

Common ILC Data Format (From Raw to Physics Data)



Roman Pöschl
LAL Orsay



- Introductory remarks
- Calice Testbeam Data Taking
- Status of the ILC Software w.r.t 'real' data
- Role of upcoming R&D projects Module
- Summary and Outlook

ILC Workshop Valencia/Spain November 2006

Prologue – Citations from the LDC CDR

“The DAQ System will also benefit from the use of common operation system like Linux and high level programming languages already at the event building and finding stage **making the separation of online and offline code obsolete and therefore avoids the need to re-write code for on-line and off-line purposes.** This results in a more efficient use of common resources”

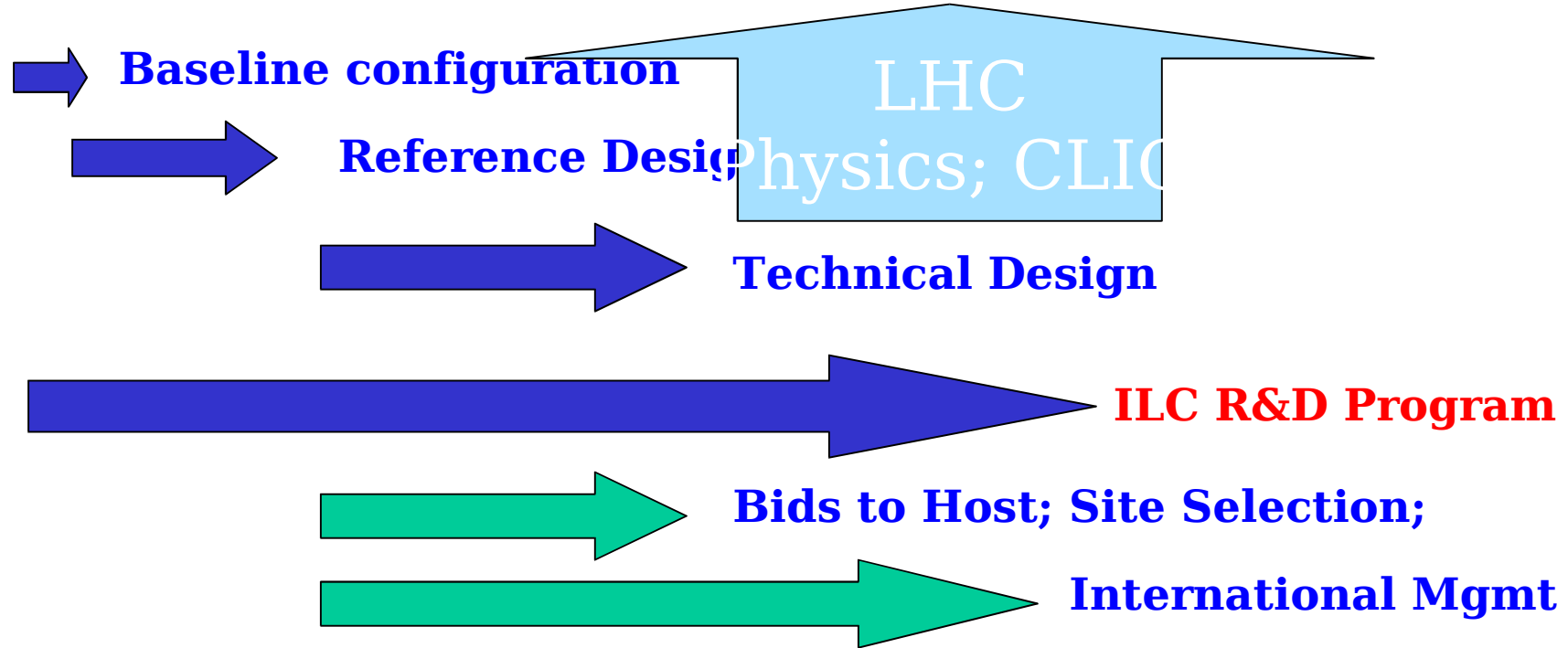
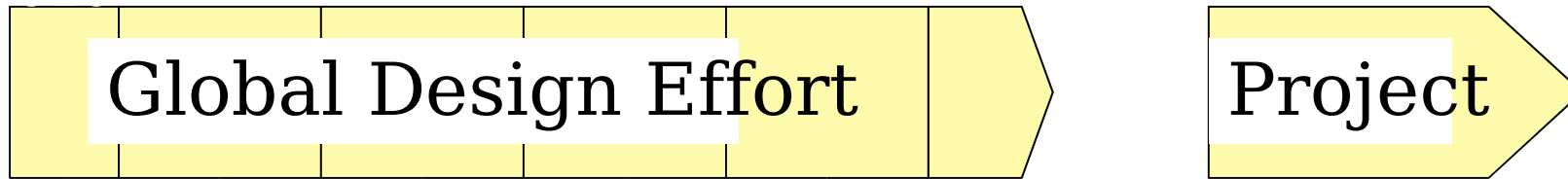
“The machine and beam conditions are vital for the high precision physics analysis ...
a common data storage model should be used”

