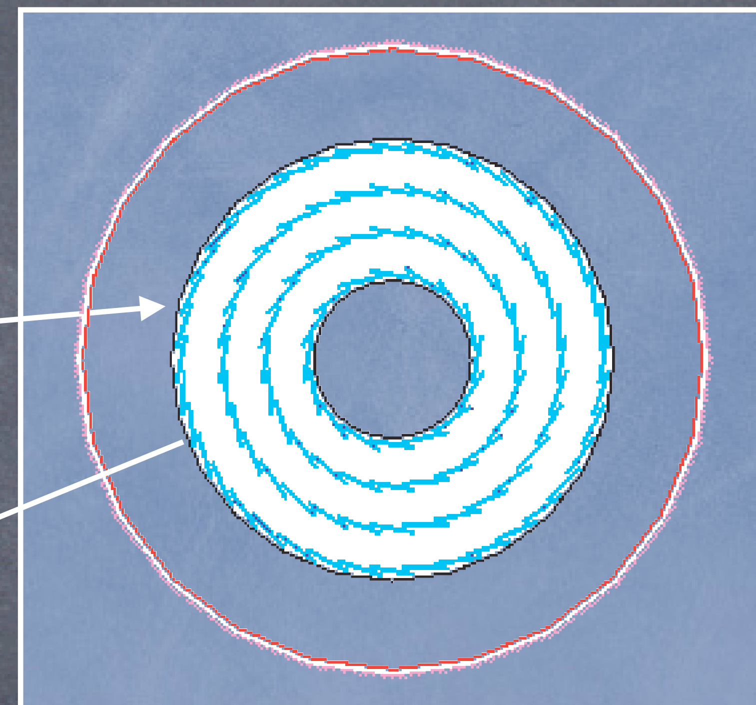


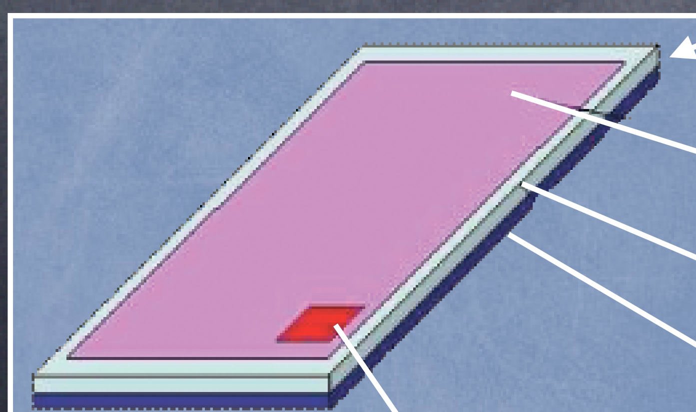
Examples of Details Implemented



Individual drift cells and wires

VTX Detector (T.Aso)





