Software for the ILC Vertex Tracker

Software Development and Validation in the `Marlin` Framework

M Battaglia
UC Berkeley and LBNL

Contributions by D Contarato, B Hooberman, W Yao

LCWS07 Conference
DESY June 2, 2007

From Sensor Simulation to Physics Analysis

Developing full suite of simulation and recontruction from pixel response to analysis of ILC events at highest energy;

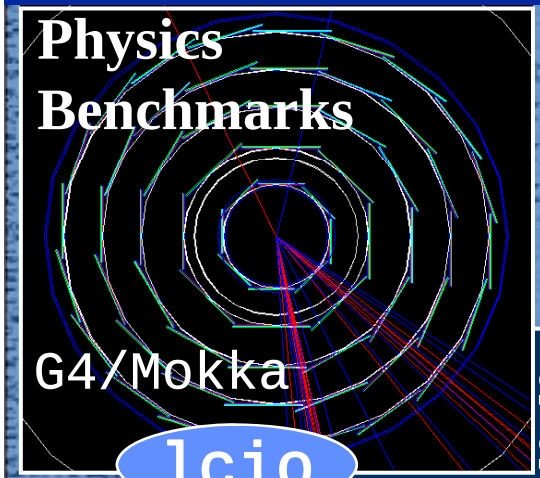
Implemented **sensor simulation** (`PixelSim`) & **cluster analysis** (`PixelAna`) in `Marlin` and interface to `LCIO` for beam test data (`PixelReader`)

`Marlin` Processors to analyze beam test data and validate simulation response, estimate effect of changes in sensors response and detector geometry and obtain realistic digitized simulation of full physics events and overlayed backgrounds;

CDF Vertex Fit and b-tagging ported to `Marlin` framework, being tested;

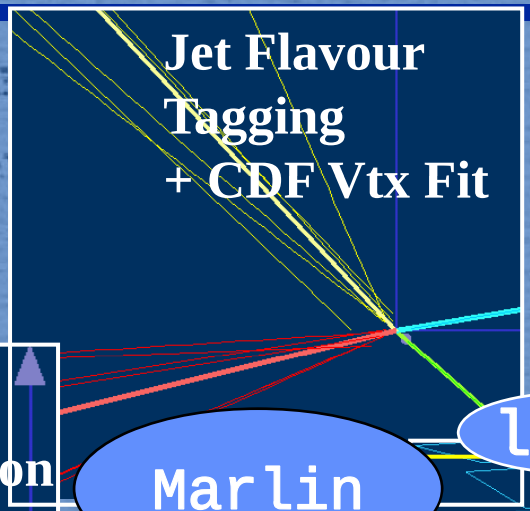
Jet Flavour Tagging and **Physics Analysis** of Dark Matter-motivated SUSY scenarios with fully simulated and digitized VTX currently in progress: physics benchmarking to provide guidance to sensor R&D.

