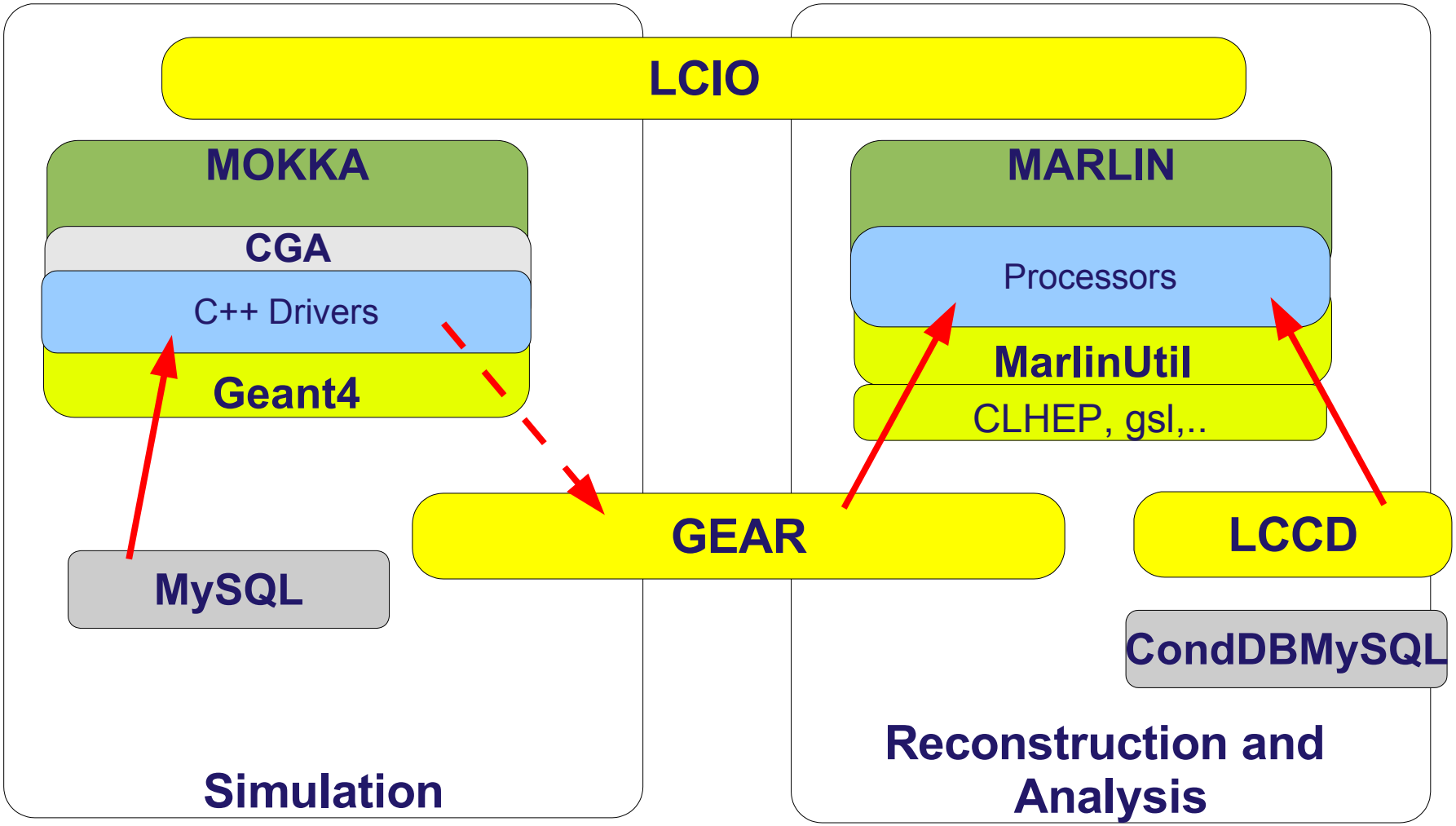
Analysing events for the ILC, using Marlin and MarlinReco

Outline

- LDC Simulation and Reconstruction Framework
- Marlin
- Processors for Marlin
- Software Packages and MarlinReco
- Event Reconstruction with MarlinReco on the GRID
- Summary

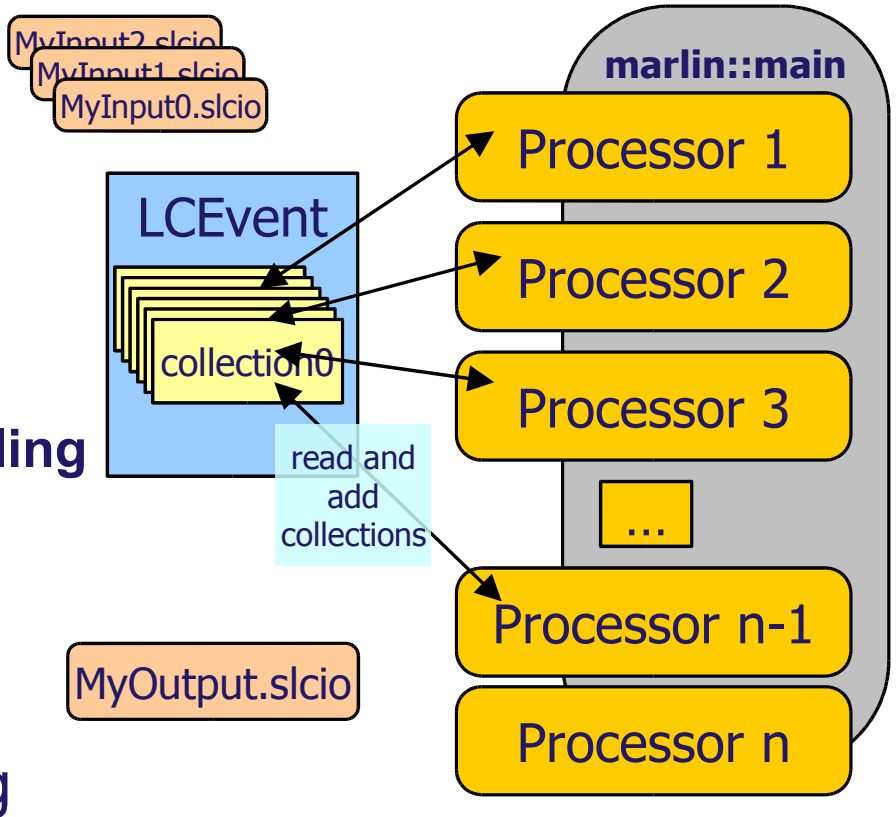
LDC Simulation and Reconstruction Framework