Pixel Area

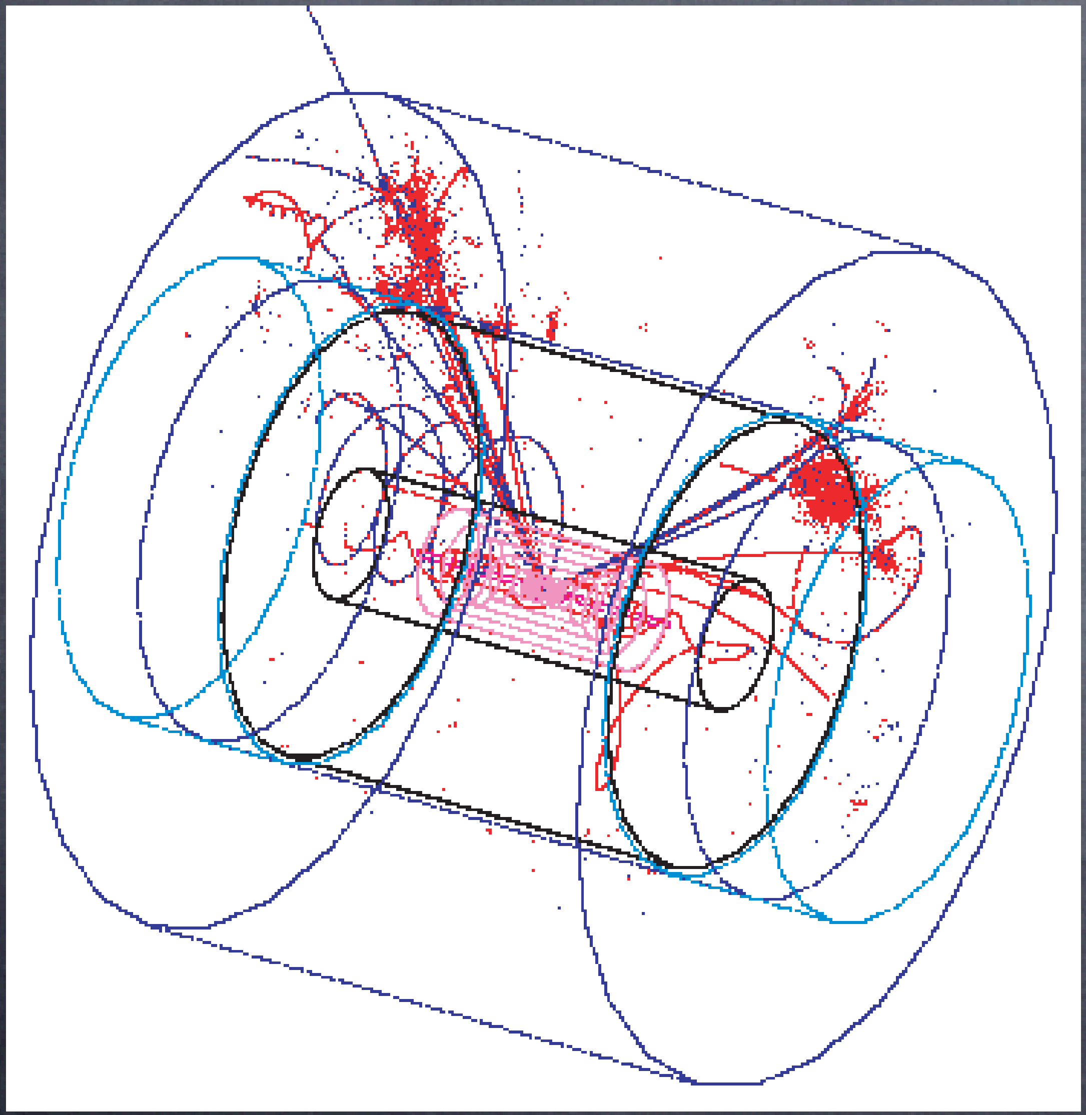
Epitaxial

Substrate

Sensor

Individual Pixel

A Typical Event Simulated Through Jupiter



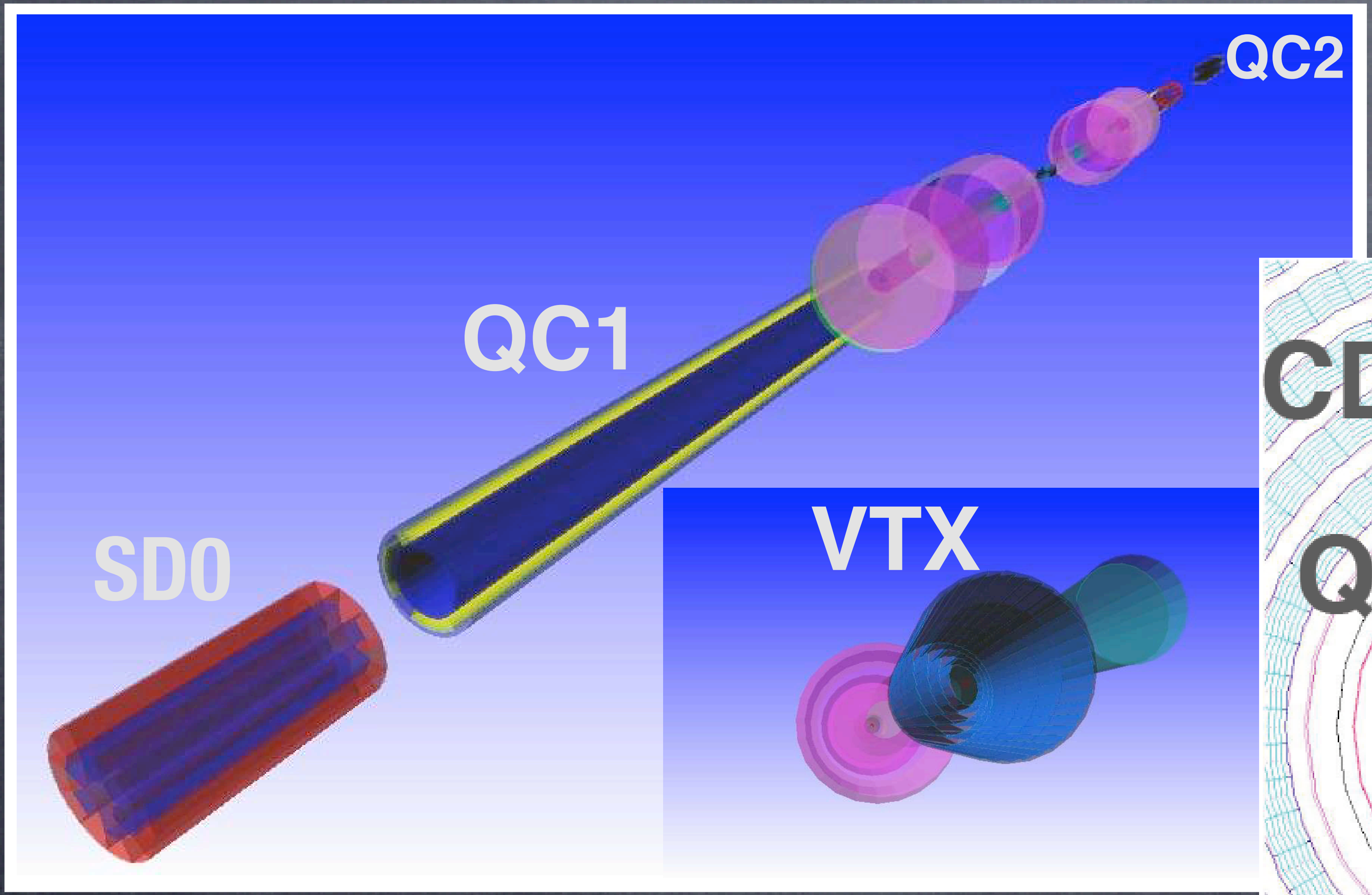
$$e^+e^- \rightarrow Z^0H^0$$

followed by

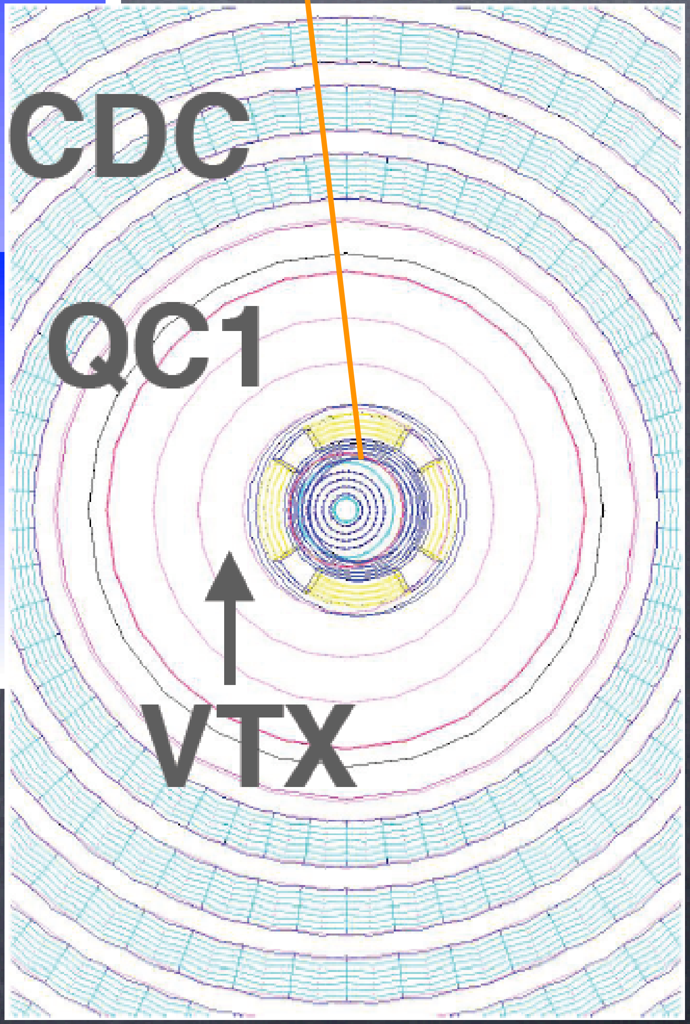
$$Z^0 \rightarrow \nu\bar{\nu}$$

$$\sqrt{s} = 350 \text{ GeV}$$

Beam Delivery System for Beam BG Study



Crossing Angle
=3mrad

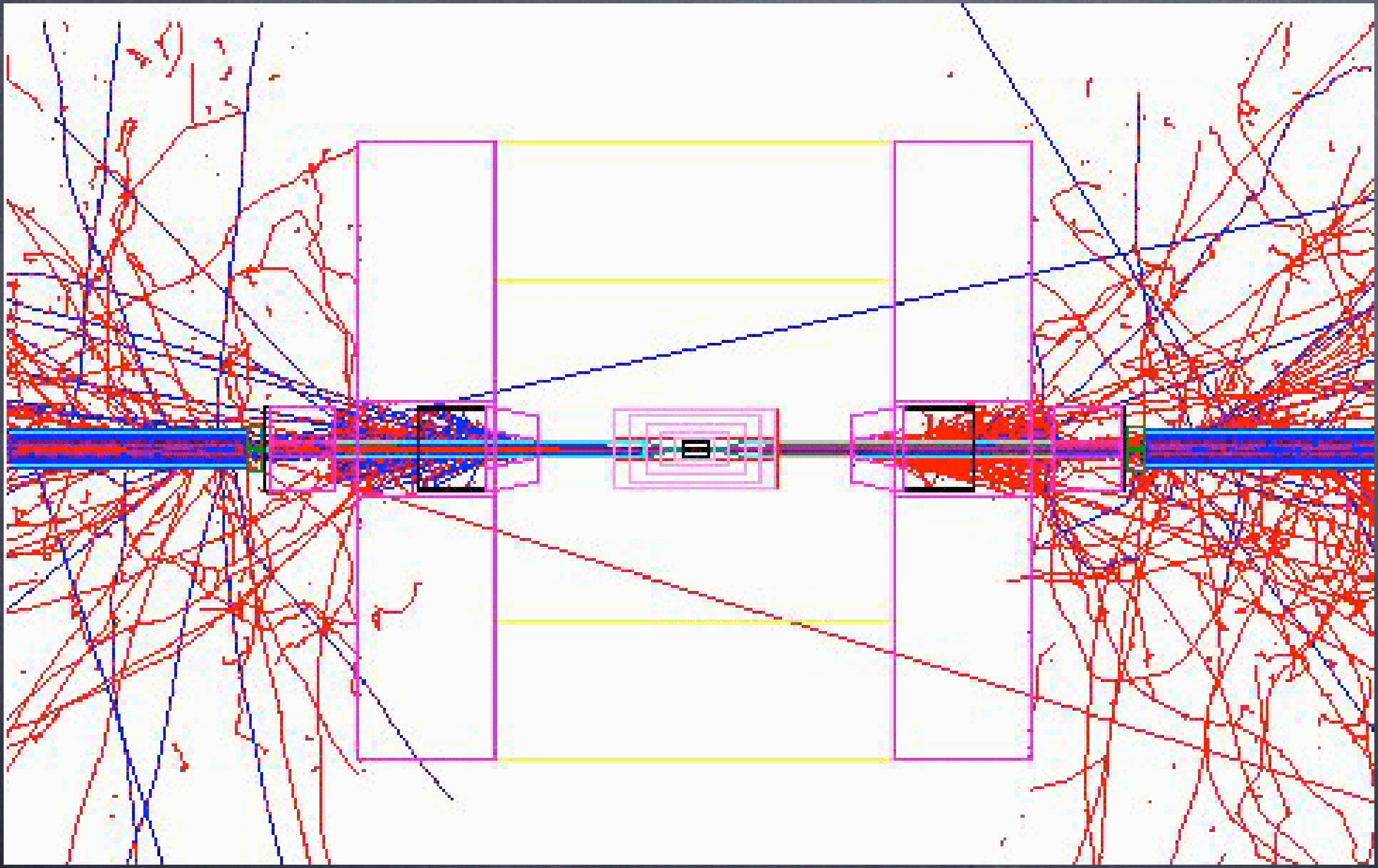


T.Aso

L* = 4.3m
3T Solenoid

Beam BG Simulated by Jupiter

T.Aso



Event Source: Generated by Cain

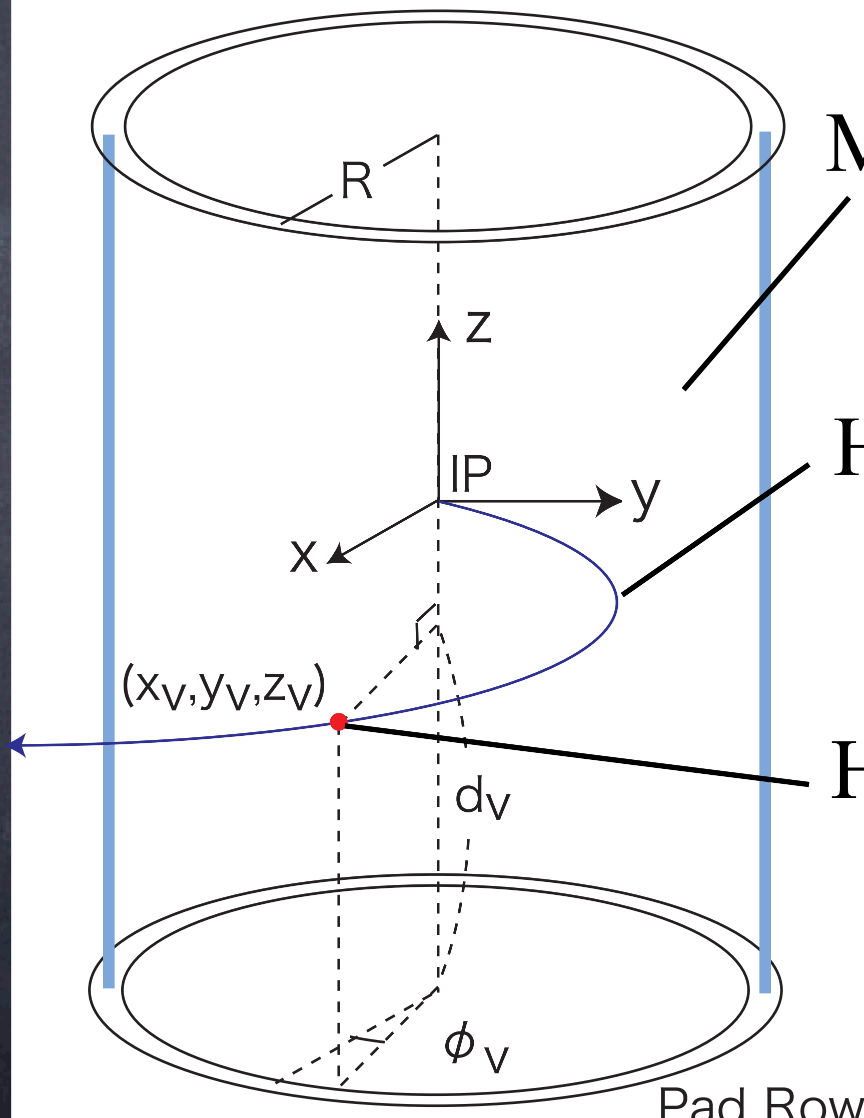
Kalman Filter Library

Kalman Filter Library Features

-
- KalLib: base classes that implement algorithm
  - TVKalSystem, TVKalSite, TVKalState
- KalTrackLib: that implements pure virtuals of KalLib
- GeomLib: geometry classes that provide
  - track models (helix, straight line, ...)
  - surfaces (cylinder, hyperboloid, flat plane, ...)
  -
- Minimum number of user-implemented classes
  - MeasLayer : measurement layer
  - KalDetector : an array containing MeasLayers
    - You can put different kinds of MeasLayers
  - Hit : coordinate vector as defined by the MeasLayer
- Track model can change site to site which allows B-field variation along a particle trajectory

