

Software Review

Calice Technical Board Open Session

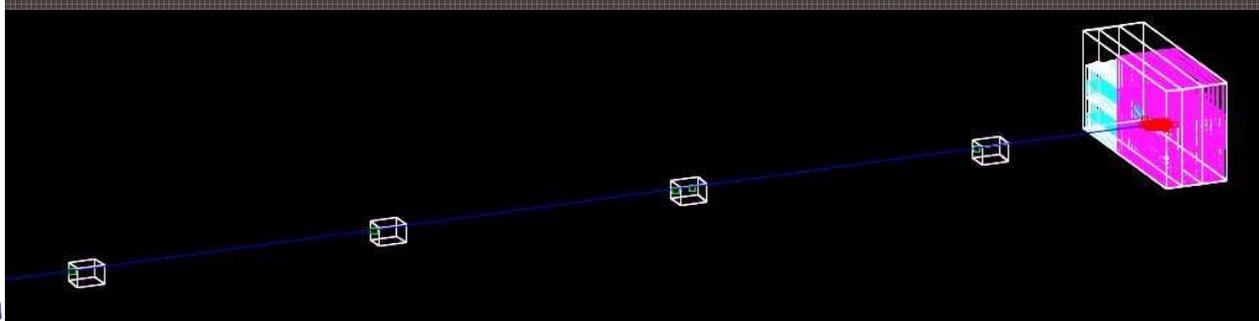
Comments on status relative to
Feb. 2005 TB Review

David Ward
Nigel Watson

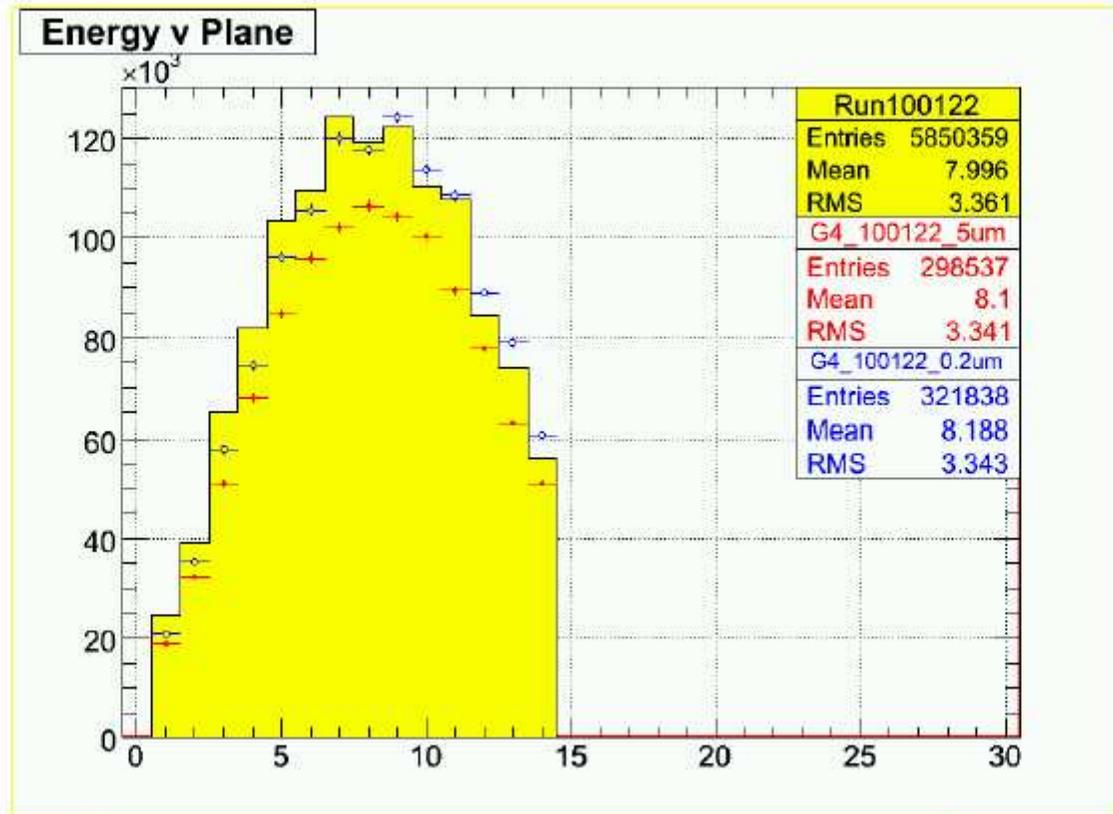
Monte Carlo - Feb. 2005 Recommendation

Each detector group will have to be responsible for and maintaining the geometrical description of their detector within Mokka and for implementing the digitization (noise, crosstalk etc.) as and when necessary. We recommend the use of the DigiSimframework within MARLIN for digitization. Although detailed work may need to await the arrival of data, each group should consider whether the information stored by Mokka is sufficient for their needs.