Marlin

Modular**A**nalysis & **R**econstruction for the **L I N**ear Collider

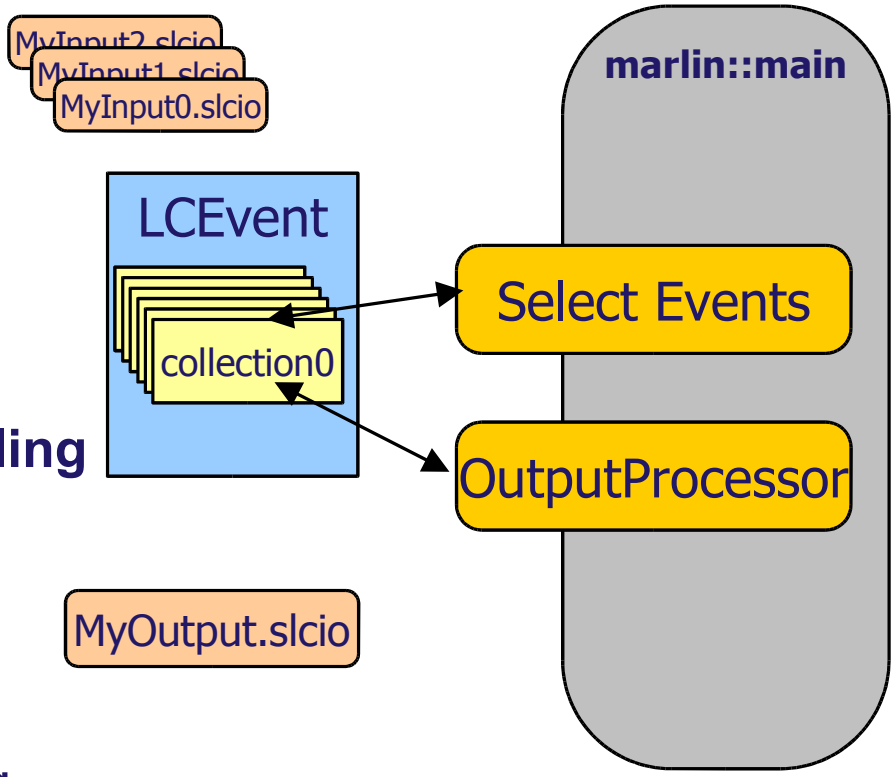
- modular C++ **application framework** for the analysis and reconstruction of LCIO data
  - modules called Processors
  - provides **main program**
- user steering via steering file
  - exchange Processors **w/o recompiling**
  - **same Processor** with **different parameters** in **one job**
  - conditional jump (if)
- task can be as simple as selecting events or as complex as a full event reconstruction



Marlin

Modular**A**nalysis & **R**econstruction for the **L I N**ear Collider

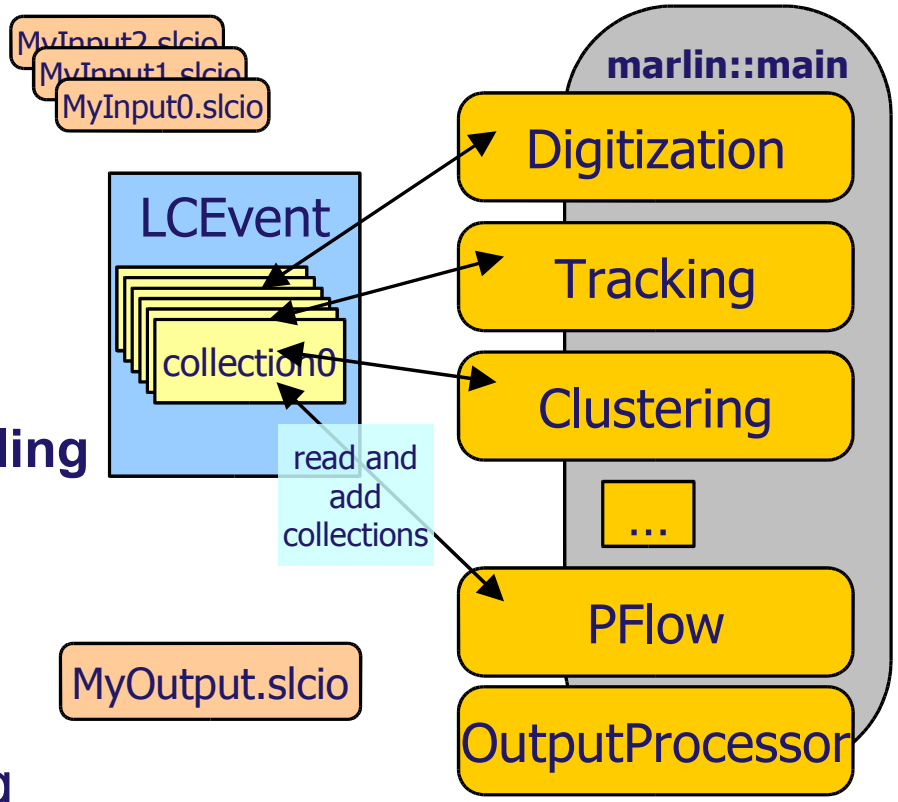
- modular C++ **application framework** for the analysis and reconstruction of LCIO data
  - modules called Processors
  - provides **main program**
- user steering via steering file
  - exchange Processors **w/o recompiling**
  - **same Processor** with **different parameters** in **one job**
  - conditional jump (if)
- task can be as simple as selecting events or as complex as a full event reconstruction