Example of Detector Implementation

How to define TPC



Measurement Layer

$$S(x,y,z) = x^2 + y^2 - R^2 = 0$$

Helix

$$\begin{cases} x = x_0 + d_\rho \cos\phi_0 + \frac{\alpha}{\kappa}(\cos\phi_0 - \cos(\phi_o + \phi)) \\ y = y_0 + d_\rho \sin\phi_0 + \frac{\alpha}{\kappa}(\sin\phi_0 - \sin(\phi_o + \phi)) \\ z = z_0 + d_z - \frac{\alpha}{\kappa}\tan\lambda \cdot \phi \end{cases}$$

Hit

$$\begin{pmatrix} R \cdot \phi_v \\ d_v \end{pmatrix} = \begin{pmatrix} R \cdot \tan^{-1}\left(\frac{y_v}{x_v}\right) \\ L/2 \mp z_v \end{pmatrix}$$

EXMeasLayer for TPC

HitToXv

$$\begin{pmatrix} x_v \\ y_v \\ z_v \end{pmatrix} = \begin{pmatrix} R \cdot \cos\phi_v \\ R \cdot \sin\phi_v \\ \pm(L/2 - d_v) \end{pmatrix}$$

XvToMv

$$\begin{pmatrix} R \cdot \phi_v \\ d_v \end{pmatrix} = \begin{pmatrix} R \cdot \tan^{-1}\left(\frac{y_v}{x_v}\right) \\ L/2 \mp z_v \end{pmatrix}$$

CalcDhDa

Meas.Vector Derivative w.r.t. Track Parameter Vector

$$\begin{pmatrix} \frac{\partial(R\cdot\phi_v)}{\partial \mathrm{a}} \\ \frac{\partial d_v}{\partial \mathrm{a}} \end{pmatrix} = \begin{pmatrix} -\frac{y_v}{R}\left(\frac{\partial x_v}{\partial \mathrm{a}}\right) + \frac{x_v}{R}\left(\frac{\partial y_v}{\partial \mathrm{a}}\right) \\ \mp \frac{\partial z_v}{\partial \mathrm{a}} \end{pmatrix}$$

$$\frac{\partial \mathrm{x}(\phi(\mathrm{a}),\mathrm{a})}{\partial \mathrm{a}} = \frac{\partial \mathrm{x}}{\partial \phi} \cdot \frac{\partial \phi}{\partial \mathrm{a}} + \frac{\partial \mathrm{x}}{\partial \mathrm{a}}$$

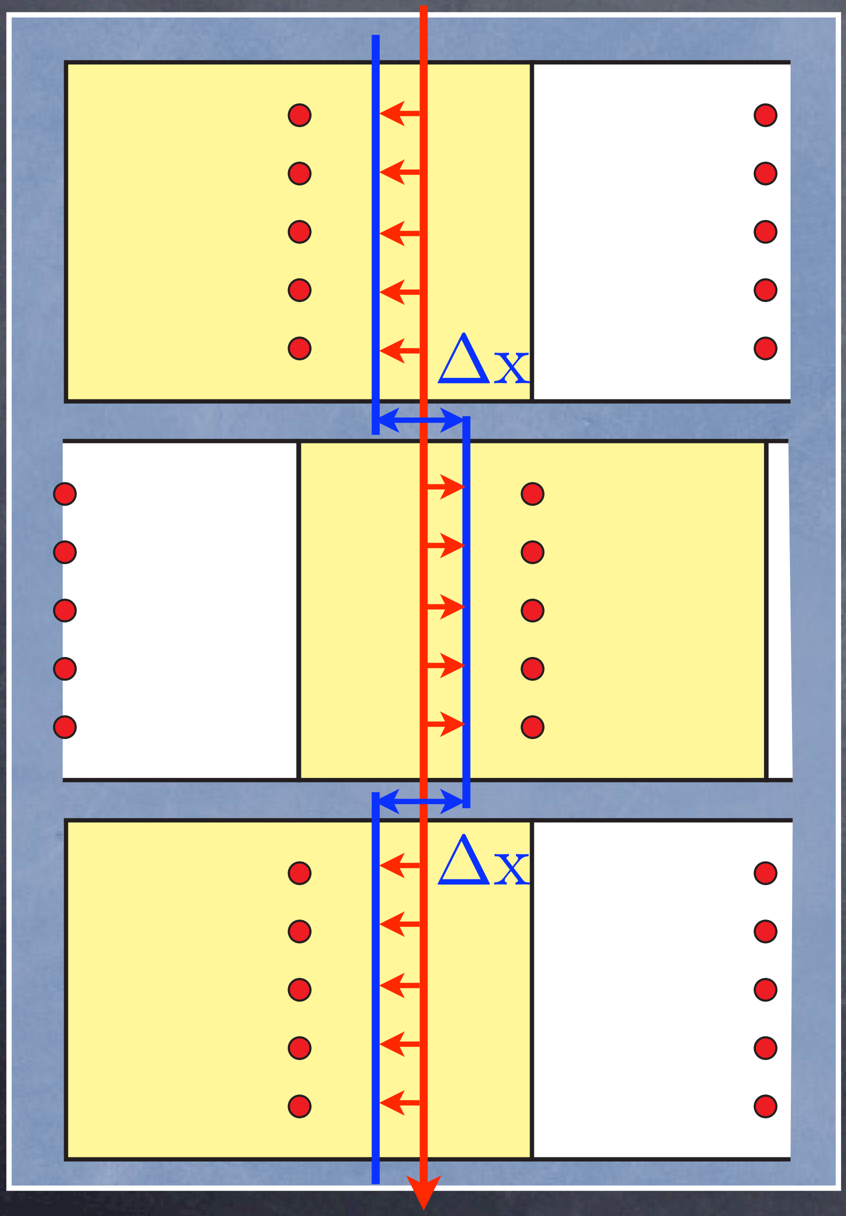
$$\frac{\partial \phi}{\partial \mathrm{a}} = -\frac{1}{\left(\frac{\partial S}{\partial \mathrm{X}} \cdot \frac{\partial \mathrm{X}}{\partial \phi}\right)} \frac{\partial S}{\partial \mathrm{x}} \cdot \frac{\partial \mathrm{x}}{\partial \mathrm{a}}$$

