Software tools for JLC studies

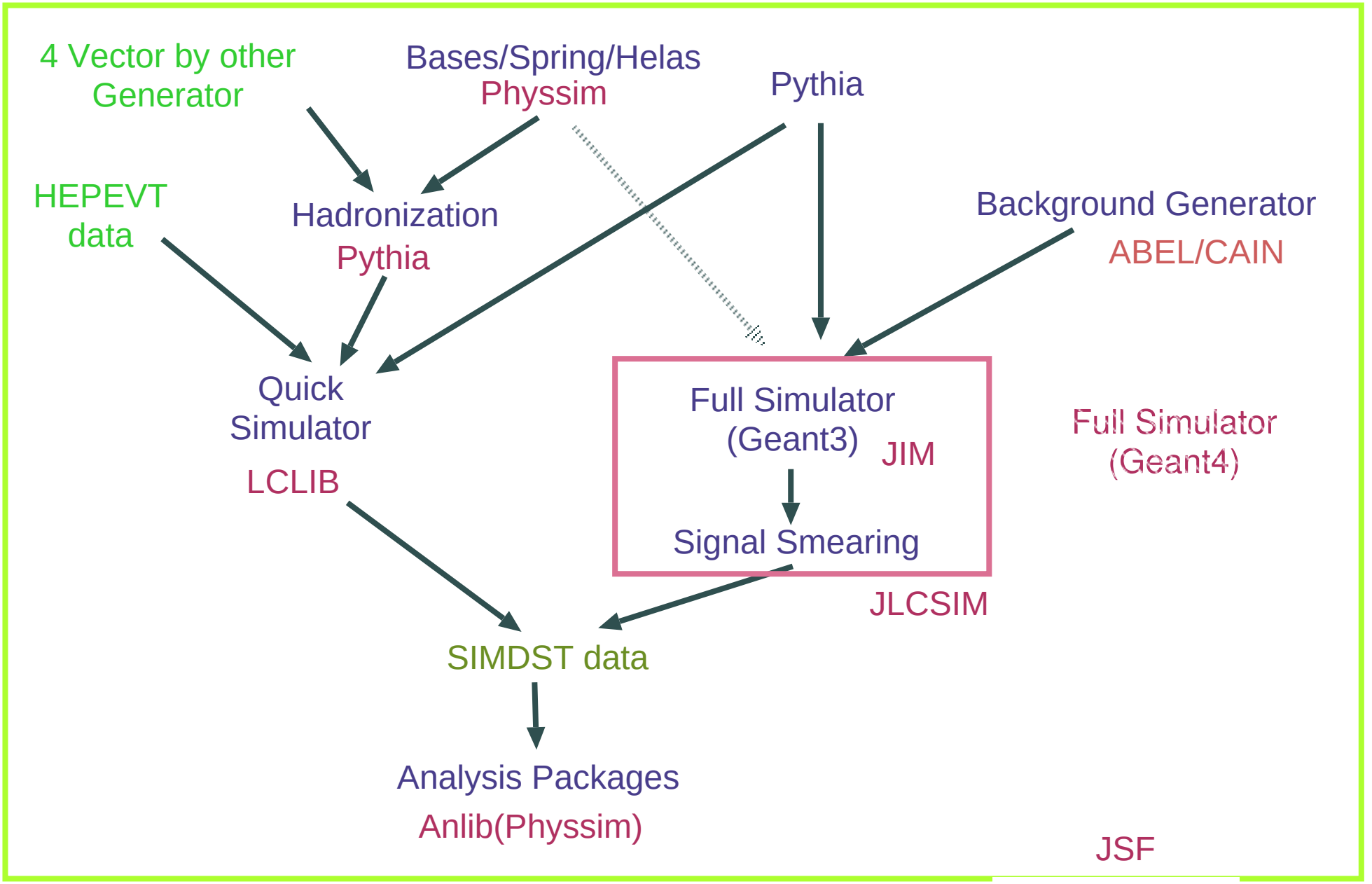
Akiya Miyamoto, KEK
25-October-2000

Contents

1. Software Overview
2. JLC Study Framework (JSF)
3. Quick Simulator
4. FullSimulator (JIM and JLCSIM)
5. JSF demo

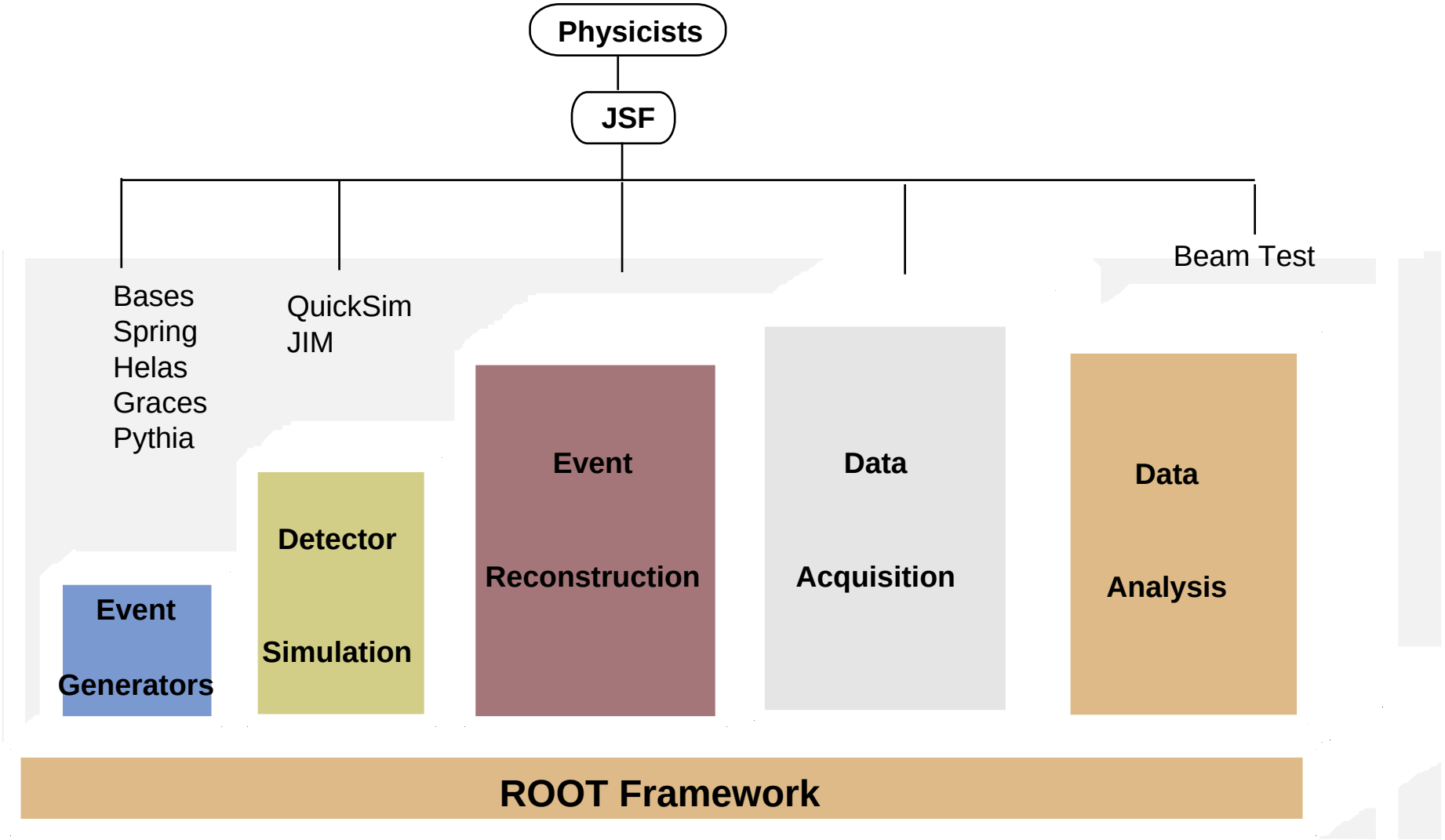
Postscript file is available at
http://www-jlc.kek.jp/~miyamoto/reports/docs/lcws2000-soft.ps.gz