Marlin

Modular**A**nalysis & **R**econstruction for the **L I N**ear Collider

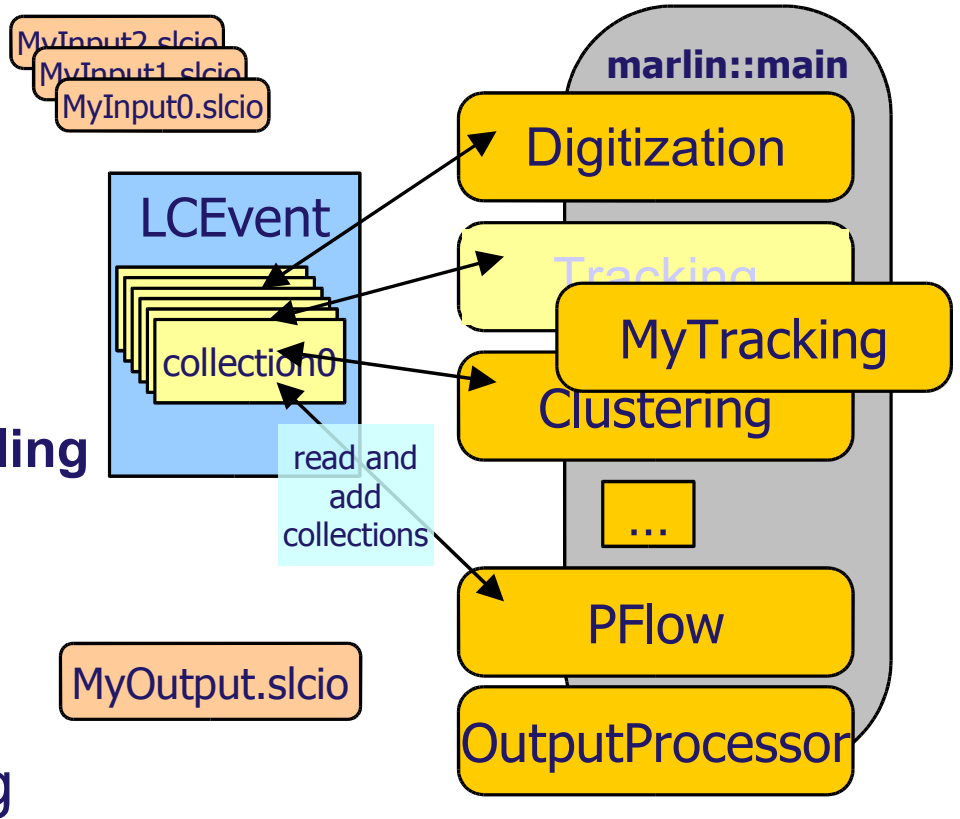
- modular C++ **application framework** for the analysis and reconstruction of LCIO data
  - modules called Processors
  - provides **main program**
- user steering via steering file
  - exchange Processors **w/o recompiling**
  - **same Processor** with **different parameters** in **one job**
  - conditional jump (if)
- task can be as simple as selecting events or as complex as a full event reconstruction



Marlin

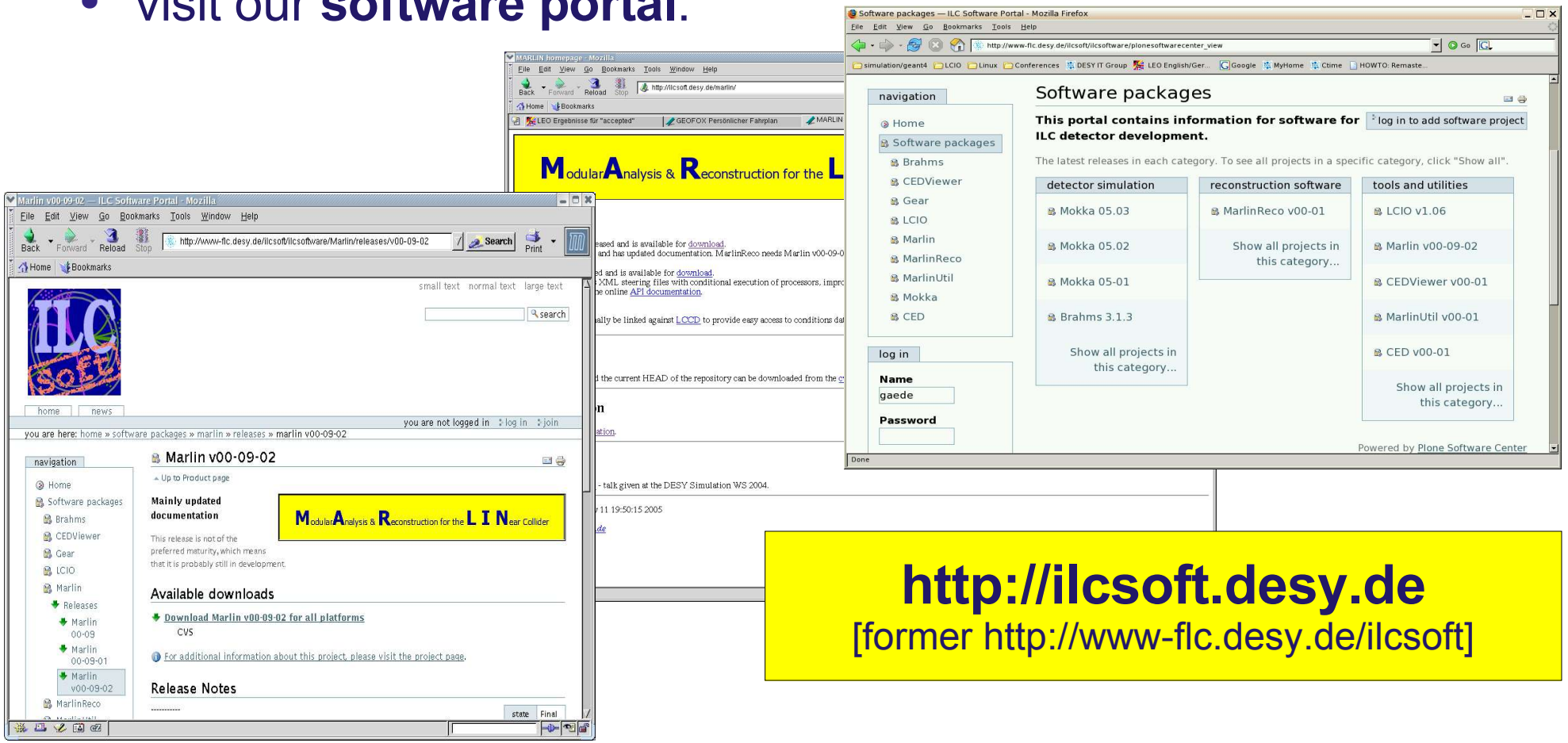
Modular**A**nalysis & **R**econstruction for the **L I N**ear Collider