What is true w.r.t. the machine is even more true for the detector



GDE Timeline

2005 2006 2007 2008 2009 2010



(Hopefully) tight time line for the ILC demands highly integrated R&D effort

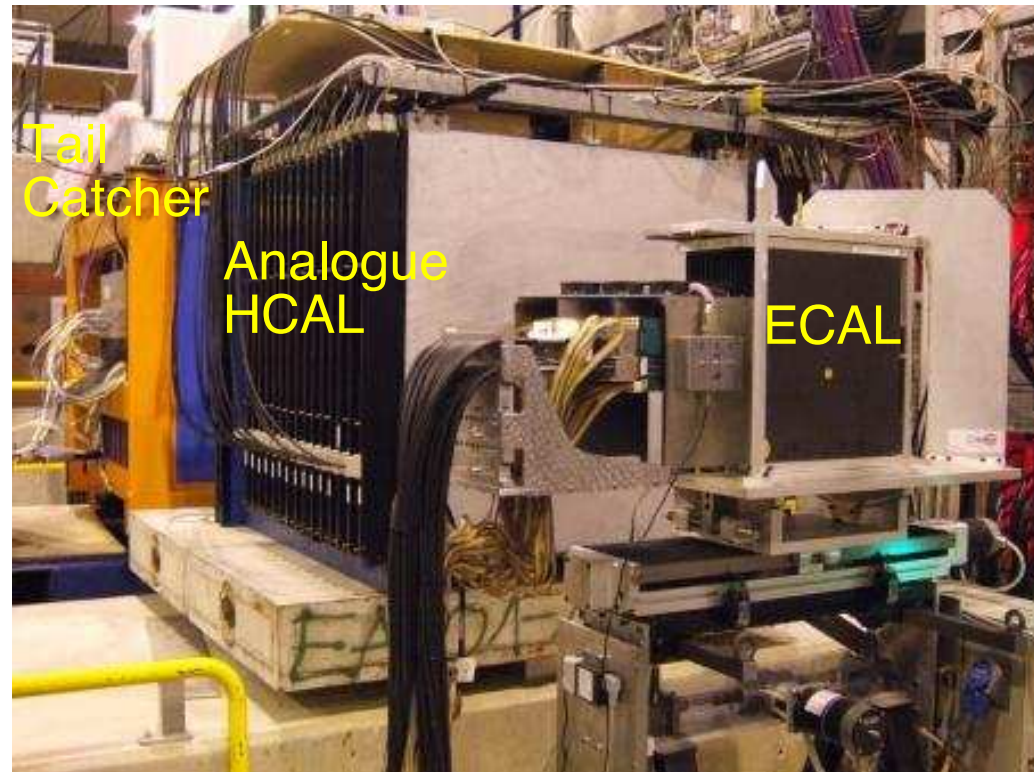
CALICE Testbeam Data Taking

CALICE collaboration is preparing/performing large scale testbeam
'Real' Data are taken as we speak