Software overview



Root-JSF-QuickSim

JSF is a root based application to provide a common interface to physicists



http://root.cern.ch/ for root

http://acfahep.kek.jp/subg/sim/softs.html for links to JSF, QuickSim, JIM, Physsim, ...

Concept of JSF run control

General feature of HEP data analysis:

1. Event-by-event analysis
2. Event data consists of several sub-components, analizer of them needs initialization and termination when job or run begins

Standard flow of JSF job

Create *modules*
Job Initialize
Begin run
Event Analysis
End Run
Job Termination

- Execution flow are controled by a class JSFSteer.
- One Modules are created, calls of their function are controled by JSFSteer
- Thus, inclusion/exclusion of analysis module is easy.

All analysis classes must be inherited from JSFModule and JSFEventBuf

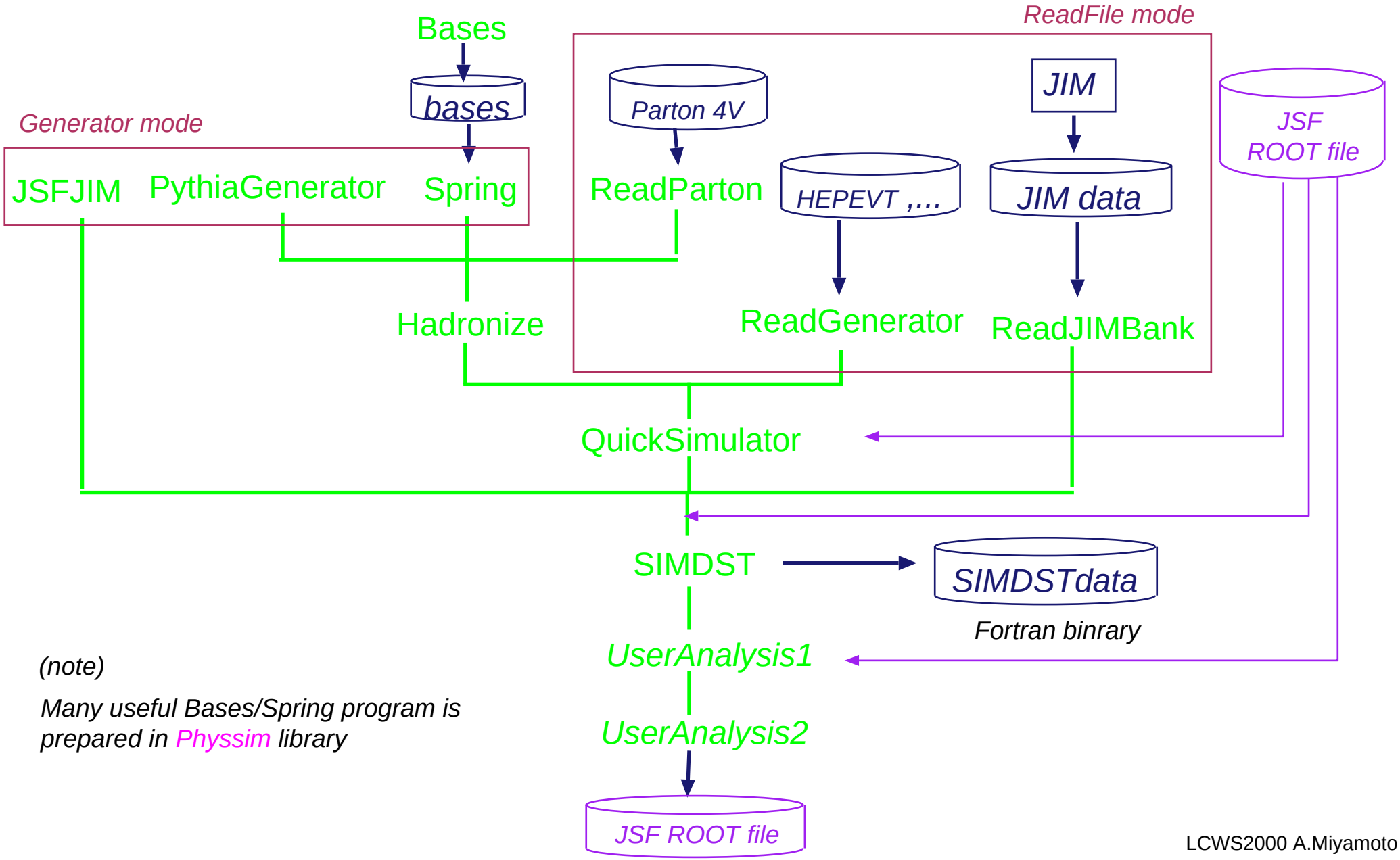
JSFModule : provide functions such as

Initialize(), BeginRun(), Process(), EndRun(), Terminate()

JSFEventBuf : A class to save event data in a ROOT file as a tree

Run mode and Modules

Stadard module and their relation



(note)

Many useful Bases/Spring program is prepared in Physsim library

Physsim Packages

(http://www-jlc.kek.jp/subg/offl/physsim)

Generator and Analysis packages developped by K.Fujii and his colleague

Generator:

Based on HELAS+BASES+SPRING

Processes:

higgs: zh
susy : sfsf, xcxc
top : eett, nntt, tth, tt, ttz
twoph: eeff (f=e,mu,tau,q)
wz : eeww, eez, enw, nnww, nnz, ww, wwz, zz

Included effects

Bremsstrahlung and beamstrahlung
W, tau polraization
Virtual Higgs diagrams

Analysis:

Packages such as jet finder, event shape routines, combiners, vertex tagger

Examples will be found at out web site

Classes to Read Data

In addition to the input from ROOT file, following classes are prepared to read non-ROOT file.

JSFReadParton

Read parton 4 momenta, helicities, etc, from a text file,
for subsequent hadronization and detector simulation.