- modular C++ **application framework** for the analysis and reconstruction of LCIO data
  - modules called Processors
  - provides **main program**
- user steering via steering file
  - exchange Processors **w/o recompiling**
  - **same Processor** with **different parameters** in **one job**
  - conditional jump (if)
- task can be as simple as selecting events or as complex as a full event reconstruction



Status of Marlin

- Marlin v00-09-02 available on the web
  - ➔ it allows to use GEAR and has updated documentation
- visit our **software portal**:



Processors for Marlin

- full set of Processors for **event reconstruction** available

- **Track Digitisation**
  - simple smearing
- **Calorimeter Digitisation**
  1. simple: calibration, ganging, threshold cut for energy
  2. advanced: based on detector geometry, calibration, ganging
- **Tracking**
  - algorithms taken from LEP (ALEPH and DELPHI)
  - full tracking in TPC
  - VTX, SIT hits included in the track fit
  - TPC tracks used as 'seeds'

Processors for Marlin

- **Clustering (Trackwise Clustering)**
  - spatial information needed only
  - → applicable to digital and analogue calorimeters
  - minimal dependence on detector geometry
  - → can be used for different detector designs
- **PFlow (Wolf)**
  - Track – Cluster matching
  - → extrapolate tracks into the calorimeter
  - get (E,p) for charged particles from track parameters
  - get (E,p) for neutral particles from cluster
  - Particle ID assigned by fraction of energy in ECAL / HCAL

Processors for Marlin

- **Track and Cluster Cheater**
  - use Monte Carlo information to combine hits to tracks and clusters
  - tracker hits are fitted with simple helix hypothesis
  - → use instead of realistic tracks and clusters
  - → compare with them
- **Analysis**
  - Event Shapes:
  - → Thrust Reconstruction (Tasso & Jetnet algorithms)
  - → Sphere (sphericity, aplanarity, ...)
  - Satoru Jet Finder
  - → originally developed by Satoru Yamashita for OPAL

Software Packages for Marlin

- **Package** = **set** of Marlin **Processors**, classes and functions
  - MarlinReco → full set of Processors for **event reconstruction**
    - Digitisation, Tracking, Clustering, PFlow and Analysis
  - MarlinUtil → **utility** and **helper** classes
  - CEDViewer → simple **event display** client processor
- stand-alone software modules:
  - CED → event display based on GLUT / OpenGL
  - RAIDA → **AIDA to ROOT** interface (coming up soon)

- Packages are **not** fixed
  - MAGIC → different Clustering, partly uses MarlinReco
  - PandoraPFA → Clustering, PhotonID, Cluster Association