VTX Simulation, Reconstruction, Analysis



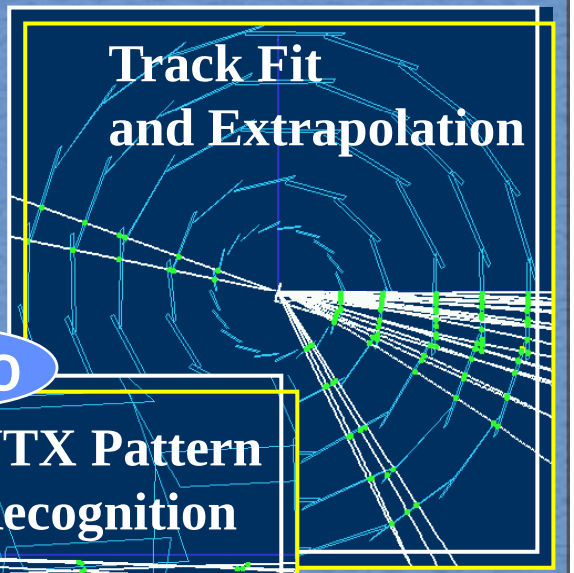
lcio

Sensor
Simulation

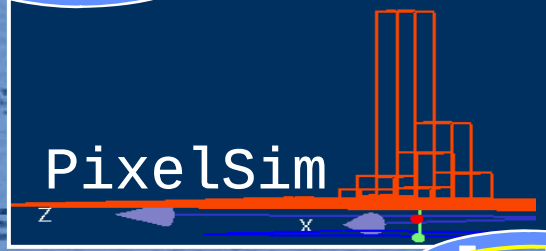


Marlin

lcio

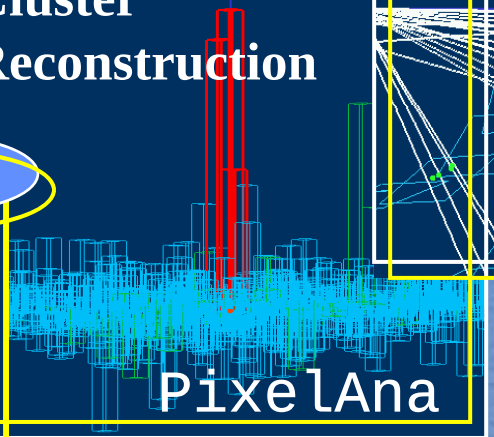


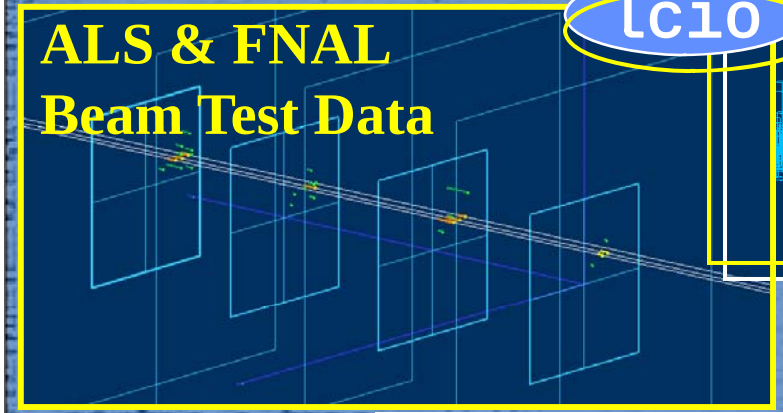
VTX Pattern
Recognition

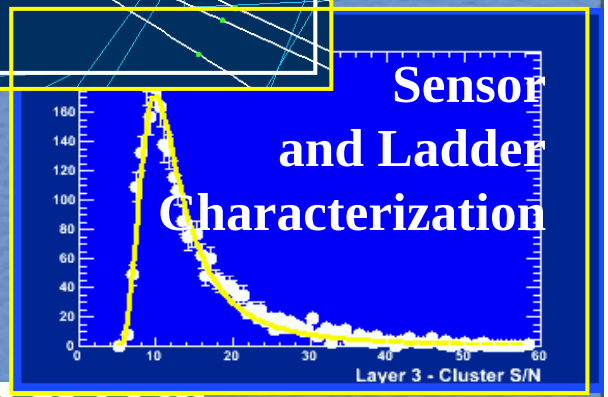


lcio

Cluster
Reconstruction



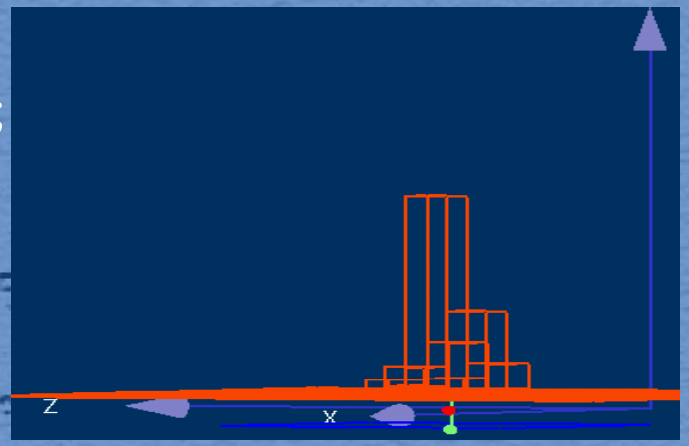




Pixel Simulation (`PixelSim`)

Develop sensor simulation starting from `VTXDigiProcessor` written by A. Raspereza as `MarlinReco` package, modifying CMS pixel sim.;

Use `SimTrackerHit` points of impacts from `Mokka` model energy loss fluctuations with `G4`;



Define pixel geometry and epi-layer thickness according to chip to be simulated, treat diffusion coefficient as free parameter, accounting for inter-pixel coupling, and adjust to fit cluster size, interface to `PixelAna` provides `TrackerPulse` and `TrackerRawData` data;

Data Conversion to lcio

Use `VtxRawData` class implemented in `LCIO`:

- Create VtxRawHits:

```
VtxRawHits* vtxData = new VtxRawHits(aDataFileName,aFrame);
LCWriter* lcWrt = LCFactory::getInstance()->createLCWriter();
```

- Create and fill header with chip geometry information:

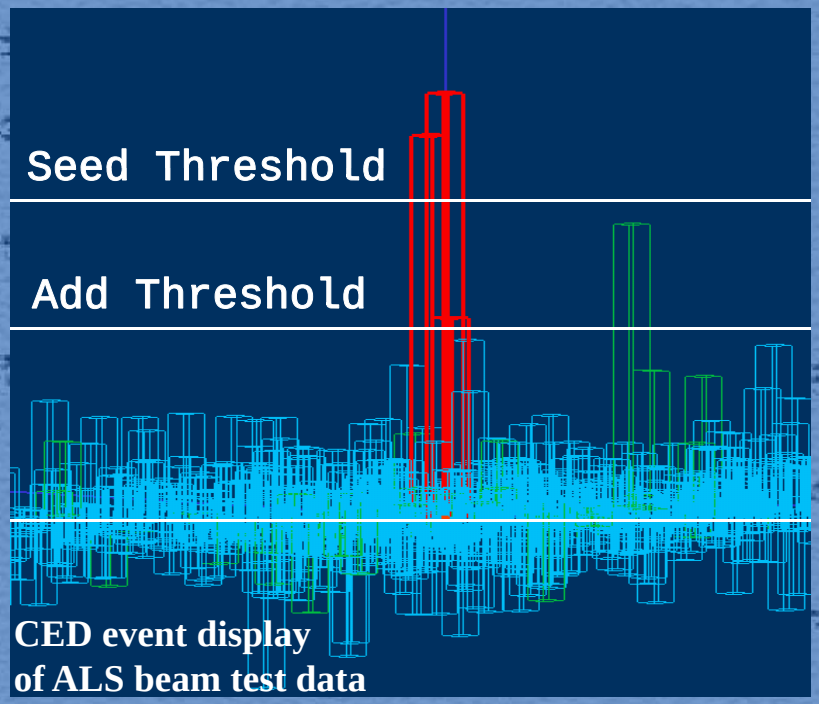
```
string detName("MIMOSA V - det02");
runHdr->setDetectorName( detName );
runHdr->parameters().setValue( "NRows" , vtxData->getNRows());
...
```

- Store ADC, Noise and Pedestal (if update online), otherwise, just store ADC and read initialisation Noise and Pedestal from Condition Database;

```
TrackerRawDataImpl* vtxRaw = new TrackerRawDataImpl ;
vtxRaw->setCellID0(vtxData->getIndex(iRow,iCol,aFrame)) ;
vtxRaw->adcValues().resize(3);
vtxRaw->adcValues()[0]=10*vtxData->getADC(iRow,iCol,aFrame);
...
```

Pixel Clustering (`PixelAna`)

`PixelAna` starts with `TrackeRawData` (pulse height and initial noise and pedestal), flags noisy pixels, updates noise and pedestal values, performs cluster search and generate `TrackerHit` data:



CED event display of ALS beam test data

1st Algorithm: set two S/N thresholds for seed and additional pixels to be added to cluster;

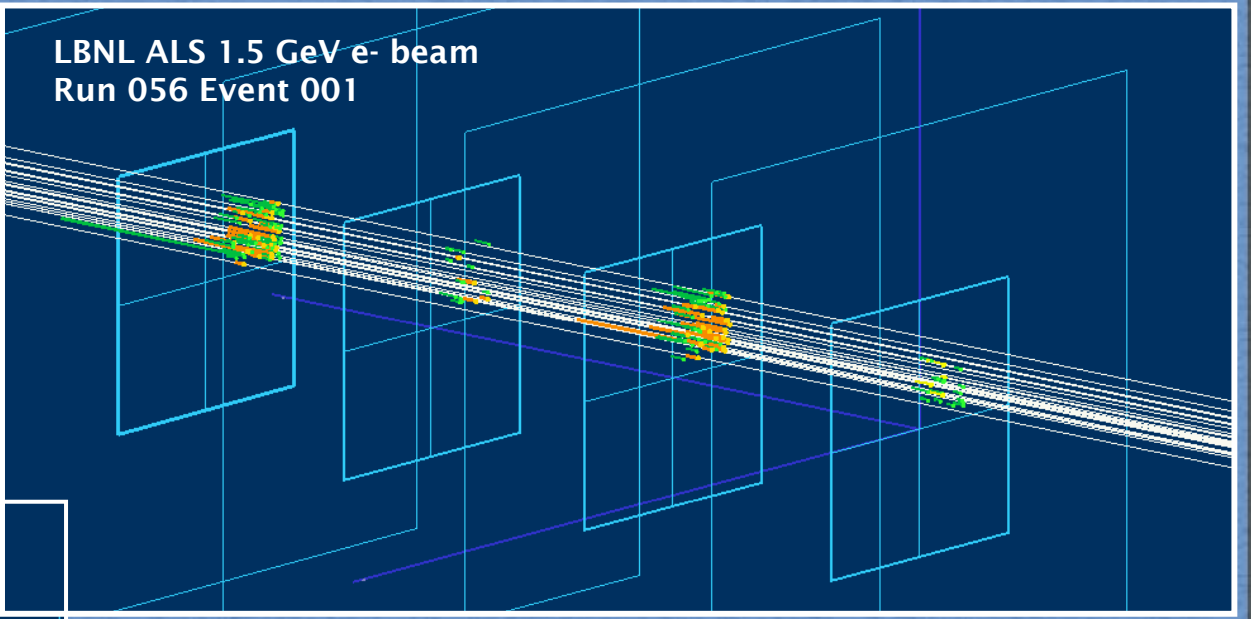
2nd Algorithm: set S/N threshold for seed pixel and add fixed NxN matrix of neighboring pixels.