Usefull for easy interface to event generators. ex: Grace, Pandora

JSFReadGenerator

Read particle information in HEPEVT format.

A file format is FORTRAN binary format.

JSFReadJIMBank

Read JIM full simulator data written in Zebra format, and saved themas class data

This class is derived from JSFSIMDST class, so analysis program used for QuickSim data can be also used to analize JIM data.

Output data

The standard output for JLC studies are called SIMDST

contents of SIMDST

Generator particles
Particle informations after CDC-VTX-Calorimeter linking
CDC Track information
VTX Hit point information
Calorimeter Hit Cell Information

JSF output SIMDST

as a tree of ROOT file for ROOT-user
as a Fortran binary file for non-ROOT user

Parameter file

All parameters are managed by JSFEnv class

In the user program, they are obtained by a class

JSFEnv::GetEnv(*"Parameter.Name"*, *default*)

At run time, parameter can be changed by three method

1. In a file, jsf.conf

Parameter.Name : value
#!*argname*
Comments
.....

argname is an alias of *Parameter.Name*
used to parse command line argument

2. As a command line argument, like

[%] jsf *--argname=value* gui.C

3. By popup menus of JSF Control panel

PythiaGenerator: Type of process, CM energy, etc
DebugGenerator: Particle ID, momentum, etc...

Each user can add their own menu by a function, *UserMenu()*

JSF Components

1. Libraries

Pre-compiled C++ classes to build JSF application
Such as libJSF.a, libJSFQuickSim.a, ...

2. Executables (= jsf)

Root libraries + JSF libraries + Fortran libraries ⇨ jsf application

Fortran libaries = lclib, jlcsim, cernlib, ...

3. Macro

C++ program is used as Macro thanks to CINT, no need to compile and link

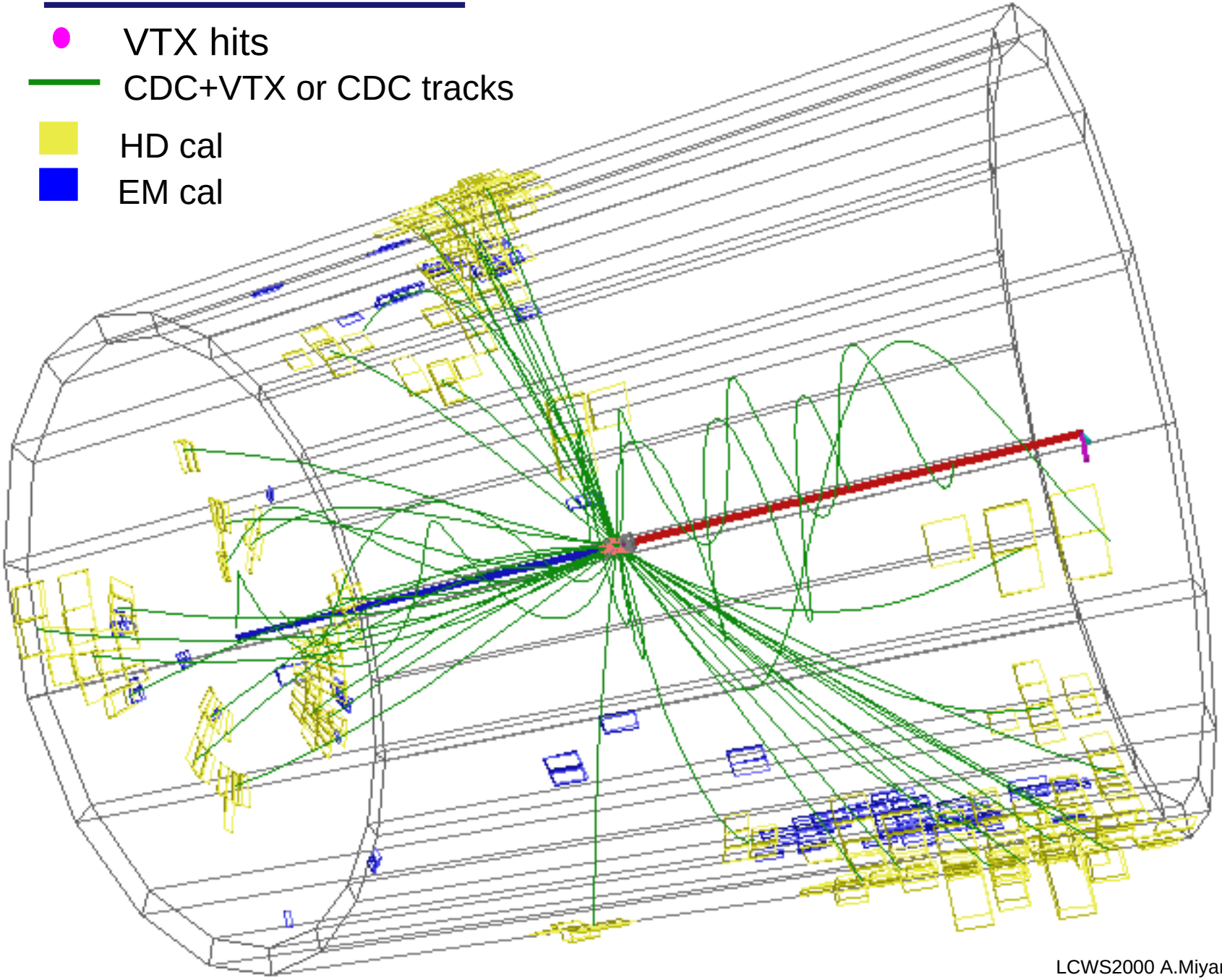
Macro can be used to set run parameters without compile/link

In the jsf distribution, gui.C, GUIMainMacro.C, and UserAnalysis.C are included as an analysis example