Status Not aware of any progress or any real problems. Main developments in Mokka have been directed towards global detector design studies; only significant change for test beam has been implementation of drift chambers by new collaborating institute (F.Salvatore, RHUL), already used in some test beam studies



Shower longitudinal profile



Showers seem to be a bit too deep in G4?

Analysis Framework - Feb. 2005

Recommendation

Work on the lightweight "intelligent" decoding of the data into LCIO objects needs to start expeditiously (**action** P.D.Dauncey, G.Mavromanolakis, R.Pöschl, D.R.Ward). Aim to agree on data content by NIU Calice meeting, and have a first version of code by end March. We recommend the use of MARLIN as the analysis framework. Individual processing tasks, such as mapping, calibration, alignment, histogramming, should be packaged as separate MARLIN processors.

Status Progress slow, but finally ~20 ECAL runs were converted at DESY around the end of August. Believe the same software framework is being used for the HCAL module tests.

Database - Feb. 2005 Recommendation

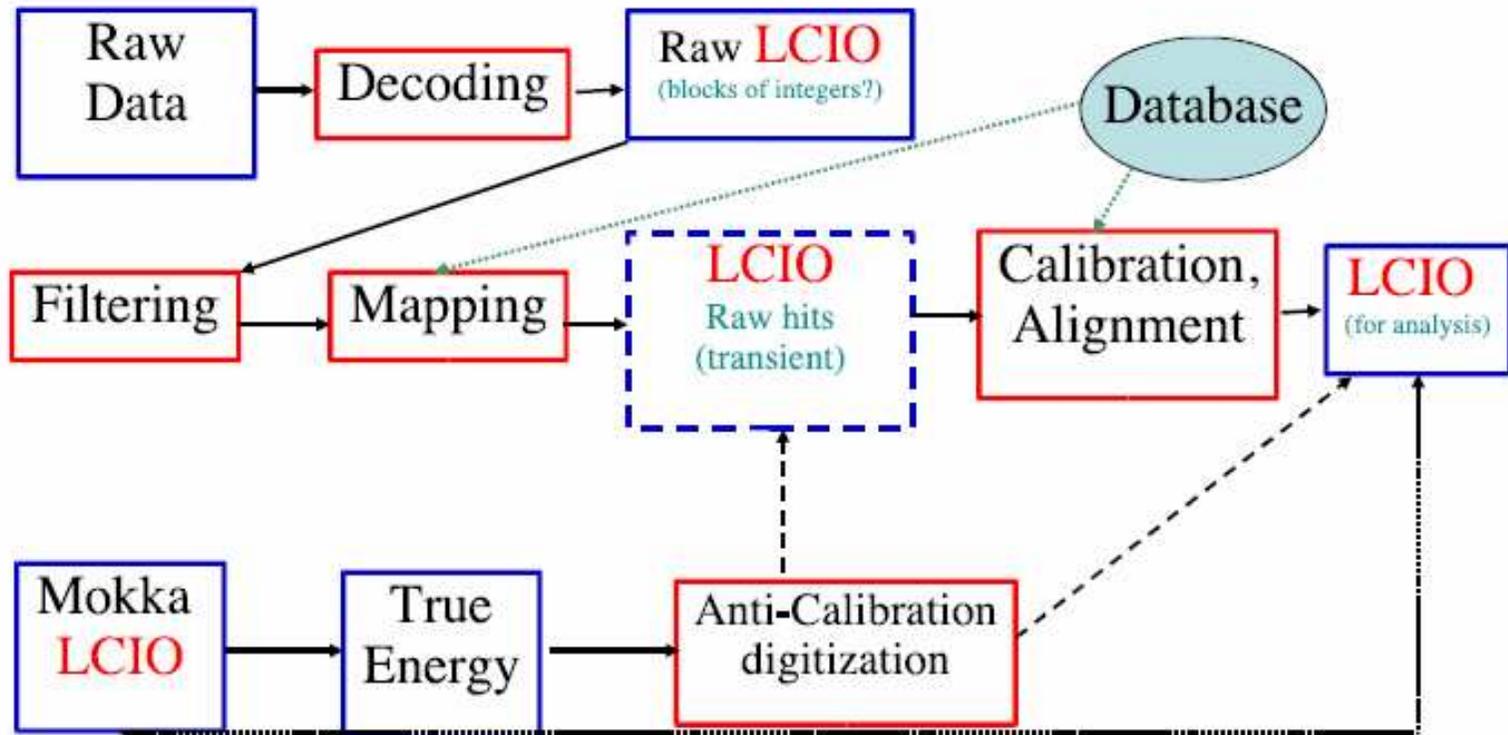
The use of the LCCD package to access a MySQL database in the LCIO/MARLIN framework is recommended. A database manager will need to be appointed.