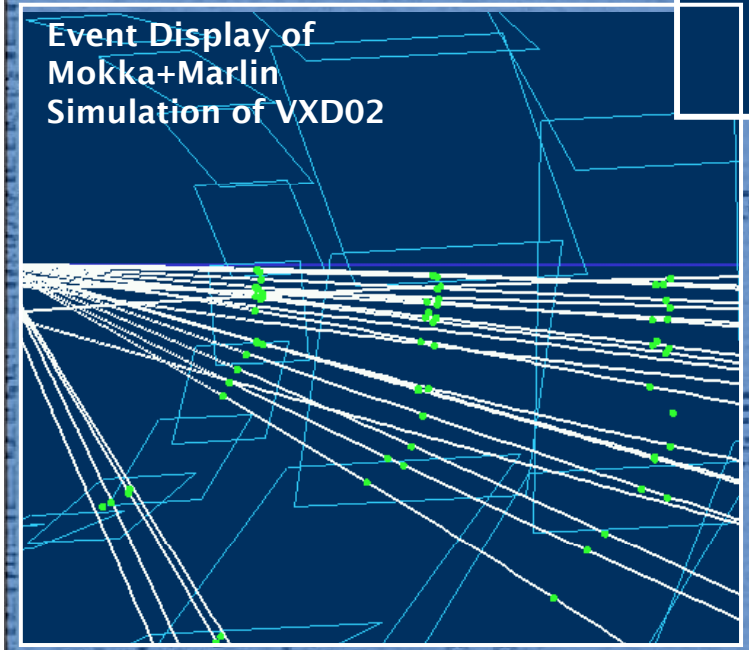
Characterize cluster shape in terms of pixel multiplicity projected over major and minor cluster axis.

Currently implementing merged cluster recovery algorithm;

PatRec & Linear Track Fit (`TelescopeTrackFit`)

Standalone Vertex Tracker **pattern recognition** tested on both simulated ILC evts and beam test data;





Implemented track fit accounting for kinks in measuring planes (G. Lutz NIM A273 (1988)), needed to analyse low momentum electron tracks at ALS.

CDF and HERAB Vertex Fit (`CTVMFTVertexFit` & `TelescopeVertexFit`)