Quick Simulator

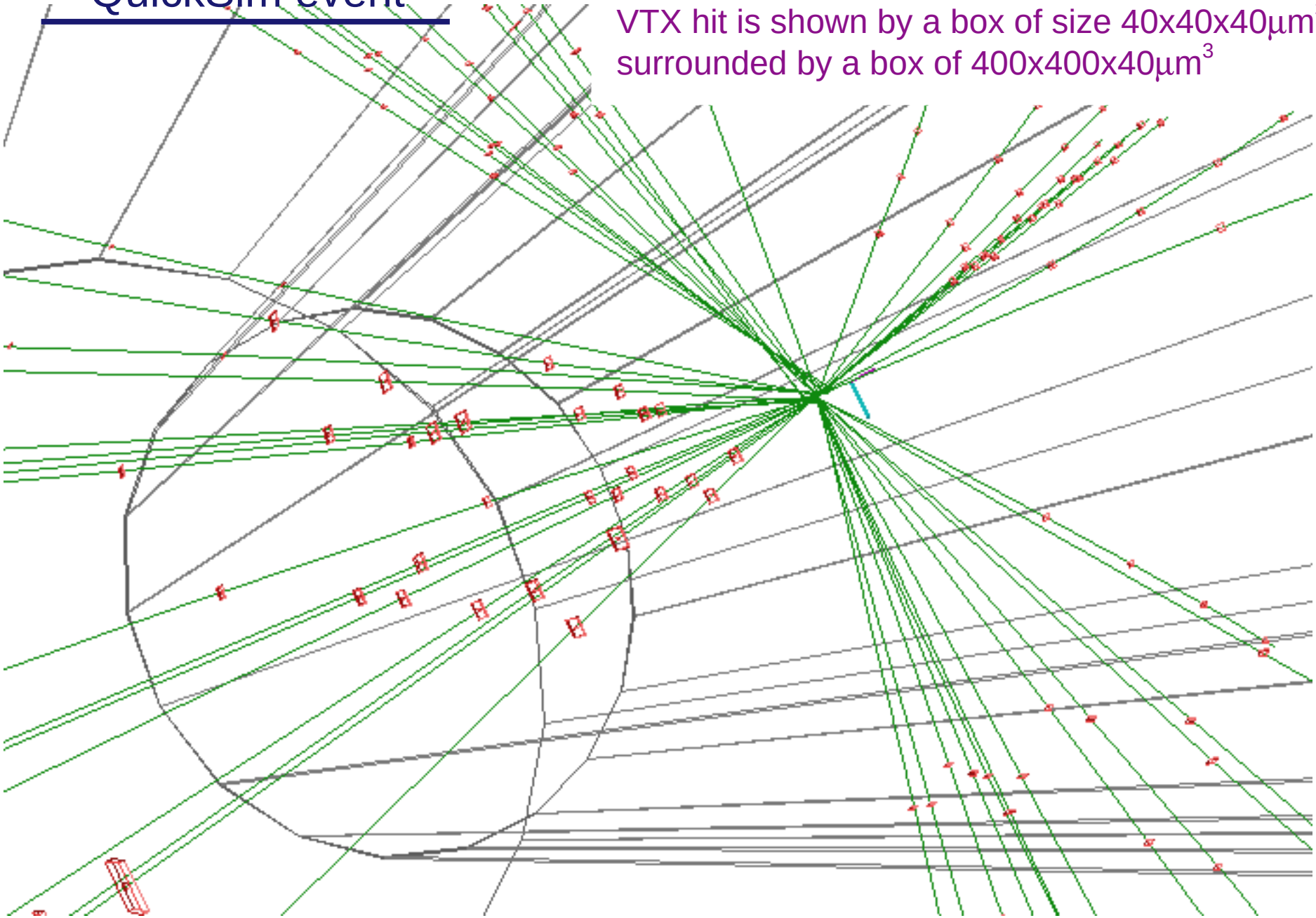
QuickSim event

- VTX hits
- CDC+VTX or CDC tracks
- HD cal
- EM cal



QuickSim event

VTX hit is shown by a box of size 40x40x40μm^3
surrounded by a box of 400x400x40μm^3



Quick Simulator

Detectr components:

VTX, IMT, CDC, CAL are included.
Detector parameters (resolution,geometry, etc) can be changed be a parameter file

Signal generation:

Particles are swimmed through VTX, IMT, CDC, and CAL.

Particles are smeared by multiple scattering by matterials such as VTX, IMT, etc.

VTX and CDC

Equally spaced N sampling with given $\sigma_{r\phi}$ and σ_z in solenoid field

5 dimensional error matric of the track parameter are smeared including the effect of the multiple scattering due to chamber gas.

VTX and CDC parameters are then averaged to get combined helix parameter

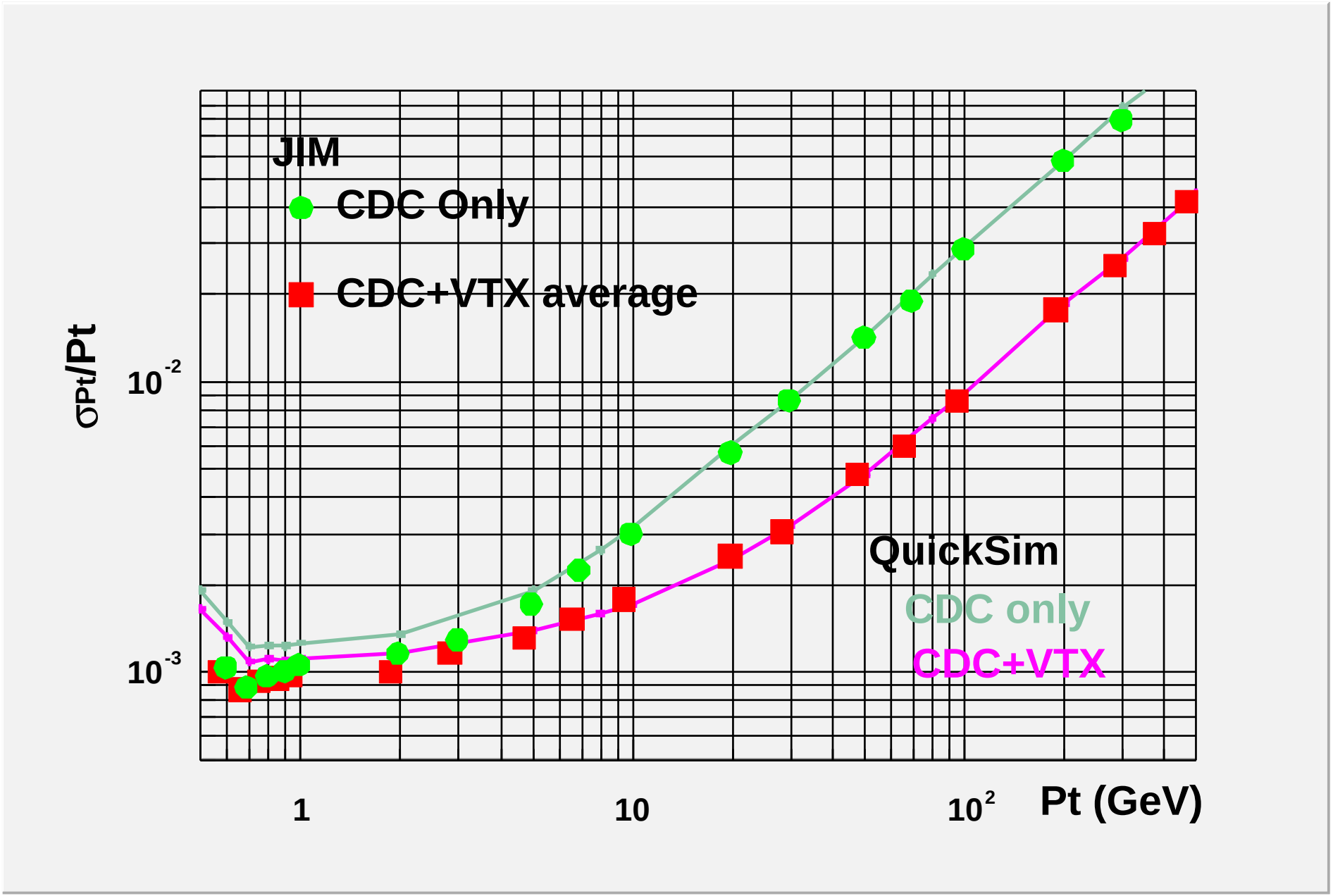
IMT Just create smeared hit points

CAL: Particle energy is spread laterally by $f(x) = a_1 \exp(-|x|/\lambda_1) + a_2 \exp(-|x|/\lambda_2)$