Kalman Filter Library now being imported into Satellites

Time Stamping

Application of Kalman Filter Library

In the Case of JLC-CDC



Staggered Cells

Wrong T0 breaks a track!

$$\Delta \mathrm{x} = 2\ v_{\mathrm{drift}} \times \Delta T_0$$

Naively we expect

$$\sigma_{\Delta T_0} \simeq \frac{\sigma_{\mathrm{xy}}}{v_{\mathrm{drift}}\sqrt{n}}$$

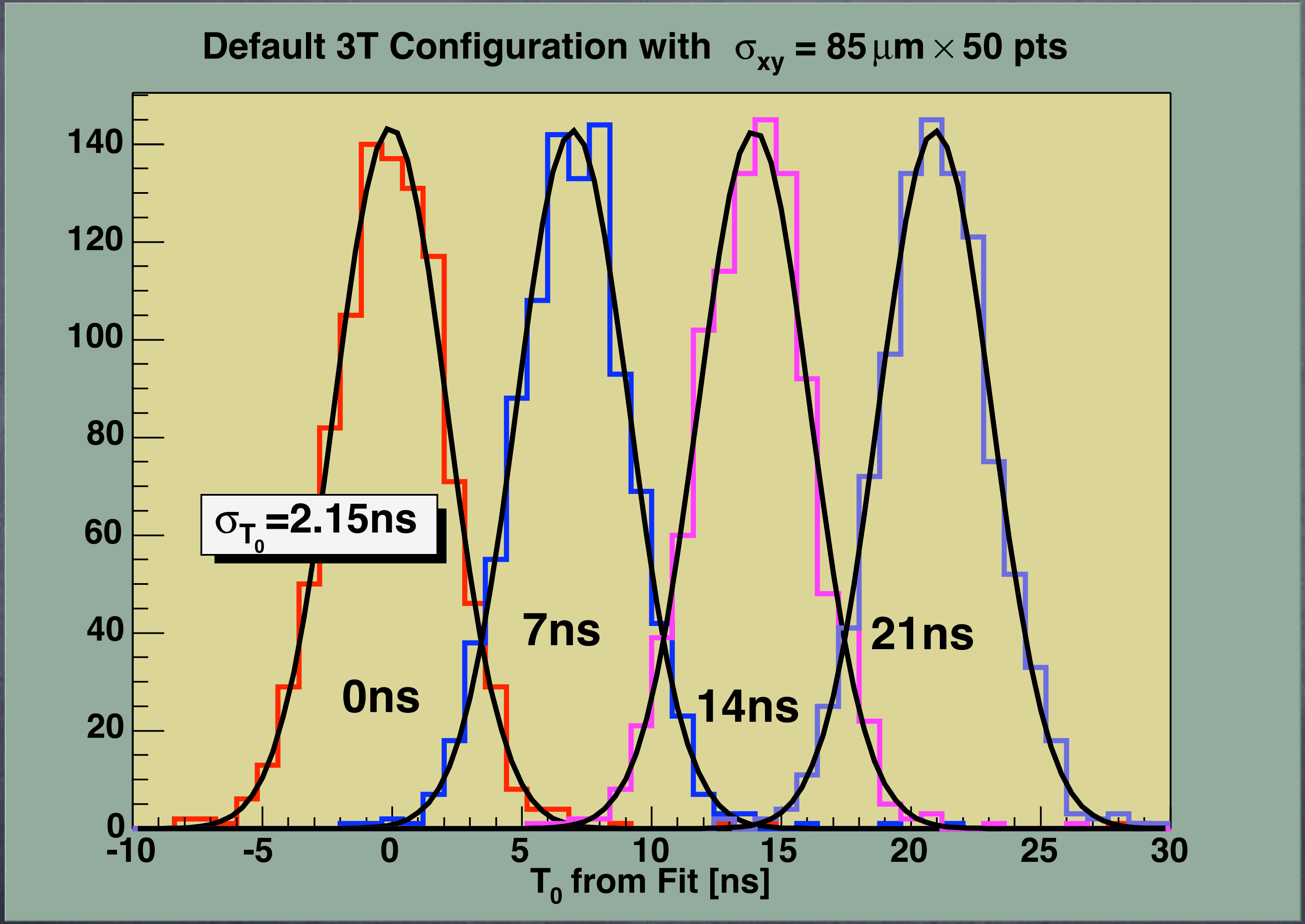
$$\sigma_{\mathrm{xy}} = 85\ \mu\mathrm{m}$$

$$v_{\mathrm{drift}} = 0.7\ \mathrm{cm}/\mu\mathrm{s}$$

$$n = 50$$

$$\sigma_{\Delta T_0} \simeq 1.7\ \mathrm{ns}$$

T0 from Helix Fit (axial+stereo, 100GeV)