Status DESY have set up a server machine (flccaldb01.desy.de) to hold the database. Can access from remote sites, after IP address registered at DESY (so far as I know, only Cambridge and Imperial have done so). This works.

- However, to keep learning curve for analysis as gentle as possible, ideally avoid routine access to database for "users". Possible simple scheme would be to implement trigger status bits into event header - satisfactory for most users, more detailed studies closer to the data would need (+ be able to get) access as necessary
- Definition of contents of database? Include e.g. conditions data such as readout of 3D stage

Dataflow in CALICE Testbeam

LCIO as backbone of Testbeam Analysis



SimCalHits

[R.Poeschl]

Data Storage - Feb. 2005 Recommendation

The data (native, raw LCIO and processed LCIO) will be stored in the dCachemass storage at DESY. All members of Calice need to be informed how they can access to these data. The preferred method of access being Grid-ftp. Write access needs to be restricted to a very small group of experts (to be identified)

Status This has been done for the converted data. Files for runs 100050 - 100224 have been registered in the Grid so far. Unless you have a DESY account, need to use Grid tools to access data (need to be member of the "calicevo"). Details announced to Calice-SW mailing list, 10-Oct by Roman P. Have managed (since yesterday) to copy the single file using grid tools, run analysis, accessing database etc. The system is just about there now!

Calice-SW Membership

[Email addresses removed from web version of talk]

<http://www.listserv.rl.ac.uk/archives/calice-sw.html>

List members free post, non-members only distributed after owner approval

Web archive of postings

Data Storage - Further Remarks

■ Actions:

- ▶ Facilitate entry to grid usage for Calice members
- ▶ Make use of facilities in addition to data retrieval
 - ⇒ Simulation
 - ⇒ Data analysis
 - ⇒ Conversion
- ▶ Makes grid overhead more worthwhile for end users (cf. sftp)
- ▶ Need a working example (for guinea pig testers), then open up
- ▶ Investigate data storage logistics for TB if at CERN - someone to follow up, easier if we have CERN member of Calice??

Code Sharing - Feb. 2005 Recommendation

Authors of code are strongly encouraged to store their work at the CVS repository recently established at DESY-Zeuthen.

Status The data conversion code is all there; not much else yet. Perhaps because there's not much to put there?

- Remember: after writing code, it will only be used/accepted if easy to do so, available, etc.
- Single Marlin processor can be 1.cc, 1.hh, +readme +demo steering file
- Code has to be "fit for purpose" - remember this is short-lived test beam project
- If you need help with making your code available, please contact: drw1@hep.phy.cam.ac.uk

Code Sharing

Reconstruction Processors

- SimpleHitSearch:
 - Pedestal/Noise determination
 - Zero suppression
 - Calibration
 → Calibrated hits with 3D coordinate.
- DriftChamberReconstruction:
 - Edge selection
 - Track fit
 - Visualisation
 → a track.

[G. Gaycken]