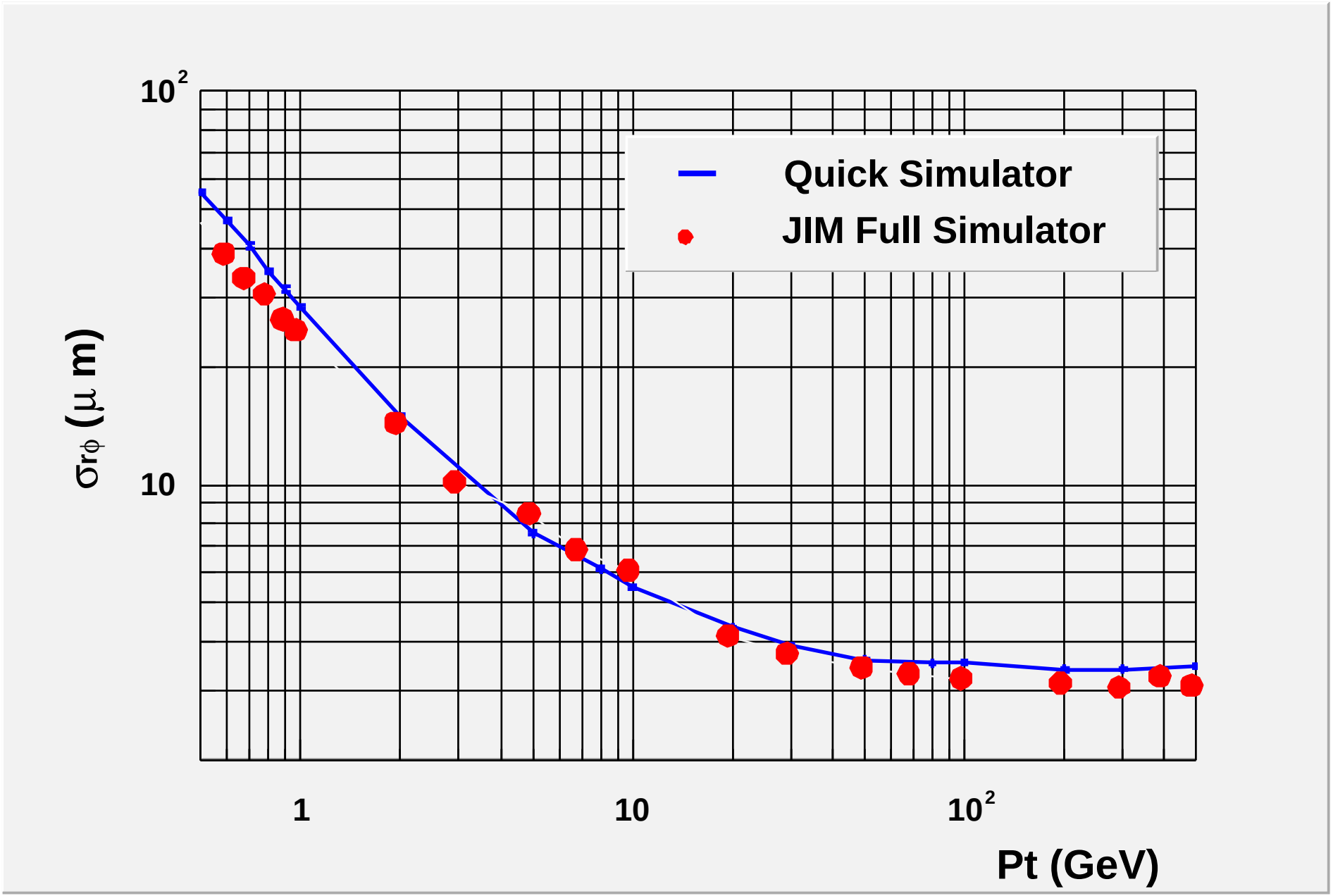
Generated energy is distributed to each countes after smearing according to the resolution.

e and γ : Deposite energy only in EM calorimter
hadrons : Deposite energy only in HD calorimter
μ : No energy deposite in calorimeters

QuickSim vs JIM comparison : Momentum resolution

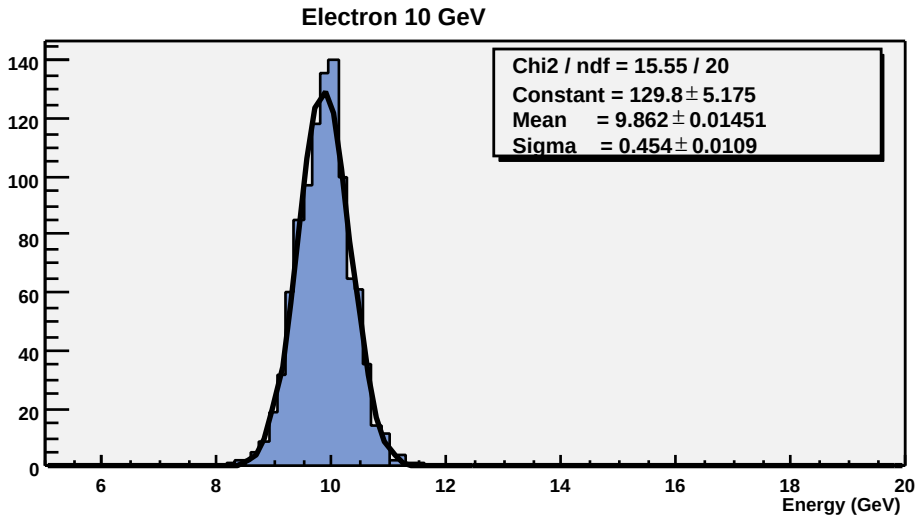


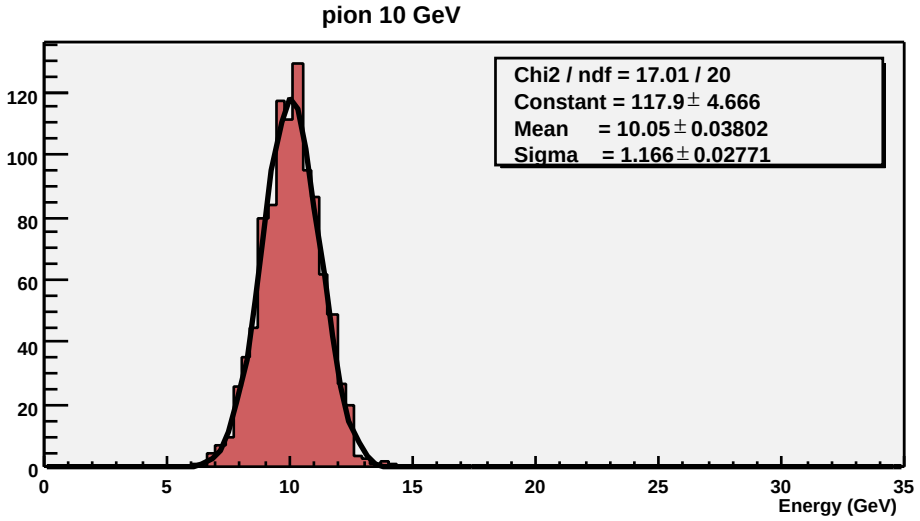
QuickSim vs JIM comparison : Impact parameter resolution