Testbeam Setup at CERN

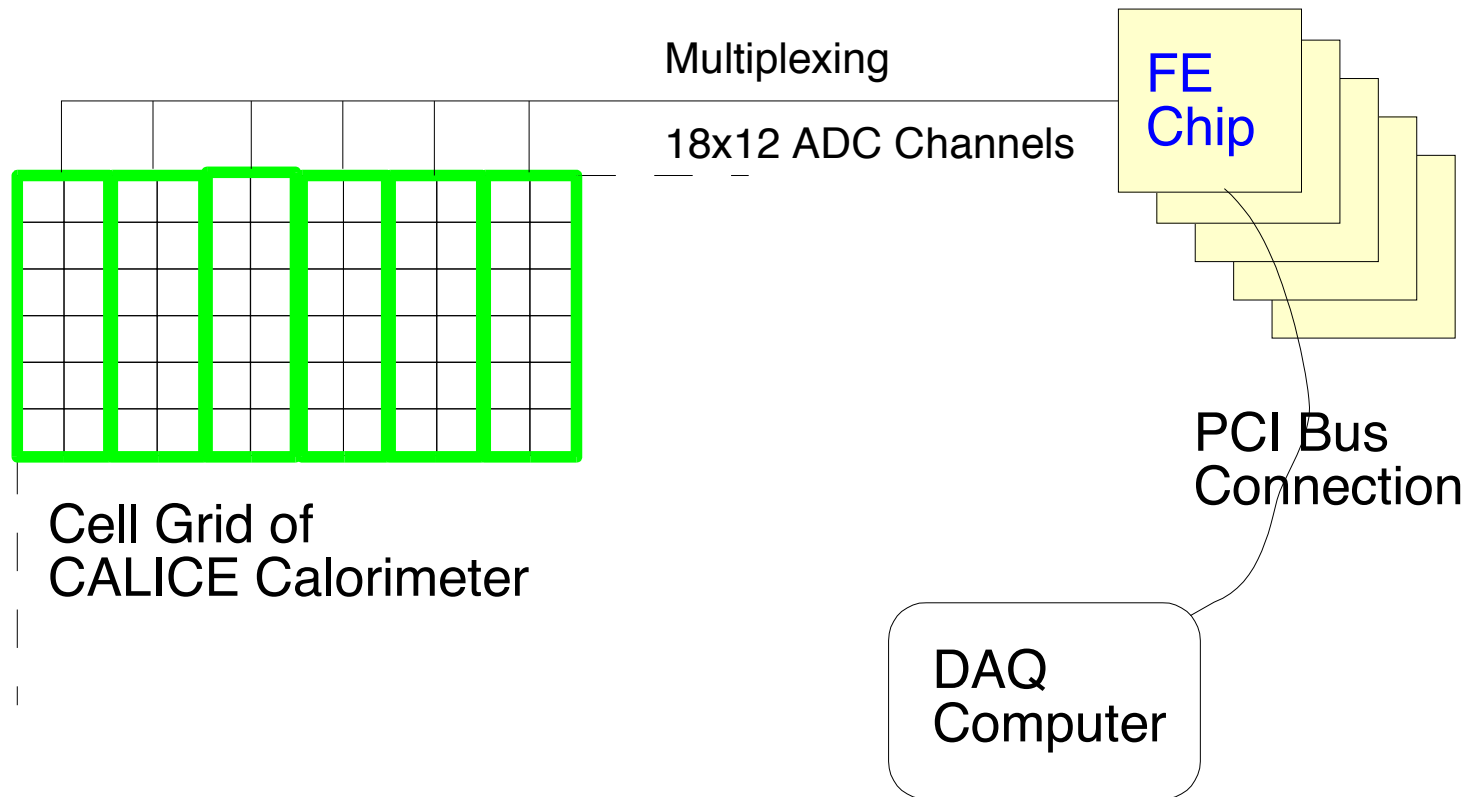
Testbeam program poses software “challenges”

- Data processing from Raw Data to final Clusters in user coherent way
- Handling of Conditions Data
Detector Configuration
Calibration, Alignment etc.
- Comparison with simulated data
'Physics' Output



