

CHEP 2004 Interlaken

Summary of the Parallel
Session: “Core Software”

Giacomo Govi IT/DB



Institutions/projects

Institutions

- CERN (12)
- FNAL (4)
- LBNL (4)
- SLAC (3)
- GSI (2)
- LAL (2)
- Northeastern University (2)
- Maryland University (2)
- DESY (2)
- Cornell University (1)
- Université de Grenoble (1)
- Universidade Lisboa (1)
- INFN (1)

Projects

- LCG (8)
- ATLAS (4)
- ROOT (3)
- CMS (3)
- GSI
- ALICE
- FNAL Accelerator
- IceCube
- Pierre Auger
- Cleo
- Babar
- EDG

Topics

Generic Frameworks

Language interoperability/extensions

Analysis tools

Data processing frameworks

Persistency

Math Libraries

Infrastructure/procedures for development

Generic Frameworks

Abstract decomposition of application modules

- Definition of components, protocols
- Assembly infrastructure
- Properties
- Dynamic/Static configuration

EMS System

- Target to stream data processing
- Java based

SEAL Component Model

- Target to any C++ composite application
- Provide basic framework services
- Context based wiring

Language interoperability/extensions

- Object dictionary
- Generic programming
- Python scripting appeal
- Bindings

SEAL reflex package

- C++ reflection
- Implemented for very general usage

PyLCGDict + PyRoot

- Python/C++ binding
- Based on LCG/CINT dictionaries

- AOP programming

Aspects

AOP Extension in HEP

Analysis tools

Applications for interactive analysis and data visualization

- API universality
- Languages/platforms
- Web interfaces
- Interactive tools

AIDA – JAIDA

- Defines a general API
- Implementation available for C++,Java,Python

OpenScientist

- Assembly of external tools
- Uses native Graphic packages on several platforms
- OpenPaw

Data processing frameworks

- Tasks: reconstruction, simulation, analysis
- External dependencies
- Analysis on distributed environment

Generic application

Go4

- Modular structure, configurable
- External algorithms

Experiment specific offline frameworks (LHC)

Athena

- Based on Gaudi

AliRoot

- Includes all ALICE offline software

Iguana

- Composite framework based on plug-in modules

Data processing frameworks

Experiment specific offline frameworks (non LHC)

- Discrete detectors with large sensible surface
- Laboratories in remote location
- Large time structure of events
- Algorithms evolving rapidly

IceTry (IceCube)

- Used in both online and offline
- Multi path processing
- Algorithm choice at run time

Auger

- C++
- Handles access to event data and detector data
- Provides geometry utilities
- Allow plug-in of external algorithms

Persistency

Frameworks for data store

- Persistency technologies, evolution
- Transient to persistent shape
- Data distribution

LCIO

- Targeted for simulation and reconstruction
- Java, C++, F77 API
- Extendable data model
- Proprietary I/O format

POOL

- LCG Persistency framework
- API neutral on storage technology
- Support for file and metadata catalogues
- Integrated in three LHC experiments

Persistency

Root I/O based technologies

- TTree feature enhancements
- Data I/O interoperability
- Access to data in root files from pure Java programs

New developments in TTree area

- Large file support
- Optimization/ enhancement of queries
- Access to RDBMS from TTree

XML I/O for Root

- XML exchange format between applications

Java Root I/O

- Support for fast reading from root files

Bitmap indices for multidimensional queries

- Query optimization

Persistence

Frameworks for meta data and detector data

- Versioning
- Metadata/ Data selection
- Architecture
- Multi-language/multi platform access
- Concurrent storage access
- Geographic distribution

Event Store

- Focus on data analysis handling

AMI

- Java based three-tiers architecture
- Data set bookkeeping catalogue/Tag collector

CMS DDD

- Stores pure detector technical data
- To be correlated with slow control and condition data
- Previously MySQL, work on going on Oracle

Persistency

Frameworks for condition data

- Target: data varying with time (calibration,alignements)
- Storage technologies
- API and Schema definition
- Data handling

LCG Condition database

- Common solution for LHC experiments
- Unified API for multi-technology backends
- Integration with POOL persistency foreseen

ATLAS vs. LCG Condition database

- Based on MySQL db

BaBar CDB

- Based on Objectivity

Math libraries

Packages with Math functions, operations, fitting tools, algorithms

- Target: simulation, reconstruction, analysis
- CernLib replacement
- Re-implement some algorithm with OO design
- Minimization packages

SEAL MathLib

- Inventory of HEP Math components from GSL, CernLib
- Validate and integrated C++ components from GSL, CLHEP, Root
- Minuit re-implemented in C++

ZOOM

- Focused on Fitting and Minimization

Infrastructure/ procedures for software development

Guidelines, Policies, Quality assurance

- Modeling requirements
- Policy-based infrastructure
- Testing tools
- Automated validation

Design a GUI

- Definition of Models for User, GUI Designer and developer
- Key principles for designing

Testing in LCG

- Definition of policies for testing, code versioning, other infrastructure
- Test framework based on Oval, X-Unit family, QmTest
- QA based on testing, metrics, policies compliance

Infrastructure/ procedures for software development

Infrastructure for developing/building/maintain software libraries

- Source organization
- Autotools
- Code repository, Versioning
- Usage patterns
- Code optimization
- Tools

CLHEP

- Improved source organization
- Use autotools

LCG libraries

- SCRAM and CMT-based configuration/building
- Effort in providing better off-site deployment

DataGrid WorkLoad package

- Uses autotools, CVS, rpm's

IgProf

- Non-intrusive tool for profiling
- Memory leaks, performance

Observations

Frameworks/Interoperability

- Developments based on generic frameworks, with high level of abstraction
- Broad Introduction of generic API's to shield user from technology-related details
- C++ still main language, Java well positioned for analysis tools, Python seen as easy-to-use language for tools and user applications
- Some extensions of the used languages implemented or explored
- Interoperability among languages extended.

Observations

Analysis tools/Data processing frameworks

- Attempts to standardize analysis tools
- Root technology largely dominant, but often used through a framework to drive the specific experiment use cases
- Offline frameworks mature for most of the functionality required
- Consolidation in a few areas desirable
- Distributed analysis still area under big evolution

Observations

Persistence

- Common persistency solution adopted by three LHC experiments and validated in production activities
- Root technology for Event data
- RDBMS considered for metadata, condition data.
- Technologies seen as possibly changing over time.
- Attempt to unify effort still possible for condition database

Observations

Infrastructure and general

- Infrastructure for large, geographically distributed developments (LCG, DataGrid) validated by the success of the projects
- More consolidations than large-scale developments
- Strategies based on integration, focus on flexibility and reliability
- Atmosphere quiet, few questions and discussions...