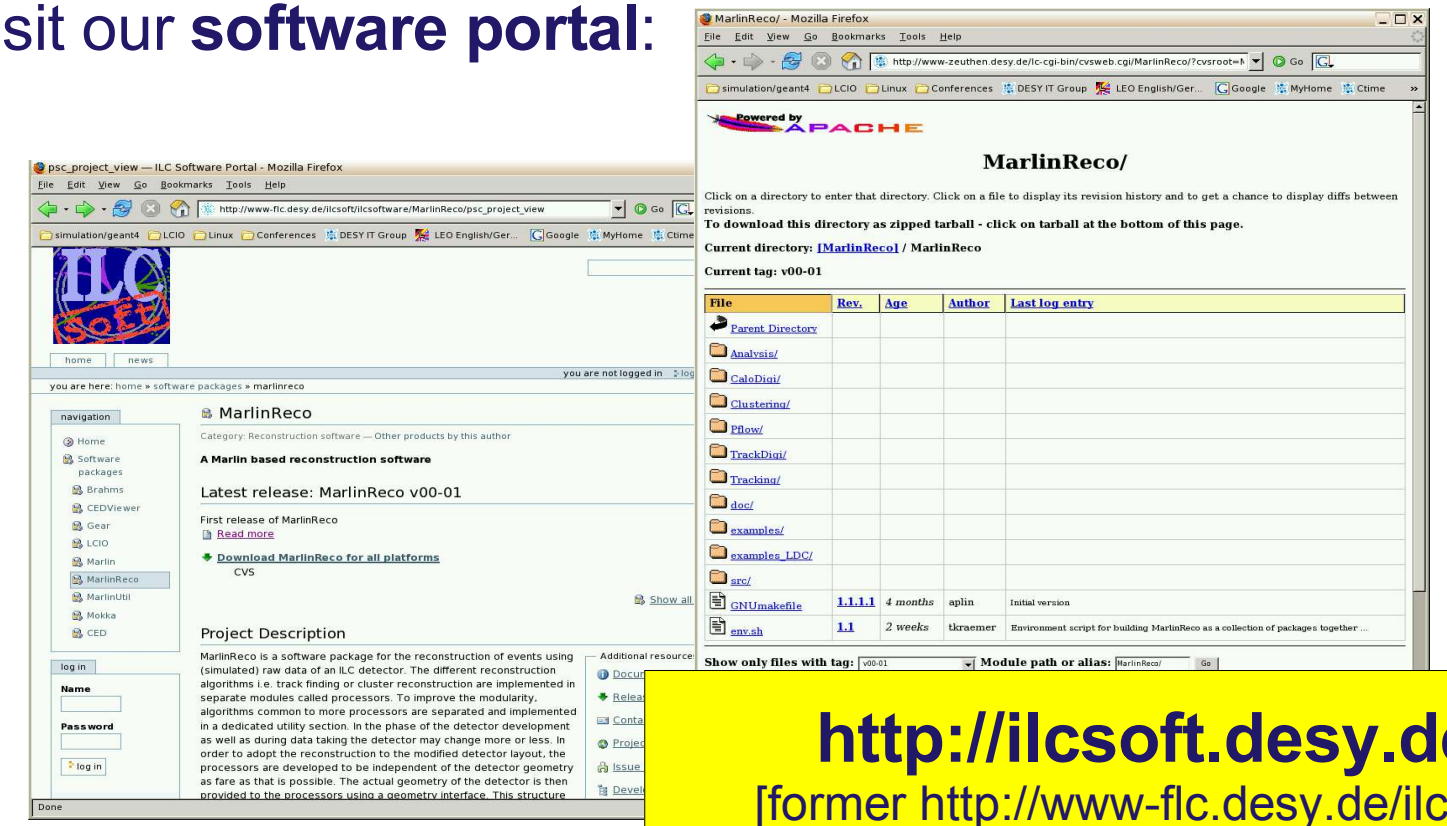
Status of MarlinReco

- **to do**
  - more realistic track digitisation
    - new VTX Digitiser (see talk of A. Raspereza)
  - vertex and forward tracking
    - new stand-alone PatRec for VTX (see talk of A. Raspereza)
  - neutral / charged vertex and kink finding
  - particle ID
  - ...

your input / ideas are welcome !

Status of MarlinReco

- MarlinReco v00-01 available on the web
  - comprehensive manual available
- visit our **software portal**:



http://ilcsoft.desy.de
[former http://www-flc.desy.de/ilcsoft]

MarlinReco on the GRID

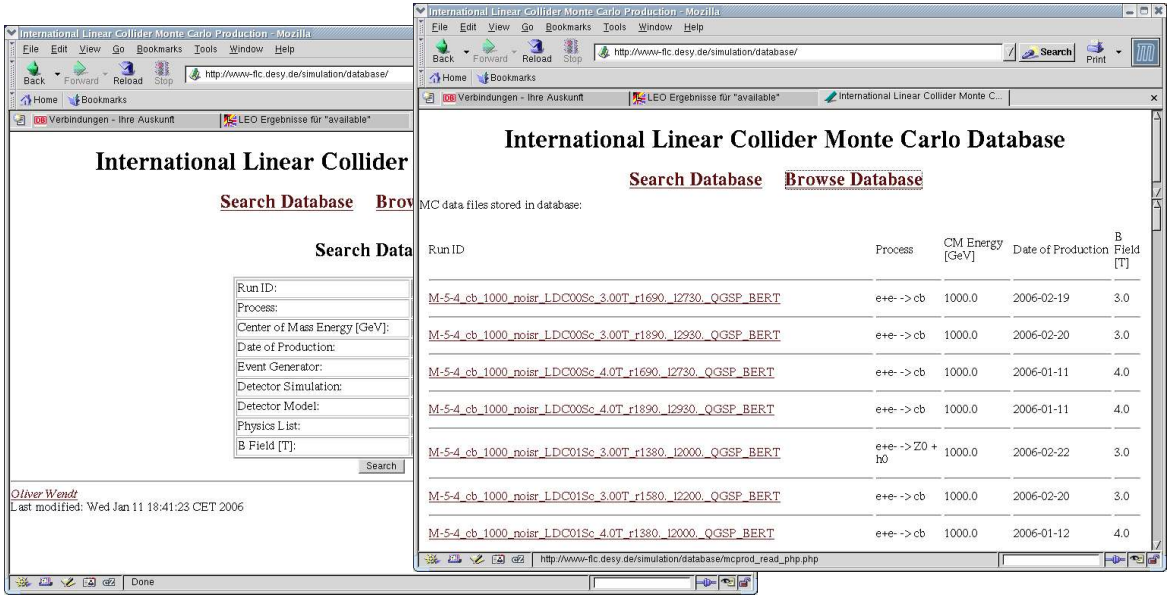
- 'mass production' of events on the **GRID**
  - latest Mokka v05.04, 4 detector geometries and 2 magnetic fields
  - 2 different detectors: **LDC00Sc** and **LDC01Sc**
  - LDC00Sc: 30 + 10 ECAL Layers
  - LDC01Sc: 20 + 10 ECAL Layers
  - same overall thickness of ECAL, HCAL unchanged
  - 2 different sizes of TPC: (**R**)adius and (**L**)enght
  - → overall 8 'different' detectors