Getting started with MAGIC

- Install **LCIO** (≥ v01-05) and **MARLIN** (≥ v00-07).
- Download **MAGIC** tar-ball from <http://www.hep.phy.cam.ac.uk/~ainsley/MAGIC/MAGIC-v01-02.tar.gz>
- **Two directories** and a **README** file (read this first!).
- The **clustering** directory contains the cluster-reconstruction (and cluster-truth) code (i.e. all processors and steering file mentioned earlier).
- Takes **.slcio** input files containing **CalorimeterHits** (data) or **SimCalorimeterHits** (MC):
 - must be generated with hit-positions stored, i.e. `RCHBIT_LONG=1` (data) or `CHBIT_LONG=1` (MC);
 - collection names must contain the string "ecal" or "hcal" (in upper or lower case, or in some combination of these) to identify the type of hit (for energy-threshold application).
- Produces **.slcio** output file with cluster-related collections added:
 - `CalorimeterHits` ⇒ hits above energy threshold;
 - `CalorimeterHitRelationsToSimCalorimeterHits` (MC only) ⇒ pointers to original simulated hits;
 - `CalorimeterStage1Clusters` ⇒ clusters after stage 1 of algorithm;
 - `CalorimeterStage2Clusters` ⇒ clusters after stage 2 of algorithm;
 - `CalorimeterStage3Clusters` ⇒ clusters after stage 3 of algorithm;
 - `CalorimeterTrueClusters` (MC only) ⇒ true clusters;
 - `CalorimeterTrueClusterRelationsToMCParticles` (MC only) ⇒ pointers to original MC particles.
- The **examples** directory contains example analysis code which performs simple manipulations with the clusters (e.g. processors which add calibrated energies to clusters, produce the plots shown earlier, calculate the reconstruction quality... and an accompanying steering file).

[C. Ainsley]⁹ General CALICE meeting

gNIKI

general Nodes Interfaced Klustering Implementation

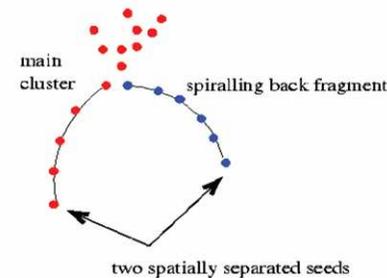
Description
 gNIKI stands for general Nodes Interfaced Klustering Implementation.
 It is a clustering algorithm which operates in two main stages, combining both types of hierarchical clustering, divisive and agglomerative. The former stage is based on minimal spanning tree theory, the later can be a concrete energy flow algorithm.
 The hit algorithm implements a so-called "top-down and then bottom-up" approach to agglomerative clustering. In the first stage the entire set of hits is first considered to be a single cluster, the minimal spanning tree, which is then broken down into smaller clusters. Then the second stage of the algorithm is applied to these clusters to merge the final reconstructed clusters of the event.



[G. Mavromanolakis]

Treatment of Splitted Clusters

- Clustering proceeds in ascending order of hit distance to IP -> cluster can be artificially splitted if it contains tracks spiraling back



For terminating hit "i" of spiraling back fragment candidate (# hits < `nMergeSpiral`), calculate merging parameter w.r.t. nearest hits "j" ($r_{ij} < RCut$) belonging to high multiplicity clusters. If $\min(s_{ij}) < sCut$, merge fragment candidate to the corresponding high multiplicity cluster

[A. Raspereza]⁷

Documentation - Feb. 2005 Recommendation

Documentation needs to be improved, and a central point of access to documentation (e.g. a web page) should be established.

Status This should remain an aspiration, which we will probably never achieve. We do at least have a Calice software web page.

■ Actions

- ▶ Write a pedestrians guide, Calice-TB-HOWTO
- ▶ Modifications to this Calice-SW web page, email to drw1@hep.phy.ac.uk
- Signup to calice-sw mailing list -
- Information sharing: use the mailing list
- There is **NO** collaboration wide mailing list
- Suggest that we **set one up, a la calice-sw**, with web archives
- Propose we use it, frequently (e.g. TB/SB news, collaboration wide requests for speakers, TB shifts, etc.)

Summary

- We just about have a working system along the lines recommended in February
- DRW may be almost the only user at present outside DESY
- George has been working with his independent system (most of which has been wrapped in a Marlin framework) and Götz has his own variant.
- Would be more productive to converge here
 - ▶ “Natural” solution is to submit individual, independent processors to Calice CVS, allow users choose the parts of packages they need
- Main risk - people won't know how to use the tools when we next start taking data.
 - ▶ Too much expertise resides with one (very good) person (Roman)
 - ▶ We need people to look at the data as soon as possible after they have been recorded, in case of mistakes
 - ▶ Having much improved communication would help here
 - ⇒ They are shared data, joint responsibility to ensure high quality
 - ⇒ If we are all kept informed about data taking it becomes easier to contribute from afar.

Comments

- Need to ensure data conversion to LCIO for hcal is ready well in advance of combined TB at CERN
- Useful to have systematic data storage at DESY dCache, but more benefit if running jobs on grid, otherwise just a complicated version of sftp
- Need to ensure that scintillators planes are included in Mokka before test beam
- Clear that would benefit from more people looking at data
 - ⌘ Need to make s/w system pragmatic for data analysis
 - ▶ Remember that all of this is relatively short-lived code
 - ▶ “Fit for purpose” but no more