Comparison of calorimeter resolution

QuickSim

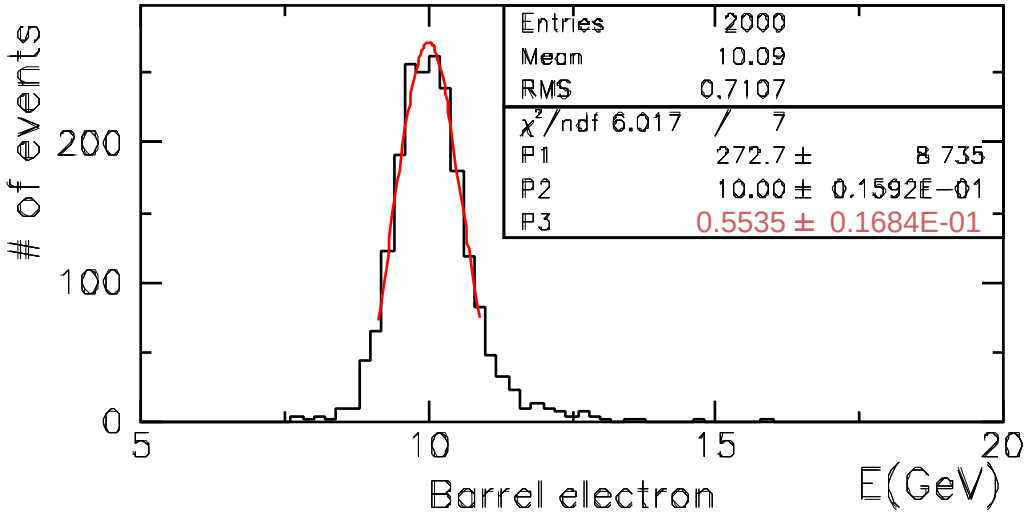


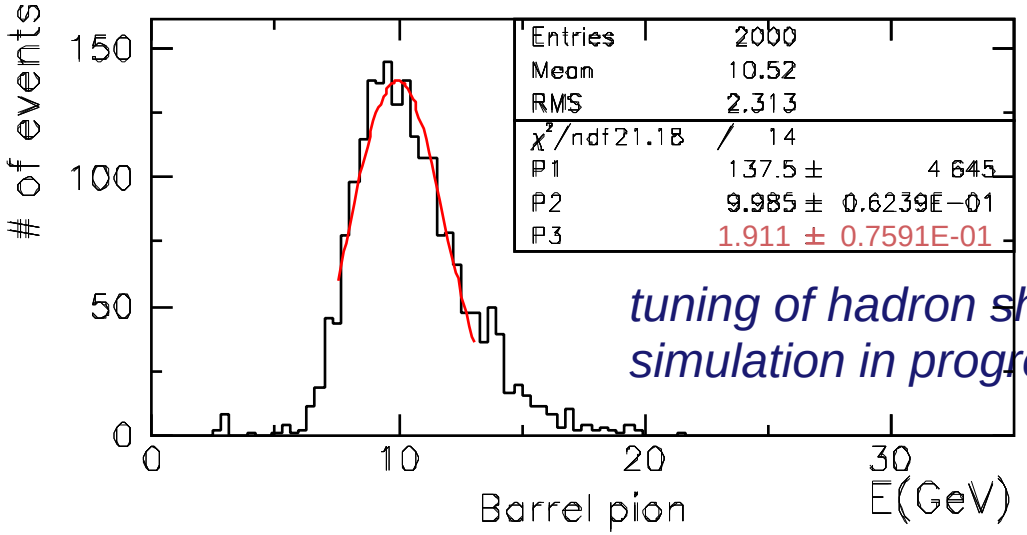


Energy resolution is OK, but tuneing of shower shape is not good yet.....

JIM

by Ishizawa Yoshio, Tsukuba Univ.





tuning of hadron shower simulation in progres....

Full detector simulator

JIM (JLCSIM)

Characteristics of full simulation program package for the JLC detector: "JLCSIM"

- a GEANT3 based FORTRAN program,
- includes ZEBRA based data structure,
- includes two versions of geometry models for both 2-Tesla and 3-Tesla of the magnetic fields,
- includes some analysis programs (not enough at present),
- can generate ntuple files which have a common variables with the quick simulation program for the JLC detector,
- managed by CVS,
- distributed via web pages:

  `http://www-jlc.kek.jp/subg/offl/www-jlcsim/index.html`

KEK

Almost all detector components are installed in a realistic shape for both geomtry models.

Status of the major detector components:

Name	2-Tesla	3-Tesla
Central Tracker	Full	Cylinder
Intermediate Tracker	Cylinder	Cylinder
Calorimeters	Full	Full
Vertex	Full	Full
Muon	Full	Full
Mask and beam pipe	Full	Full
Magnet and return yokes	Full	Full

Prospects for some improvements:

- needs some fine tuning of geometry parameters to reproduce realistic resolutions for hadron calorimeters,
- install realistic shapes for the remaining detector components,
- needs more flexible interface to event generators,
- now working on a script based (python) graphical user interface to run the program

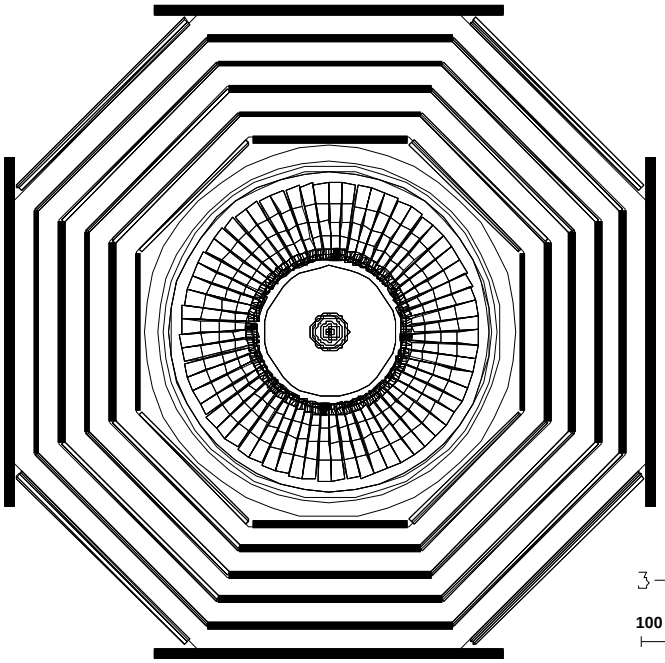
And...