O(15000) calorimeter cells
readout by Calice DAQ
No Zero Suppression

CALICE DAQ Scheme – Poor Man's View



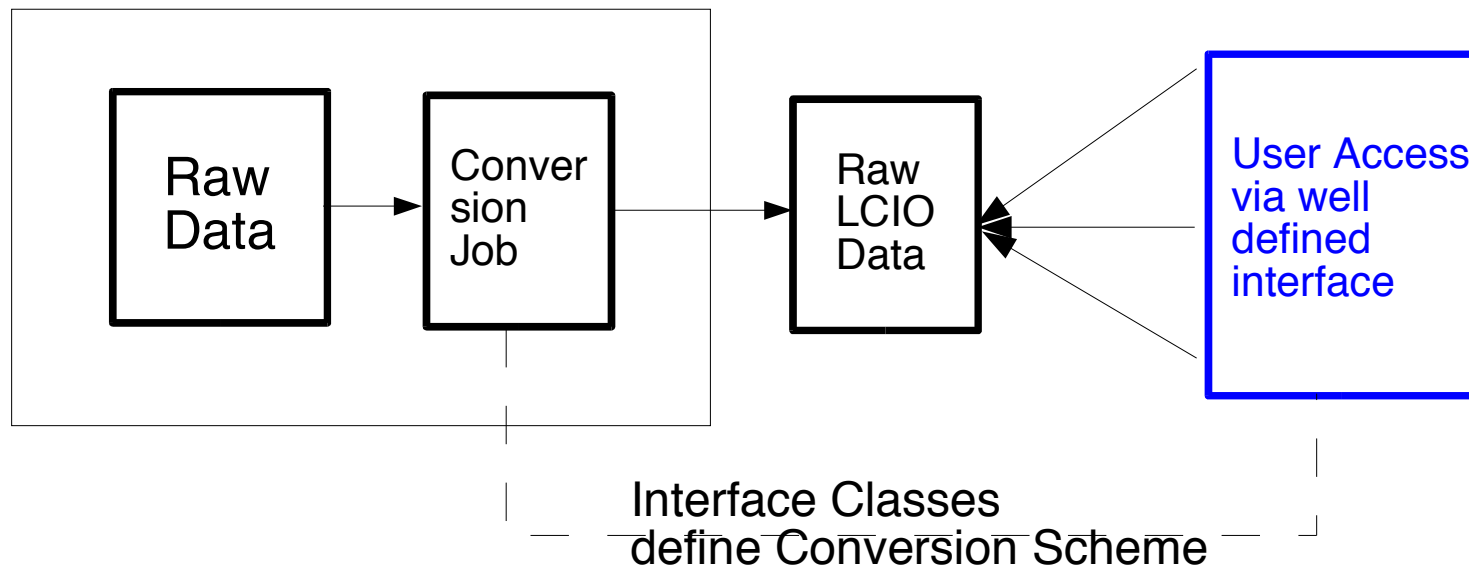
- DAQ is organized 'hardware friendly' – optimized for Speed
Data received and stored as sequence of 32 bit integers
- 'Many' people should get involved in calibration/monitoring
- Idea is/was to process data using existing ILC software tools
benefit from global development and
deliver input for improvement of ILC software!!!

Raw Data Conversion to LCIO

LCIO is data model widely used in ILC Studies

Main Strategic Decision: Raw Data should be available in LCIO Format

- Requirements:
- 'Intelligent' Conversion from Raw Data to LCIO Raw Data
 - Provide all Info on Raw Data also in LCIO Raw Data in a user friendly way