B field (T)	LDC01Sc (mm)		LDC00Sc (mm)	
3	R = 1380	R = 1580	R = 1690	R = 1890
	L = 2000	L = 2200	L = 2730	L = 2930
4	R = 1380	R = 1580	R = 1690	R = 1890
	L = 2000	L = 2200	L = 2730	L = 2930

MarlinReco on the GRID

- Physics processes
  - WW, Zh, uds, cb, tt @ 360, 500 and 1000 GeV
  - Z @ 91.2 GeV
- ≈ **450000** events (**500 GByte**) of simulated data
- all data available on the **GRID** for ILC VO members
- meta data, logical filenames etc. stored in **MC Database**

http://ilcsoft.desy.de



MarlinReco on the GRID

- run the reconstruction (**MarlinReco**) on the **GRID**
  - ≈ 70% finished
  - yet not all data analysed

➔ **GRID** is a comprehensive, powerful and 'easy to use' tool

Reconstructed M_{inv} of Z^0

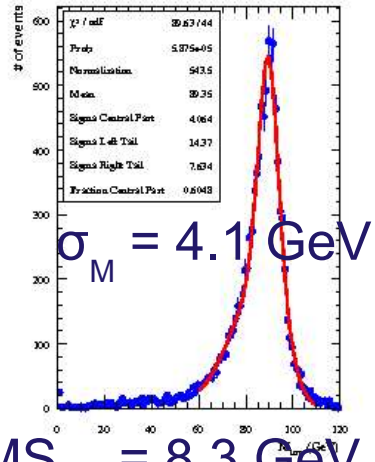
- TrackCheater used in reconstruction
- plot M_{inv} for the 4 different detectors and 2 different magnetic fields
- calculate σ_M and $RMS_{90\%}$ for them
- dependencies on Geometries / magnetic fields

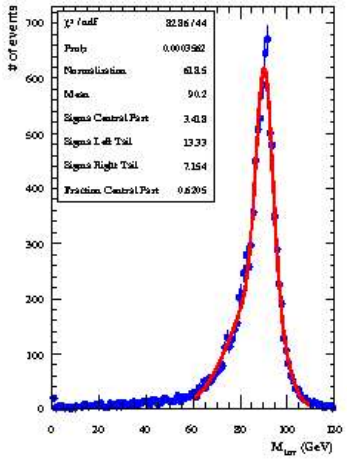
Reconstructed M_{inv} of Z^0

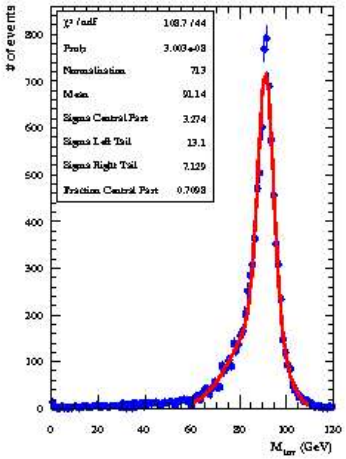
Z pole:

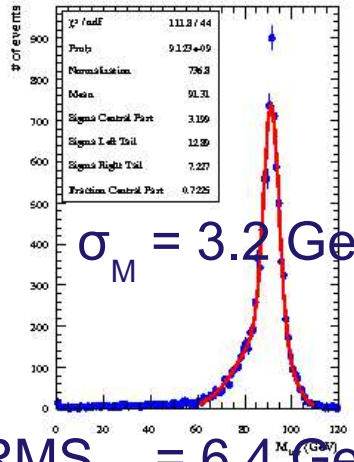
Size →

3 T





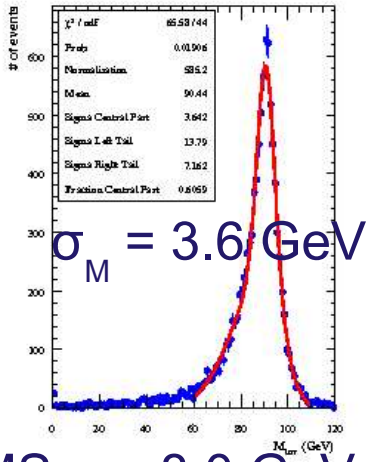


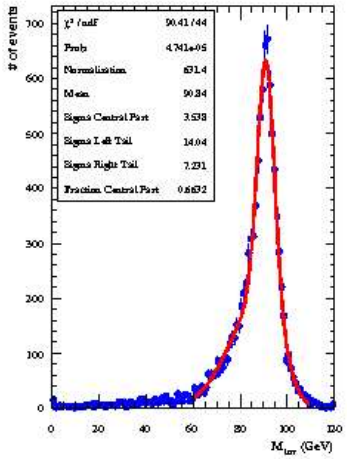


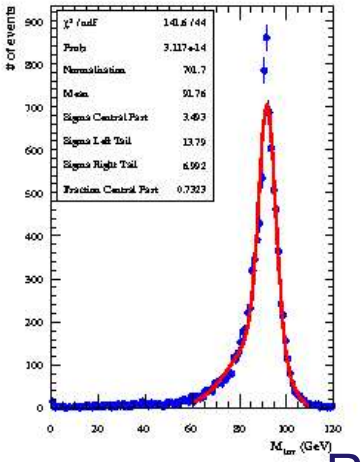
$RMS_{90\%}$ = 8.3 GeV

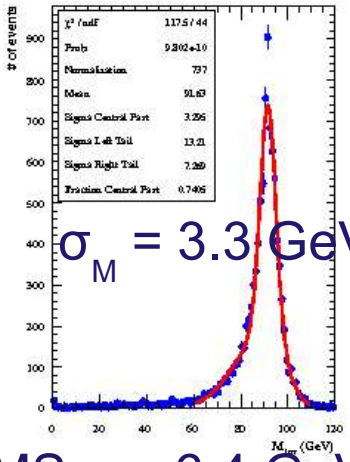
$RMS_{90\%}$ = 6.4 GeV

4 T









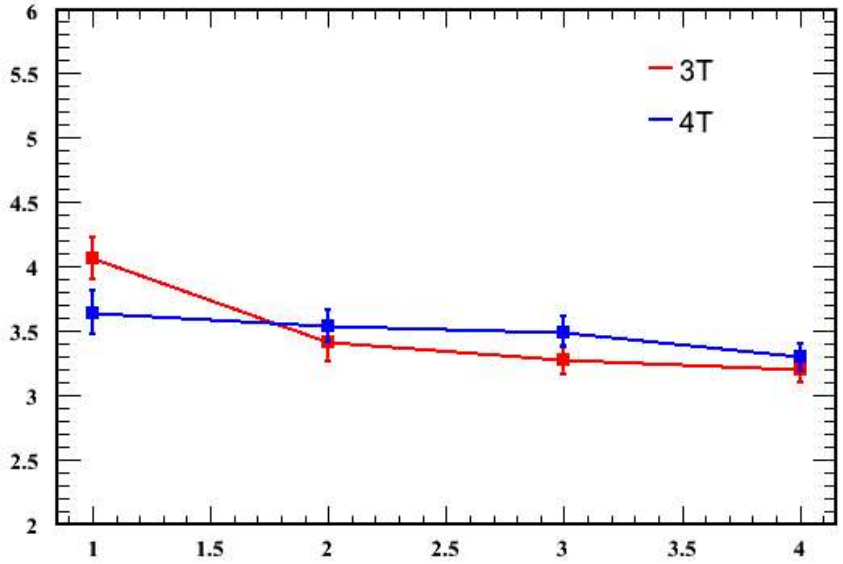
$RMS_{90\%}$ = 8.0 GeV

$RMS_{90\%}$ = 6.4 GeV

Reconstructed M_{inv} of Z^0

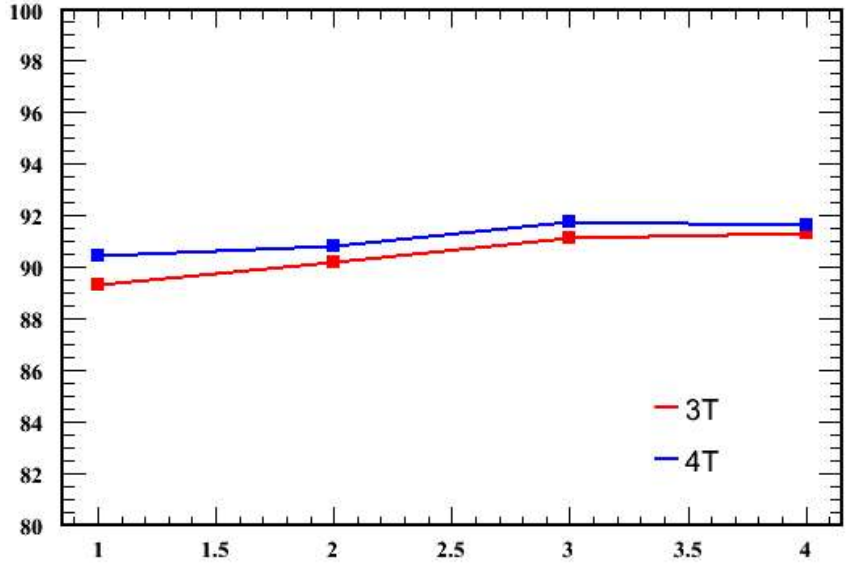
Z pole:

σ_M (GeV)



Size →

$\overline{M}_{rec}$ (GeV)



Size →

Reconstructed M_{inv} of Z^0

- Z reconstruction looks reasonable
  - → good to compare different algorithms in a certain parameter range
  - → nice 'benchmark' plot
- **NOT** appropriate for detector optimisation
  - → CM energy to small (physics processes at 500 GeV)
  - → σ_M not depending on Geometry (within errors)
  - → not so large number of tracks and clusters
  - → not so many overlapping clusters

→ need WW, Zh or ttbar @ 500 GeV and 1000 GeV

ttbar @ 500 GeV

- calculate $\Sigma E_{rec} - \Sigma E_{avail}$
- simple sum over all calorimeter cells (see talk of V. Morgunov)
  - σ_E for 500 GeV ttbar: **12.6 GeV** (LDC00Sc, 4T)
- σ_E of PFlow reconstruction with BRAHMS (SNARK)
  - σ_E for 500 GeV ttbar: **9 GeV**
- energy resolutions of reconstruction with MarlinReco
  - σ_E for 500 GeV ttbar: **25.2 GeV** (LDC00Sc, 4T)

PROBLEM

- ➔ problem in reconstruction code
  - ➔ **not applicable** for **detector optimisation** (same problem for WW)
  - ➔ PFlow concept is **not** the reason (see SNARK)
  - ➔ **might** be a fundamental problem of the cluster-based approach

Summary

- Marlin is a flexible and light-weighted **SW framework**
  - uses LCIO
- MarlinReco is a **reconstruction toolkit** for the ILC
  - full set of needed processors
  - extensible → use / combine with other processors
  - reconstruction of 'mass data' on the GRID
- Reconstruction works **fine for Z Pole**
- **problems** with the reconstruction for **ttbar @ 500 GeV**
  - ➔ studies are ongoing

please feedback to:
ilcsoft@desy.de

Thanks to all developers:
S. Aplin, F. Gaede, T. Kraemer, P. Krstonosic
A. Raspereza (MPI Munich), J. Samson,
H. Albrecht, D. Martsch, A. Vogel
and V. Morgunov

Backup Slides ...