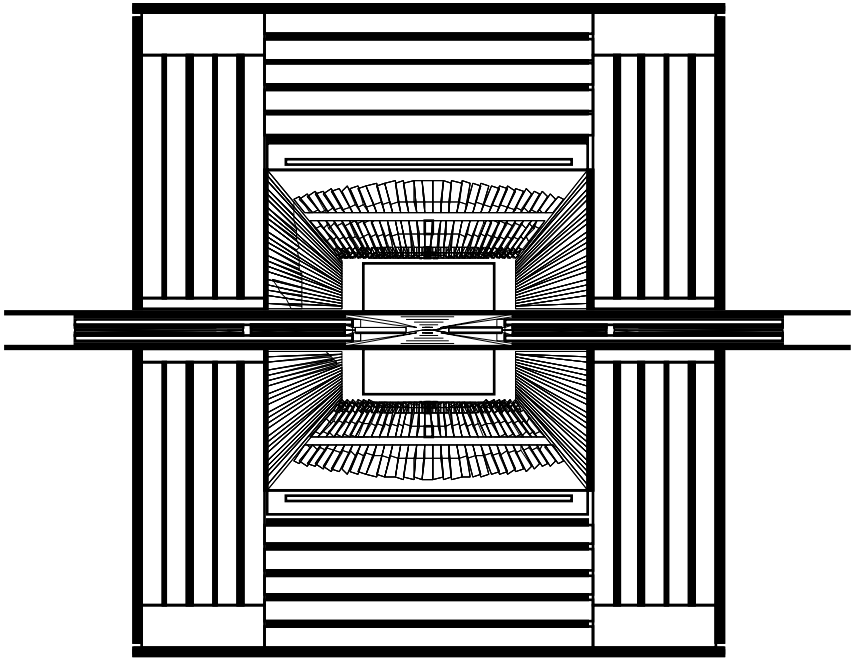
we have started to develop a new version of the full simulation program based on GEANT4.