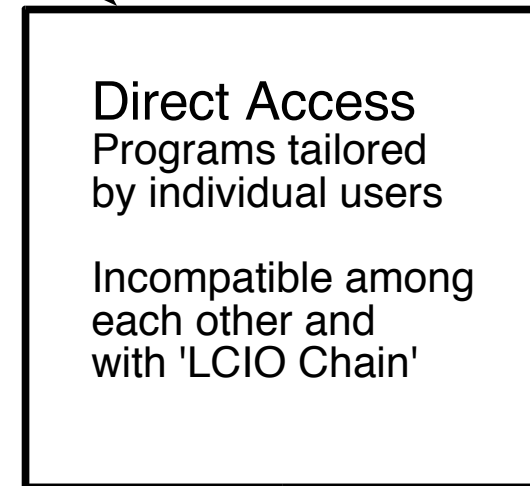
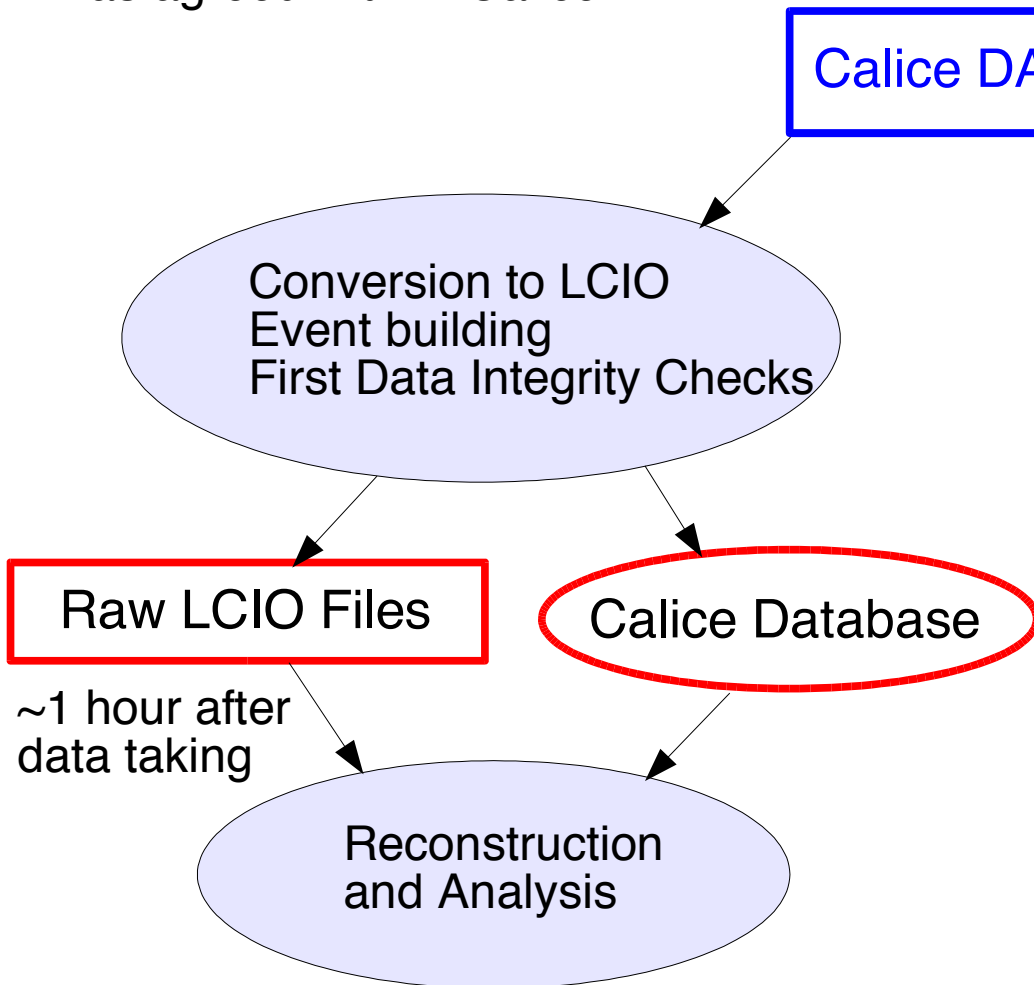
Interface is to be completely decoupled from online software

Dedicated Interface Classes are defined using LCIOGenericObjects

CALICE Dataprocessing

The desired chain
using LCIO and MARLIN
as agreed with in Calice

(Part of the) Reality



Individual Approach with
advantages and disadvantages
of a private hack

Complex chain with advantages
and disadvantages of a structured
approach

Calice DAQ and ILC Software framework

- CALICE DAQ was designed as ILC Software was at its real infancy (~2004)

No concept to treat DAQ data types in a way compatible to the ILC Software concepts, 'LCIO' (or whatever) was not part of the DAQ Concept

- Changes to LCIO Conversion can/do happen only after DAQ changes

Severe changes to the DAQ need to be integrated carefully into the the Conversion
Conversion has to run permanently and has to treat millions of events without crashing

Delay of availability of converted data

- Converted files might not contain all data types needed for analysis
Iterative process

Pros and Cons using ILC Software for Calice Testbeam Data

Pros

Benefit from existing tools/features for/of ILC Software
e.g. LCEvent allows to gather information on event

Newcomers can work in one software framework for testbeam and physics studies

Define at an early stage the of the ILC R&D the needs for a coherent data processing
Coherent s/w concept at time of ILC Detector TDR

Not just guesswork!!!

Is in the spirit of the LDC CDR!!!!