Marlin

Use **PFlow** algorithm for event **reconstruction** at the ILC

- different PFlow algorithms available
- detector optimisation studies ongoing (Geometries, B-Field)

→ need a **simple**, **flexible** and **lightweighted** SW framework

→ easy to install, use and **modify** / **expand**

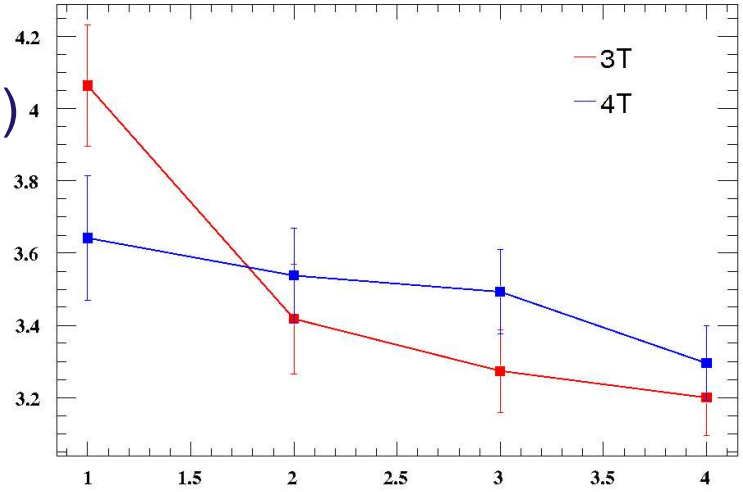
→ use widely accepted OO-language: **C++**

→ based on international ILC data format: **LCIO**

Reconstructed M_{inv} of Z^0

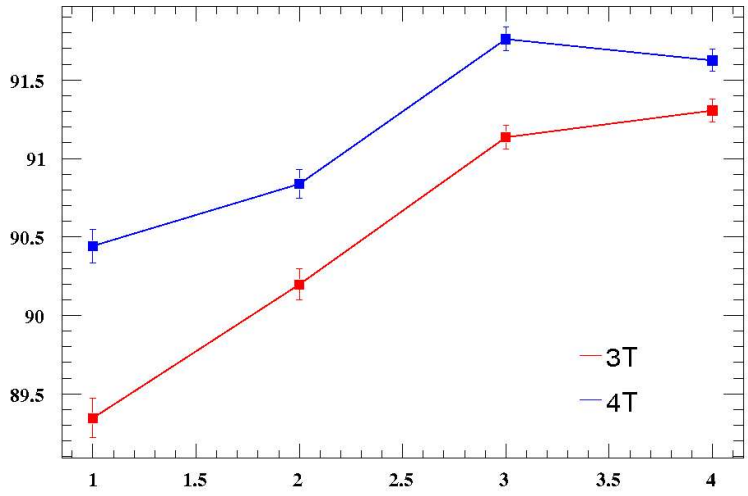
Z pole:

σ_M (GeV)



Size →

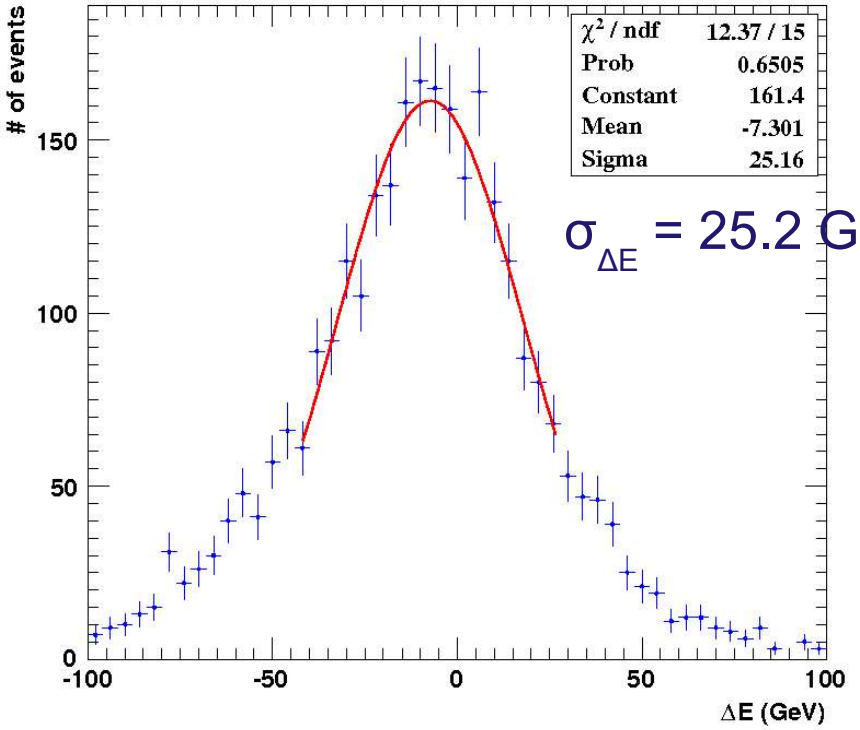
$\overline{M}_{rec}$ (GeV)



Size →

ttbar @ 500 GeV

Calculate whole energy per event and compare it with MC energy



$\sigma_{\Delta E}$ = 25.2 GeV

$\sigma_{\Delta E}$ = 12.6 GeV

$(\Sigma E_{rec} - \Sigma E_{MC})$

preliminary

$(\Sigma E_{Calo} - \Sigma E_{MC})$