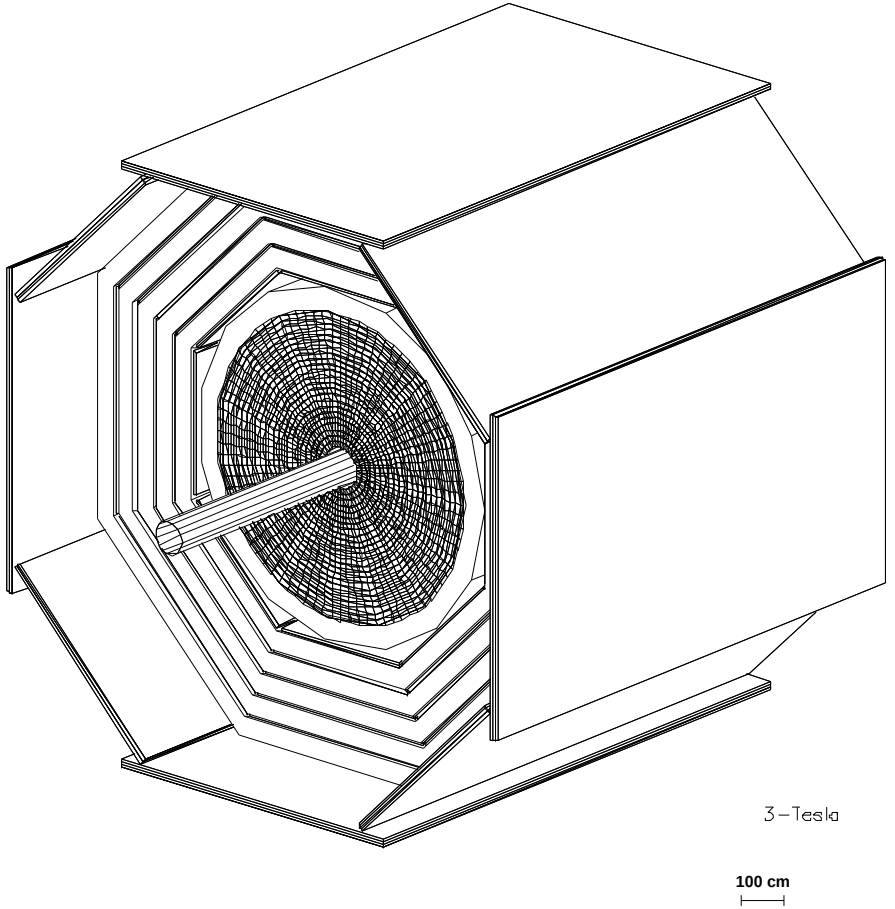
KEK

JLC 3T detector JIM

Junichi Kanzaki





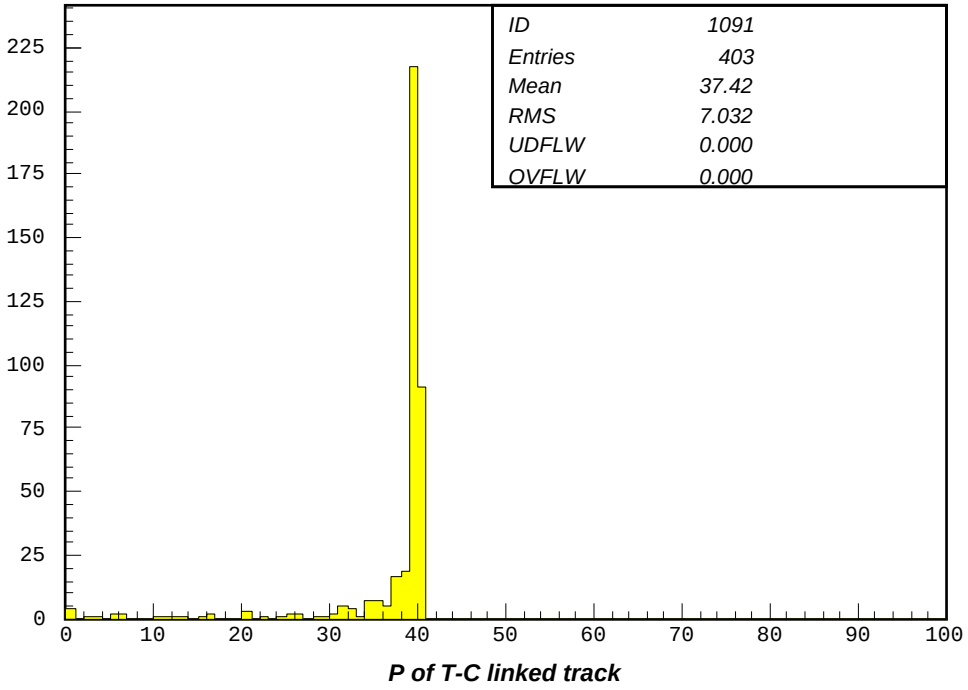


Resolution of EM calorimeter by JIM

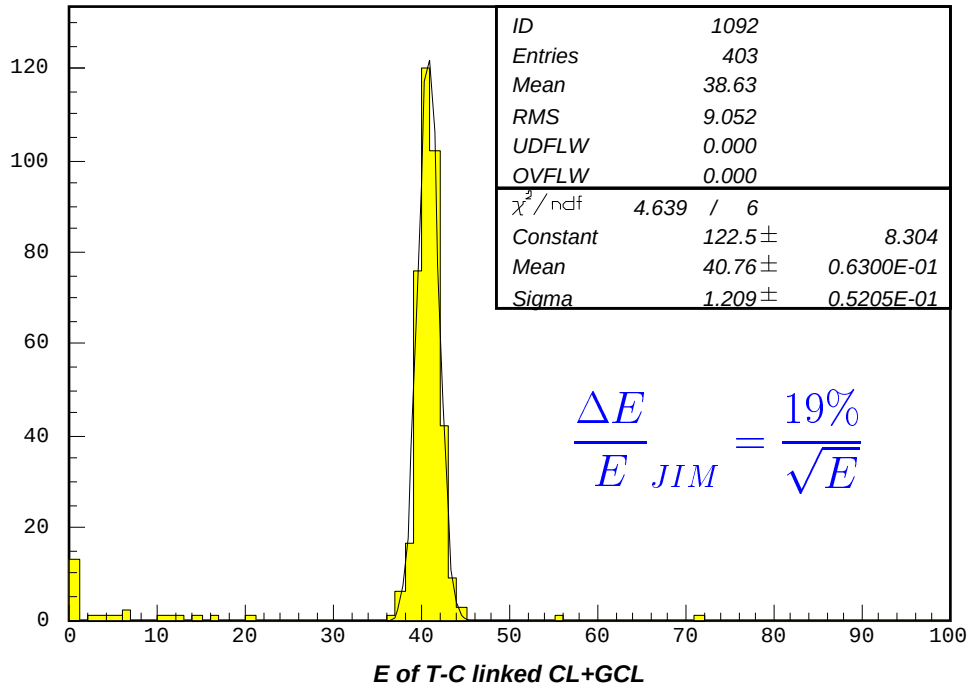
by Yoshiaki Fujii

JIM simulation of 40 GeV e^- and analysis

Momentum of e^- track



Energy of EM cluster linked with e^- track



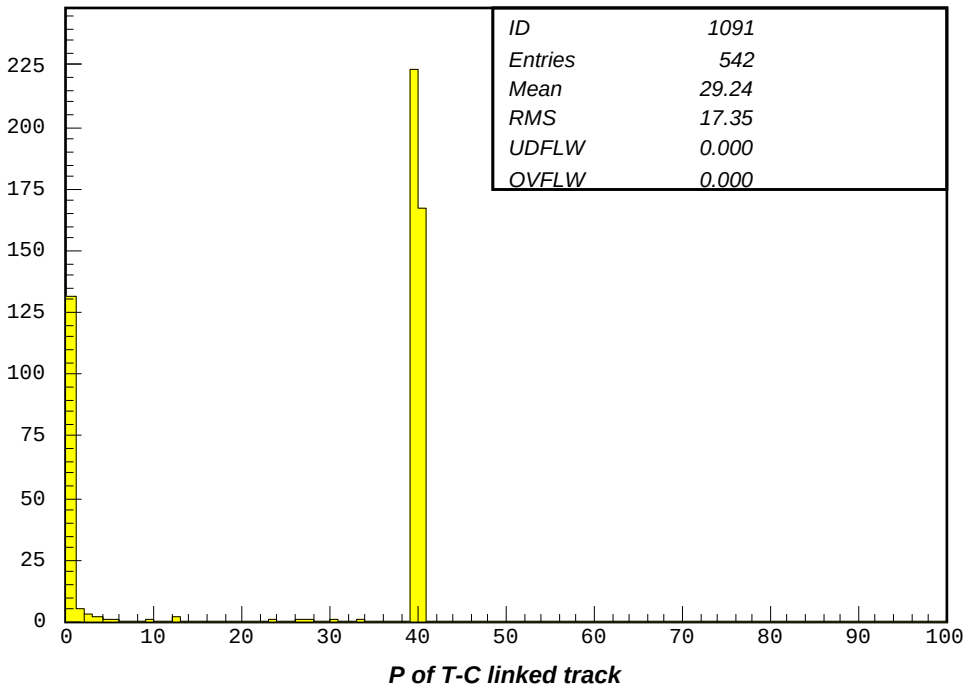
not as good as the beam test resolution yet .....

Resolution of HD calorimeter by JIM

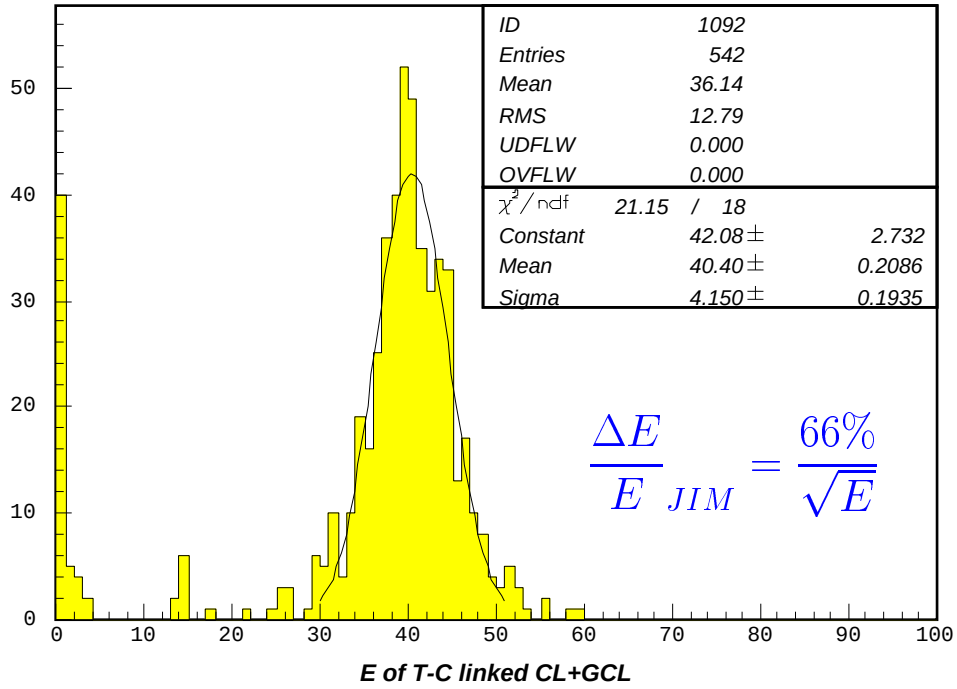
by Yoshiaki Fujii

JIM simulation of 40 GeV π and analysis

Momentum of π track



Energy of EM cluster linked with π track



tuning of simulation code is to be done
to achive beam test resolution of $40\%/\sqrt{E}$