Cons

Need to wait for converted files
No quick turnaround in particular during development of DAQ and tests
Needs tight communication between DAQ and s/w developers

Some overhead generated by usage of ILC Software

Source of (potential) errors unclear

Status of and of ILC Software w.r.t 'Real' Data

- Existing software mainly developed for simulation studies

For more information on ILC Software packages see ilcsoft.desy.de

- LCCD

conditions data framework was/is first approach to store data like Calibration Constants, Detector Configurations Currents Temperatures and to read them back into e.g. your MARLIN Analysis

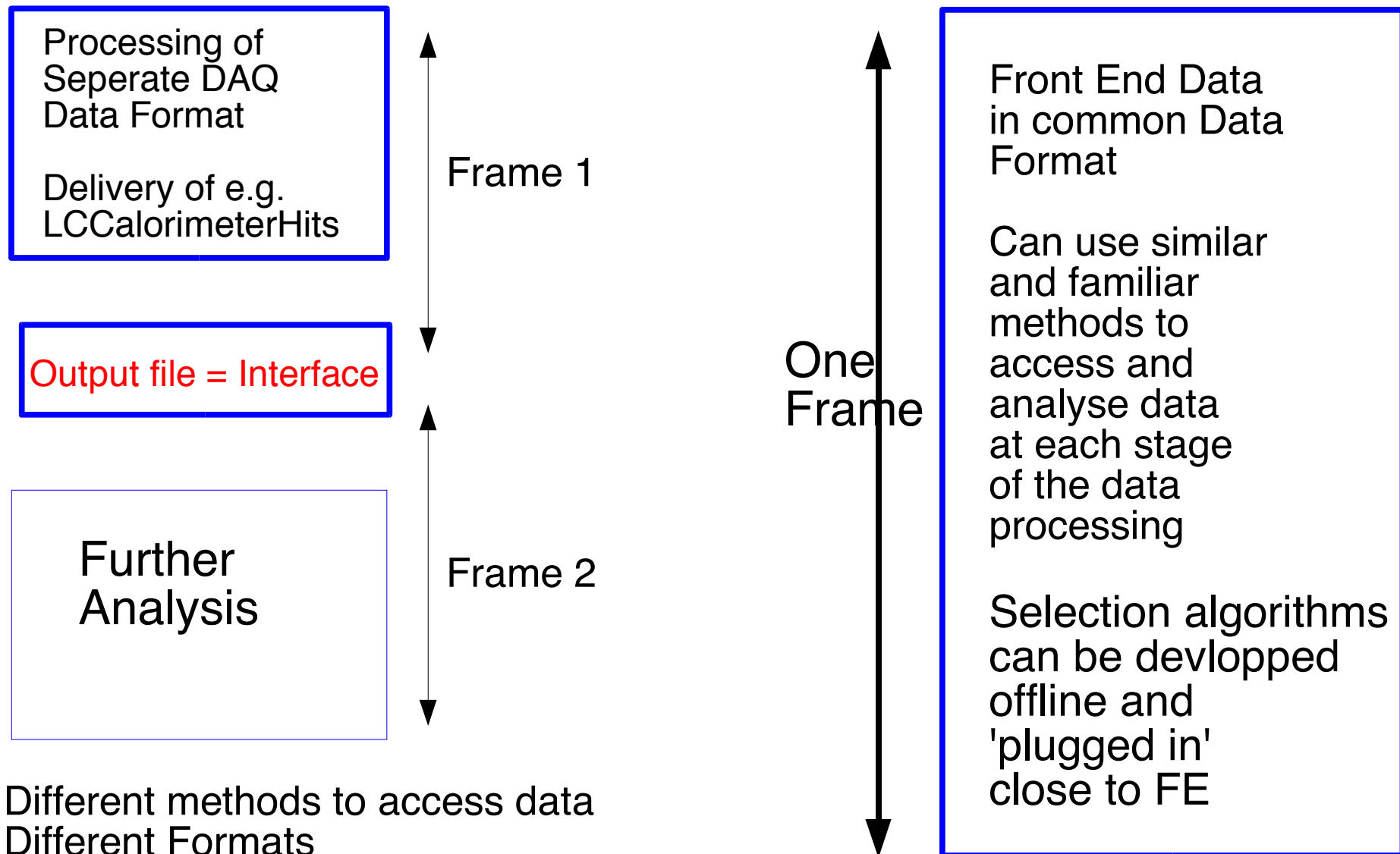
- LCGenericObjects

Allow for an arbitrary definition of user defined classes within LCIO
Application should be the exception
Extensive use in calice LCIO conversion
Lack of type safety, 'no' official LCIO DataType

Not much though ...