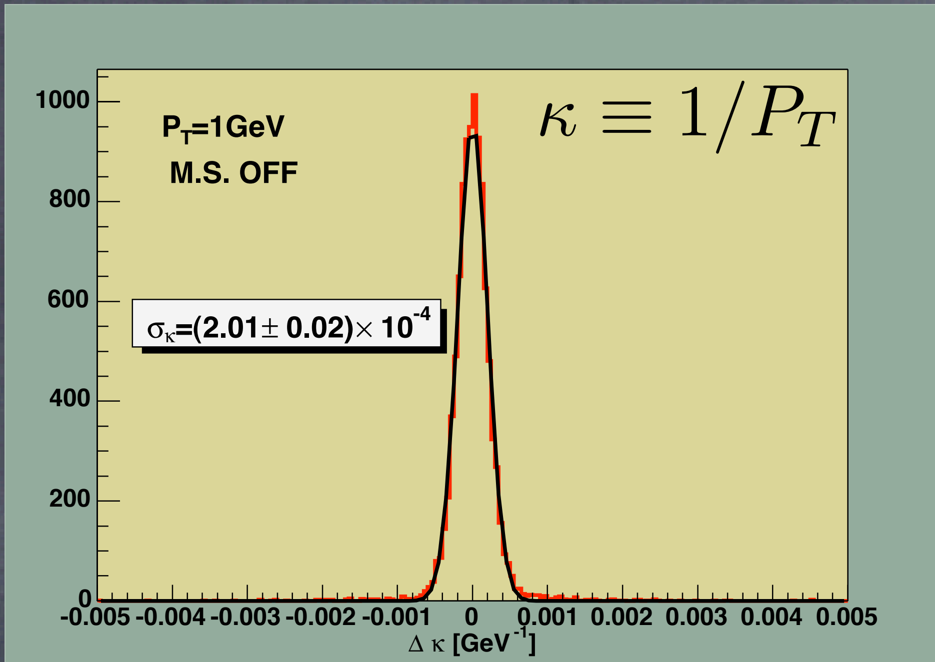


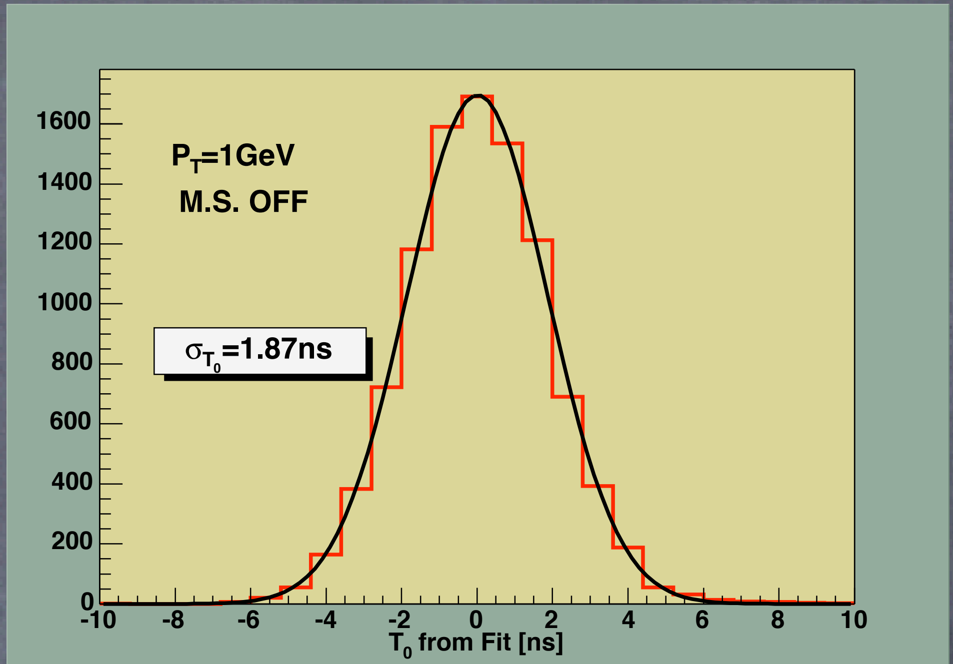
We can still determine T0 with ~2.2ns accuracy!

K.Fujii @ ACFA LCWS2003

Multiple Scattering Effects (axial+stereo, 1GeV)