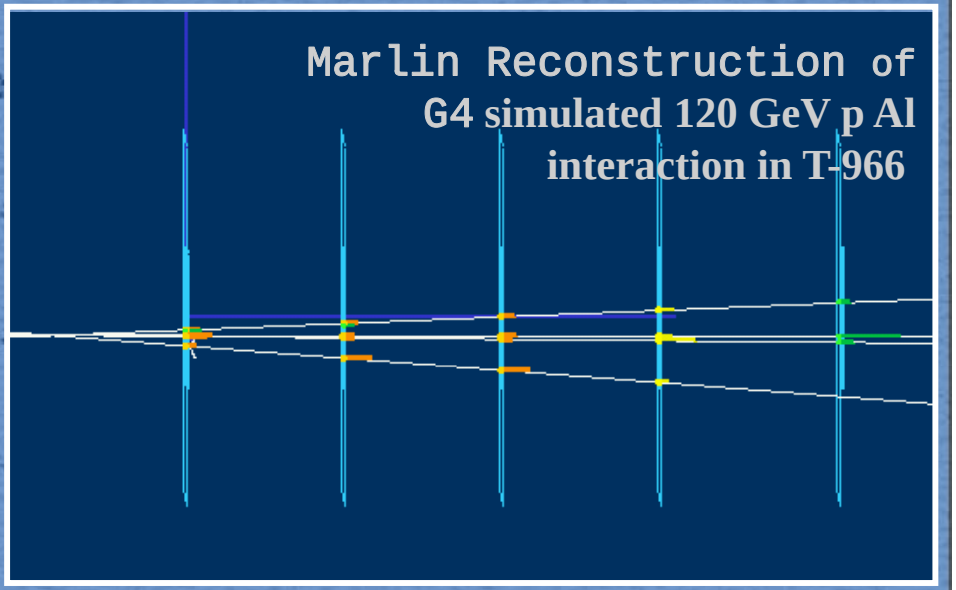
Port of `CTVMFT` developed for CDF (J Marriner, JP Berge, F Bedeschi), and used by ATLAS, to `MarlinReco` framework using a `C++` wrapper;

Geometrical vertex fit allowing mass, beamspot constraints; currently used to implement recovery of b-jet energy with s.l. decays;

Port of `VTlib` originally developed for HERAB (T Lohse), and later ported to LHCb, to `MarlinReco` framework, by a `C++` wrapper;

Kalman Filter to perform vertex fit of straight tracks, allows to impose mass constraints;

Useful for beam tests w/o B field, will be used for analysis of T-966 data at FNAL MBTF.



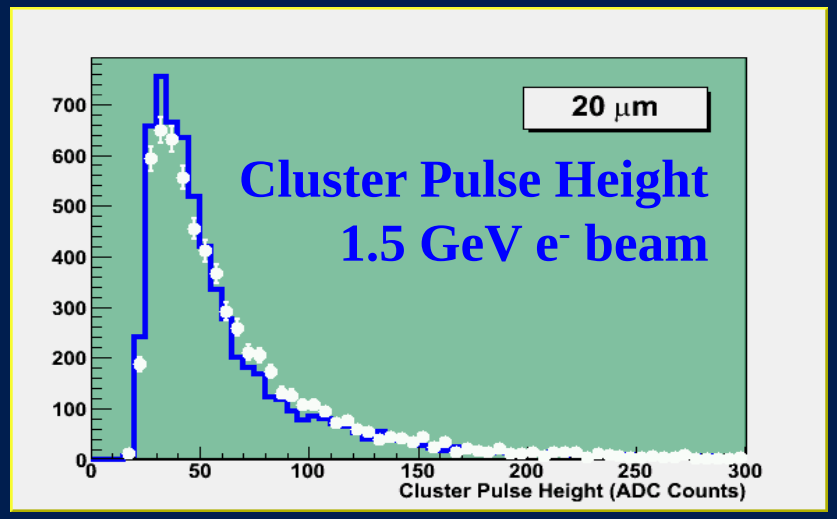
Starts from `Tracks` collection and generates a `Vertex` collection