Interface between 'Online' and 'Offline' World

Decision to be taken on first data format visible to the world



Different methods to access data
Different Formats
Impossible to access information
created/used by the other frame
Roman Pöschl

(Basic) Issues for common Data Processing Chain

'High' level data are well addressed CaloHits, Clusters, Tracks etc.

Need to develop a software concept for treatment of low level data

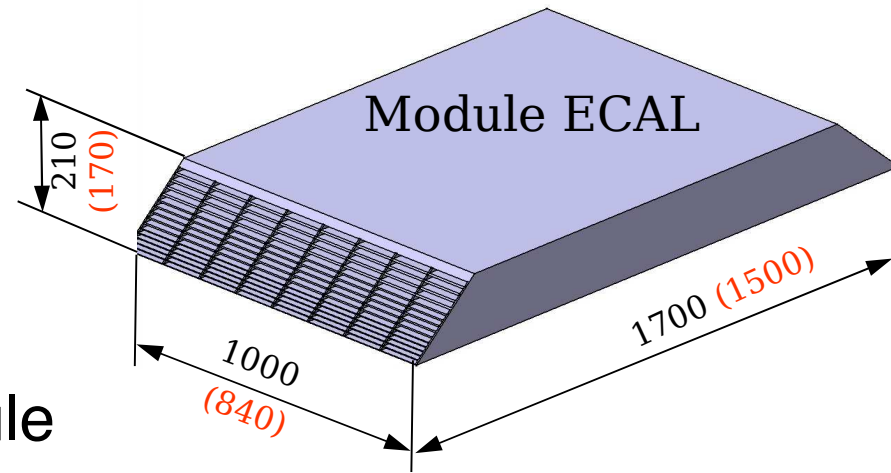
- **Avoid several (incompatible) chains on analysis**

One source of information

- 'Model for 'close to detector' data
Hardware information needed to judge on data quality
Extension
LCIO as backbone ?!
- Software models flexible enough to react to changes
in DAQ system
- DAQ has to be developed true to an agreed scheme
and embedded into overall software development

Coming close to the ILC – The EUDET Module(s) and a common Calo DAQ

EUDET module of Ecal
is $\sim 1^{\text{st}}$ Modules of ILC
Detector



Common DAQ for EUDET module
(and other EUDET Calos)
may be \sim (Calo) DAQ Concept for ILC Detector

Ideal testing ground to qualify concepts for whole
ILC Dataprocessing
There will be also Testbeams!!

Develop and optimize ILC Software e.g. LCIO for 'Real Data'

Items to be Studied

- Where to implement the (LC)Event building?

Data could be written into a shared memory and of a DAQ PC read e.g. by an event builder software (e.g. the lcioconverter)
This part could then also be used for online monitoring

Speed is an issue, 'High' level software should not compromise Data taking rate

Building of LCIO Events with calice lcioconverter
~ 100 Hz (on a standard grid worker node)
no event selection but first quality flagging.

Input from experts needed

- General Concept of Event Building?

SubEventBuilder i.e. Passing 'prebuilt' subdetector information to a General Event Builder or one Event Builder
Possible to test if more than two units are put in a test beam

- Development of event selection algorithms and implementation in event building

Testbeams have 'rich' structure, double events etc.
Event Selection would be studied with MC but need maybe further input from detector data

- accommodate the needs of a well designed software to the necessity to implement changes and fixes

In particular during the development of both DAQ and ILC Software

Summary and Outlook

- Calice uses ILC Software framework for processing of Testbeam Data
LCEvent allows for gathering event information which arrive a bit as a jigsaw
Runs stable over millions of events but success/acceptance suffers from non-integration into Calice DAQ Concept
'Newcomers' can participate easier in Testbeam analysis
- EUDET DAQ and EUDET Modules allow for development of and real life tests of coherent scheme for ILC data processing
DAQ Development and Software Development NOT separate entities
- Integration of high level analysis algorithms into DAQ is part of LDC Detector Concepts
All relevant detector data must be available in one format
- Needs tight communication between 'online; and 'offline' world
Let's start today