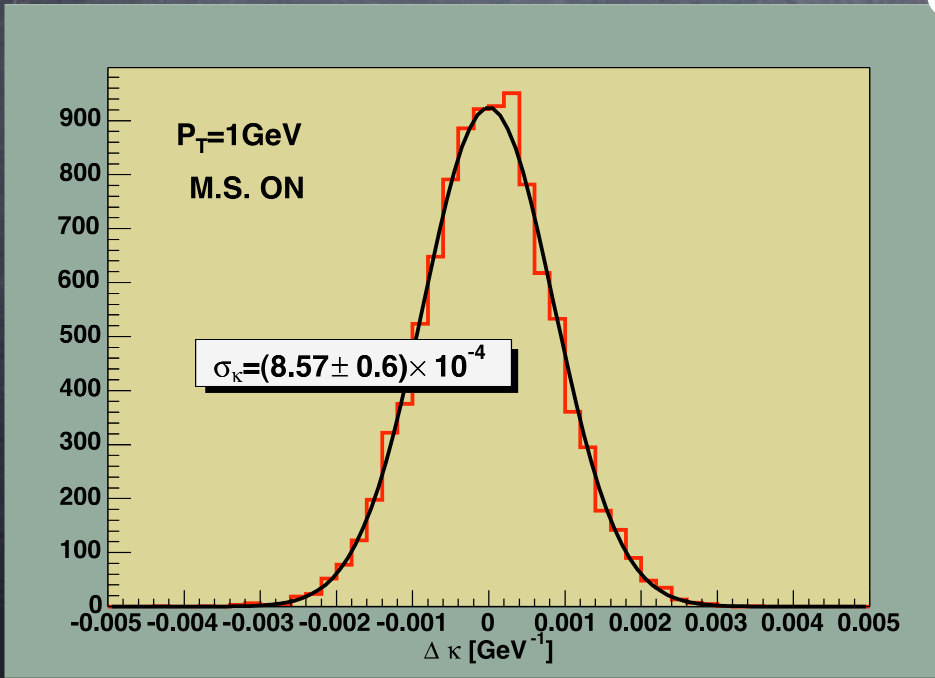
MS OFF

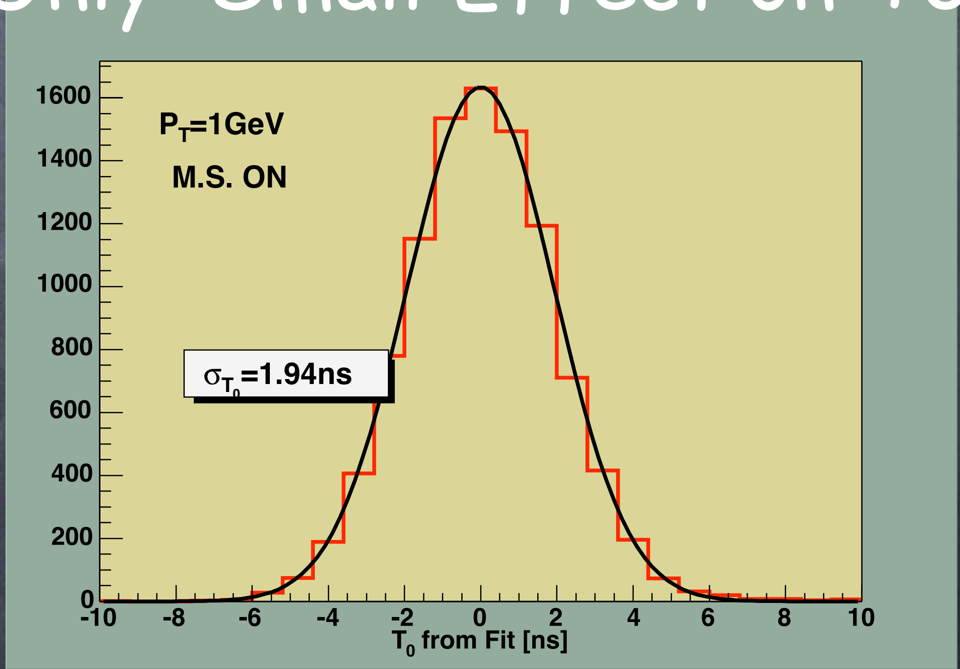




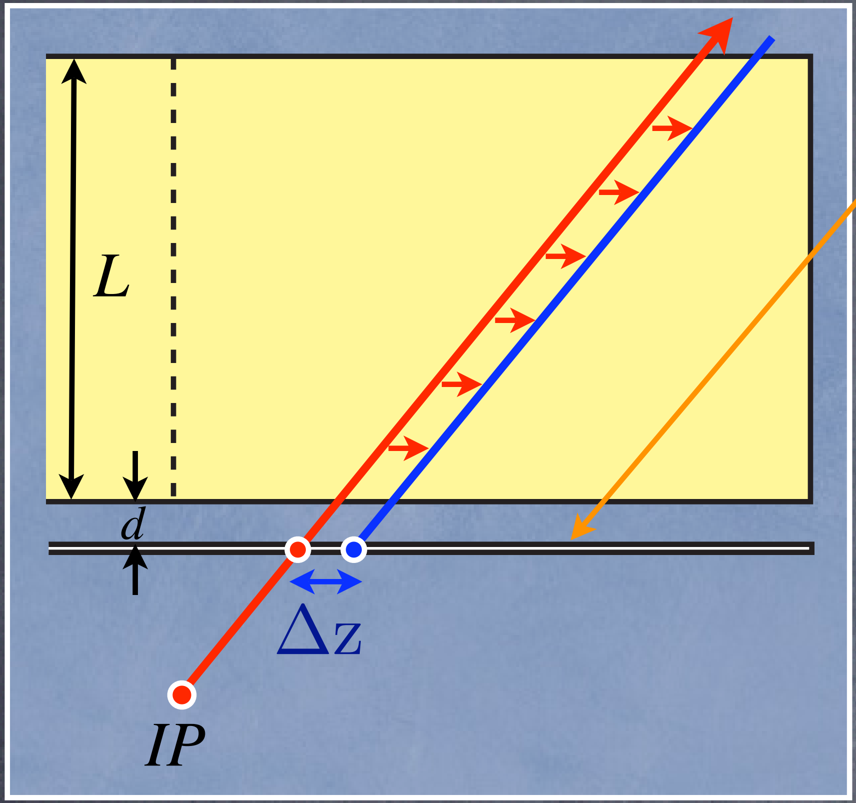
Only Small Effect on T0

MS ON





In the Case of TPC



External Z Detector (T0 Device)

Wrong T0 makes a Z-shift!

$$\Delta \mathrm{z} = v_{\mathrm{drift}} \times \Delta T_0$$

Naively we expect

$$\sigma_{\Delta T_0} \simeq \frac{2\sigma_{\mathrm{z}}}{v_{\mathrm{drift}}\sqrt{n}} \left[1 + 3\left(\frac{d}{L}\right) + 3\left(\frac{d}{L}\right)^2 \right]^{-\frac{1}{2}}$$

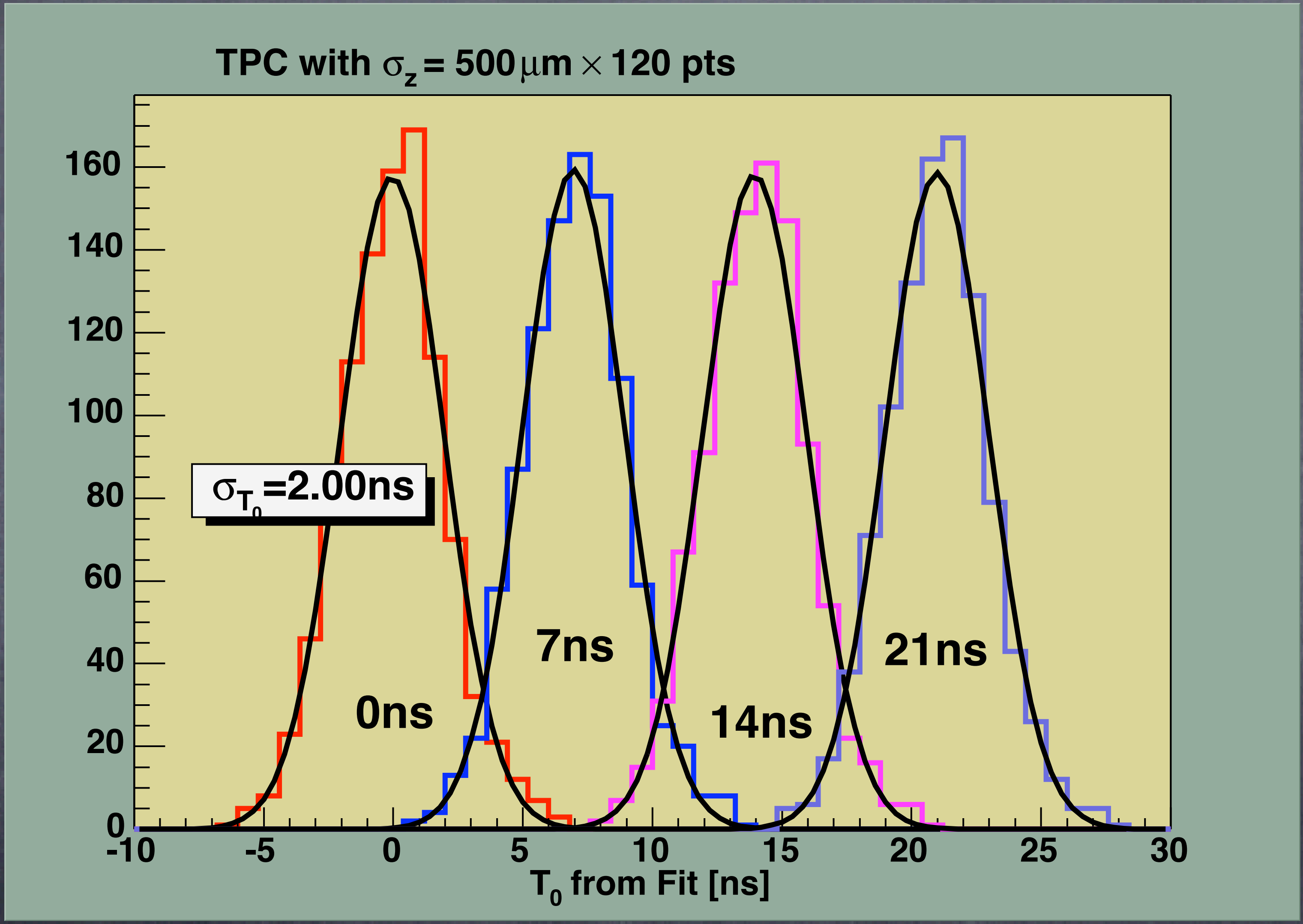
$$\simeq \frac{2\sigma_{\mathrm{z}}}{v_{\mathrm{drift}}\sqrt{n}} \quad \text{if} \quad \left(\frac{d}{L}\right) \ll 1$$