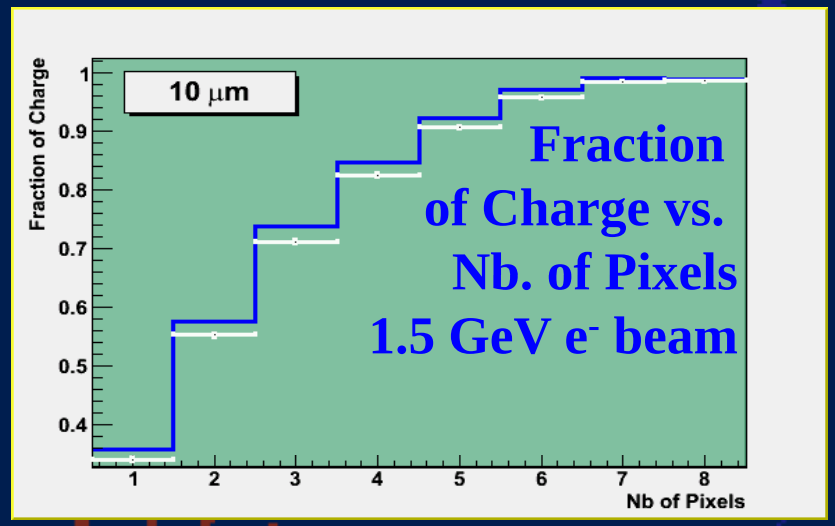
Simulation Validation

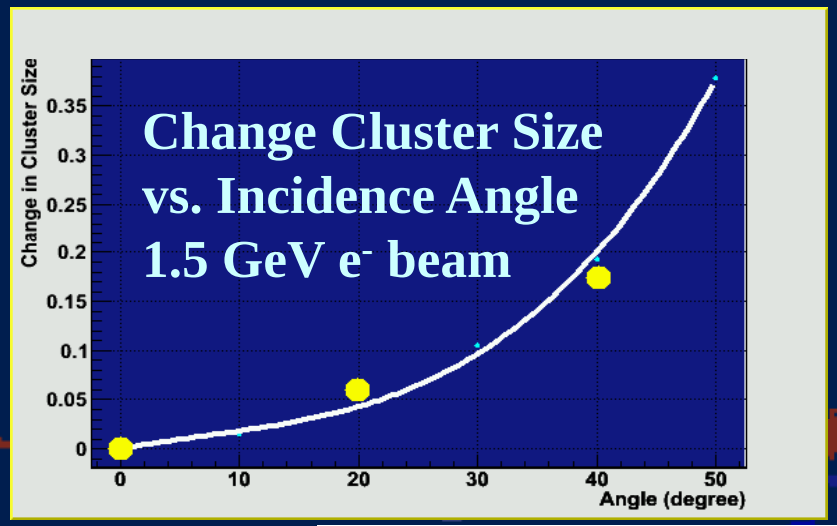
Several test beam data sets converted into `LCIO` format and used for validation of pixel simulation/reconstruction:

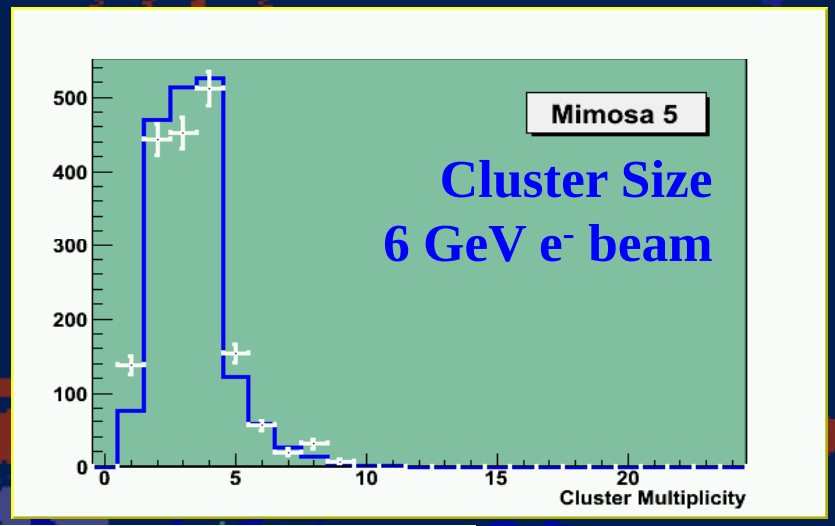
Chip	Pixel Size (μm)	Epi-layer (μm)	Beam	Energy (GeV)	
LDRD-1	10	10	e^-	1.5	ALS
LDRD-1	20	10	e^-	1.5	ALS
LDRD-1	40	10	e^-	1.5	ALS
LDRD-2	20	14	e^-	1.2	ALS
Mimosa 5	17	14	e^-	1.5	ALS
Mimosa 5	17	14	e^-	3.0	DESY
Mimosa 5	17	14	e^-	6.0	DESY
Mimosa 9	20	15	π^-	120.0	CERN

