The ALICE Off-line Framework

a successful migration to OO

Federico Carminati

CERN - ALICE

CHEP 2000

February 7, 2000
ALICE beg‘98

Small(er) collaboration
- Cannot afford several development lines
- Must reuse code

Software and Physics one single group
- Use programs available and working
- Minimise the legacy−code−to−be

IT/LCB/R&D official line not ready
- Working framework based on FORTRAN + ZEBRA +PAW

Users were not (yet all) speaking OO
- Split into “dinosaurs”, “prophets” and “lost users”
The LHC Computing Challenge

Design the Detectors and Prepare the Code Now

and

Design the New Software Framework

and

Train and Involve the User Community
Are we in trouble with Simulation?

Simulation is vital, but

😊 Using GEANT 3.21
   - Stay with FORTRAN, shaky physics and geometry

😊 Using GEANT 4
   - Fresh from the oven: can we trust it yet?

😊 Using FLUKA
   - Difficult to use, limited geometry

Yes, we are in trouble!
ALICE’s choice

- **Migrate immediately to C++**
  - Immediately abandon PAW
  - But accept GEANT3.21 (initially)

- **Adopt the ROOT framework**
  - Not worried of being dependent on ROOT
  - Much more worried being dependent on G4, Objy....

- **Allow use of FORTRAN and C++**
  - Allow to start with *wrapping* and *bad design*

- **Impose a single framework**
  - Provide central support, documentation and distribution
  - Train users in the framework
The AliRoot System

- Run Control
- Generators
- New Transport Engine
- Root hits structures
- Geometry Database

Root output file

7 February 2000
CHEP 2000
The Virtual MC

AliRun

Detector Code

AliMC

TGeant3

TGeant4

TFluka

G3toG4

G3

G4 geometry

G3 geometry
Comparison new/old

- **Time per event**
  - **Old** link: >40” run: 710'/event (=11h:50m)
  - **New** link: <1” run: 435'/event (=7h:15m)

- **Space**
  - **Old** galint executable 11MB
  - **New** AliRoot executable 438kB

- **Documentation**
  - **Old** Web page (by hand!)
  - **New** in the code and automatically extracted
CERN wants to support FLUKA

- A minority (man–power) and independent line
- Possible collaboration with GEANT4 in areas of complementarity?
- Source release in some time from now

G3, G4 & FLUKA: the same input/output

- Only way to compare results and models
- G3toG4 (ALICE) and FLUGG (ATLAS/INFN-Milano)
- Access to FLUKA via ALICE Virtual MonteCarlo
- Output via the same stepping routines
Tracking schema

- FLUKA Step
- GUSTEP
- Geant4 StepManager
- AliRun::StepManager
- Module Version StepManager
- Add the hit
- Disk I/O Root
ALICE’s policy

- We follow a *minimum resistance path*
  - Use ROOT containers
  - Forbid the use of templates and STL (for the moment?)
  - Completely rely on CINT (I/O, script interpreter)
  - Do not use CLHEP, LHC++ or *alternative* components

- **Not** worried by all–out modularity

- **Official request to CERN management**
  - Support for ROOT
  - Consider it for the choice of a common LHC solution
  - Consider it in the development of new solutions
The AliRoot architecture

AliRoot base classes

Run Manager

Global Reconstruction

TPC
  Hits Pars

MUO
  Hits Pars

ITS
  Hits Pars

Detector modules

Control

Global algorithms

data

IO

ROOT IO

Cluster manager

Data

Raw data manager

Data

Track manager

Data

Global data managers
blackboars
The AliRoot framework

- Dependency on CERNLIB minimised
  - We need only GEANT 3.21 (ZEBRA–MQ)
  - All libraries are shared
- Detector versions from virtual detectors
  - AliDetector→AliTPC→AliTPCv0
  - The right version will be in the output file
- Reconstruction/Analysis are naturally developing in the same framework
  - Supported on Linux, HP–UX, Solaris, Compaq
AliRoot Modules

- Modularity of AliRoot reflected in the physical structure of the shared libraries

Complete framework: 6 months x 2 people
Code development

Central cvs database divided in modules
- Hand–added access control by module
- No configuration tools for the moment

Custom coding convention checking tool
- RuleChecker developed with IRST Trento
- Cjj project (see talk from P. Tonella et al.)
Cathedrals and Bazaars

- Definitely a Bazaar model
  - Rapid prototyping, rapid response to user feedback
- Use extensively prototyping
  - Existing technology
  - Production code
- Design only what you can prototype
  - Do not hope in technological miracle
- Development based on
  - Micro cycles
  - Macro-cycles
The ALICE Mass Storage Project

- Close collaboration with CERN/IT
  - Assess our computing model in realistic conditions
  - Develop integration DAQ and Off-line and event model
  - Provide a framework to assess new technologies
- MSS situation confused
  - Enstore, CASTOR, HPSS IBM, Eurostore I
- Participation into Eurostore II
  - EU financing, proposal submitted to Jan 15 call
  - CERN, DESY, QSW, Tera
- We will interface ROOT with Eurostore II
ALICE DC II

NA 57 data source
9 PowerPC AIX

Intel/PC Linux + PowerPC /AIX +Sun

ALICE DAQ data source

DATE=GDC+LDC
7 February 2000

Computer Centre

Intel/Linux PC Cluster 10/15 nodes

GDC Event Builder

ROOT Objectifier

pipe

10MB/s GB eth

HPSS
CASTOR
Eurostore II

5 MB/s

Switch

Switch

Switch

Switch

10 MB/s
Future milestones

- LHC main choices to be made by 2001
- Computing review started by H. Hoffman
  - Launched in Marseille to be concluded in < year
- We hope that
  - ROOT’s position will be recognised at CERN too
  - Proper support provided
- Virtual ALICE run from Jun 00 to Jun 01
  - We are trying to meet the deadline!
  - *Real life test of our framework*
- WA data challenges (with Monarc)
Lesson (re)Learnt

- Users are ready to move to OO if there is
  - A clear framework and a path to it
  - Support and consultancy for new users
  - Training targeting the framework
  - Some flexibility toward FORTRAN inclusion

- People are afraid to be unproductive for a long time or to waste their work
  - Many old FORTRAN users provide their contributions
  - Remote users contribute once a framework is decided

- A mutual trust relationship has to be built
Conclusions

😢 C++ is a VERY complex language
  - Nobody understands it all (Stroustrup)
  - The code produced can be very obscure

😃 FORTRAN lasted more than 20 y in HEP
  - It could bring us through LHC

=Migrating to a new environment makes sense only if it is better AND simpler
  - ROOT seems the right tool to achieve this goal

http://alisoft.cern.ch/offline