Assuming that Z resolution of the external detector is negligible

$$\sigma_{\mathrm{z}} = 500\ \mu\mathrm{m}$$
$$v_{\mathrm{drift}} = 5\ \mathrm{cm}/\mu\mathrm{s}$$
$$n = 120$$

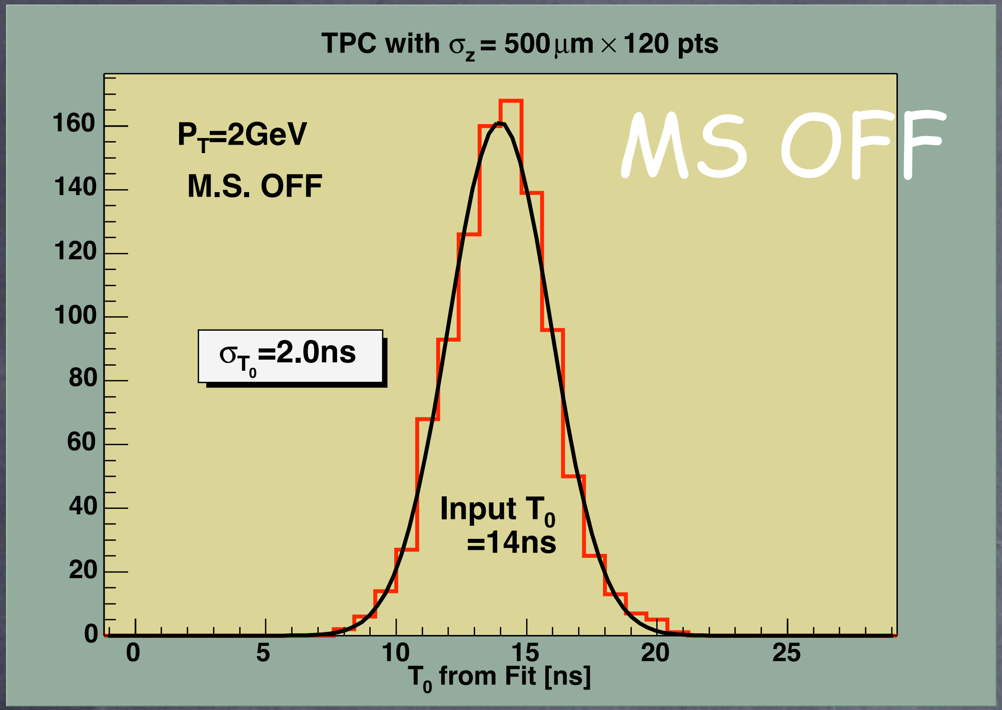
$$\sigma_{\Delta T_0} \simeq 2.0\ \mathrm{ns}$$

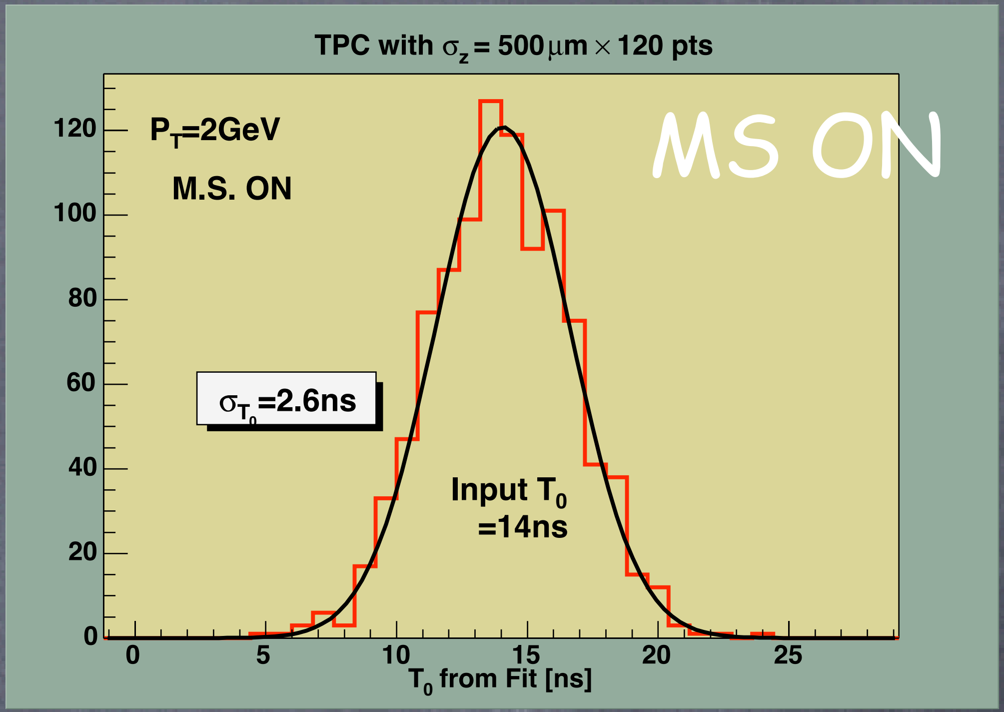
T0 from Helix Fit (d=5cm, 100GeV)



We can determine T0 with ~2.0ns accuracy as expected!

Multiple Scattering Effects (d=5cm,0.6%X0, 2GeV)





MS Effect more significant than for CDC

This is probably due to the fact that there is only a single break point to decide T0.

The material thickness between TPC and T0 detector does not matter as long as it stays just in front of the T0 detector.

0.6%X0 to 3.0%X0 --> 2% shift in T0 resolution

Summary and Future Plan

- Basic framework for full detector simulation based on Geant4 (Jupiter) + JSF/ROOT (Satellites) has been developed.
- Basic detector components have been implemented into Jupiter.
- Beam Delivery System has been implemented and used for beam BG studies.
- Kalman Filter Library has been developed and used for time stamping study.
-
- Future Plan for Jupiter/Satellites
  - Put Kalman filter library into Satellites framework
  - Prepare some sample code for Satellites
  - Release Satellites
  - Make them LCIO-compliant
  - XML-based I/O of detector geometry
  - ....