Simulation Validation

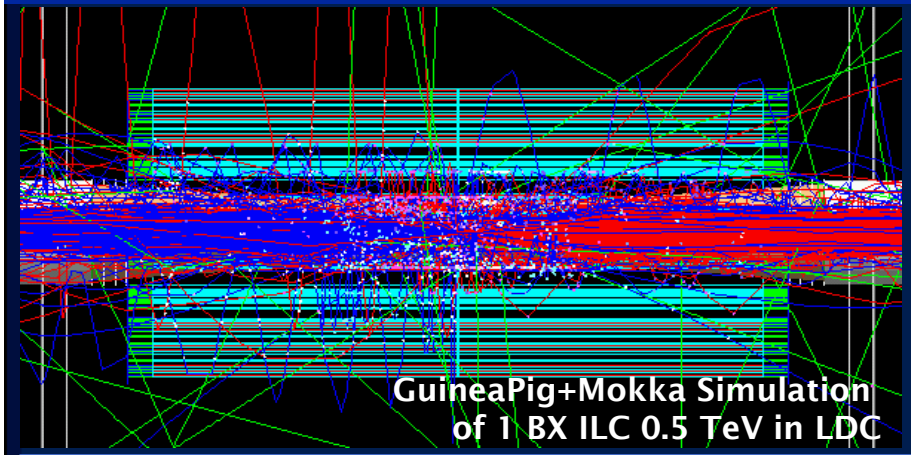


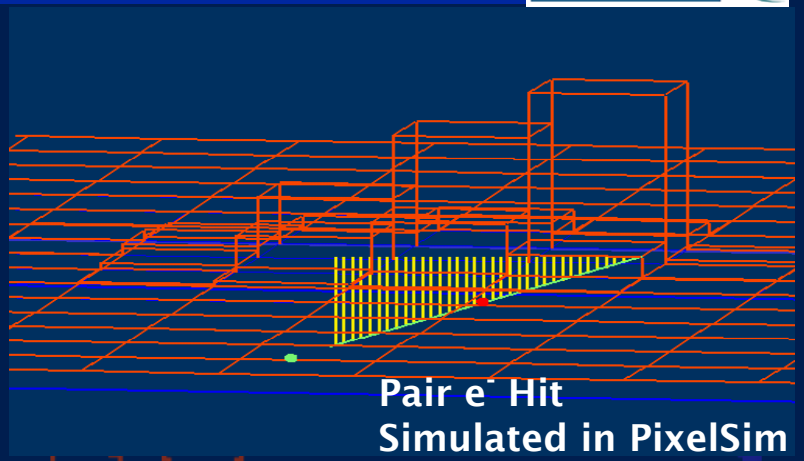




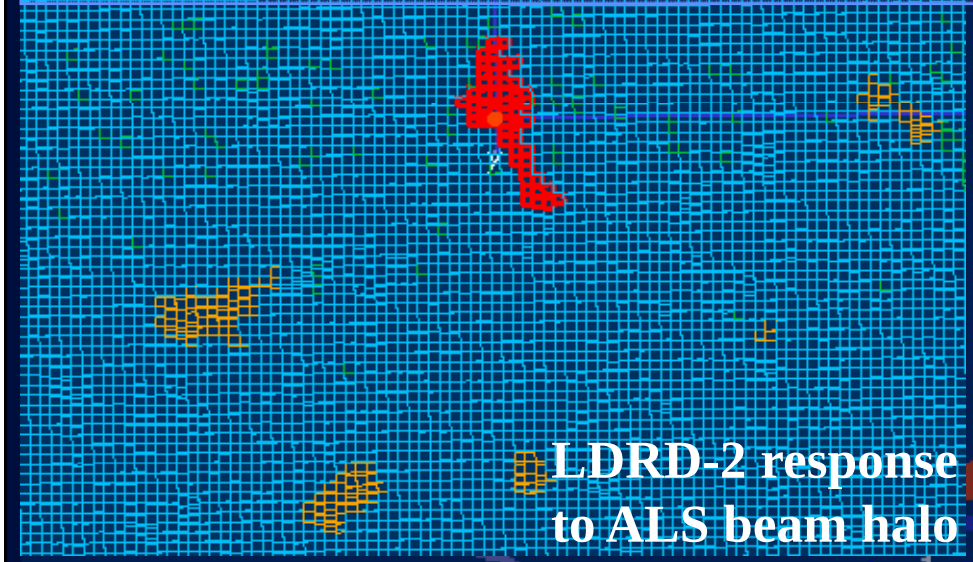


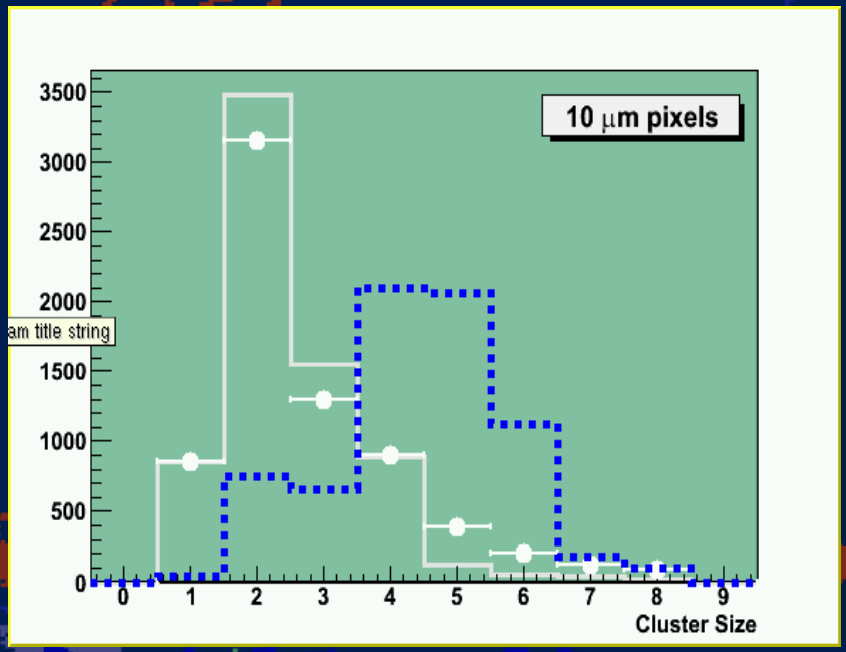
Response to Pair Background





Beam tests using 0.05 - 1.0 GeV e^- beams at **LOASIS** and **ALS** to validate simulation.





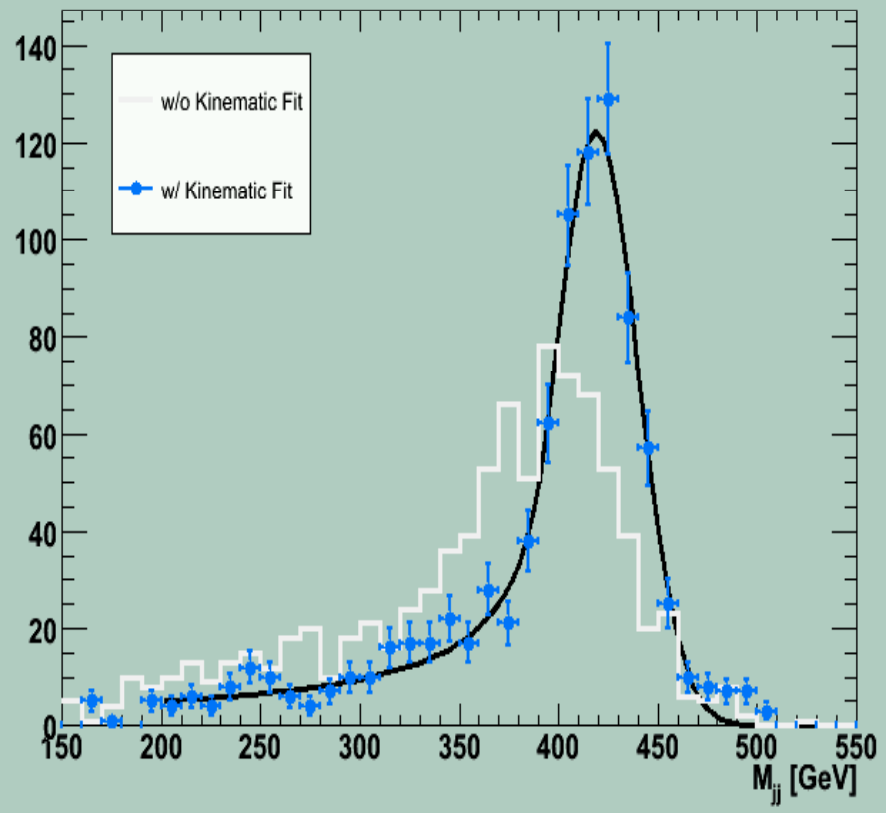
DELPHI Mass Constrained Fit (`MassConstrainedJJFit`)

Port of `PUFITC+` developed for DELPHI at LEP2 (N Kjaer, M Mulders) to `MarlinReco` framework using a `C++` wrapper;

Performs constrained kinematic fit to 4-jet system, which uses Lagrange multipliers and minimises a χ^2 constructed from the measured energies and directions of the jets;

Allows to impose centre-of-mass energy and momentum conservation, decay mass, equal masses for jj pairs;

Example in HA→bbbb at 1 TeV



Conclusions

Program of software development being carried out at LBNL: MarlinReco processors to support the Vertex Tracker data analysis and the physics simulation and reconstruction of benchmark processes;

Processors being applied to both real and simulated data;

Expect finalisation of core set of processors in late summer for analysis of T-966 beam test at Fermilab;

Plan for release of tested processors starting in Fall.