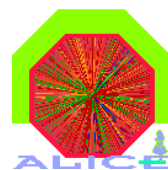# G4UIRoot

Isidro González

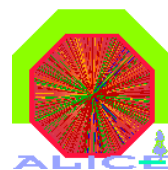ALICE

ROOT 2002

15/10/2002

Outline

- Motivations
- What are we talking about
- Features
- Known problems
- To be done
- Conclusions
- Acknowledgments
- Interesting addresses

Motivations

- Geant4 has its own command terminal user interface

```
Idle> /gun/particle pi-
Idle> /gun/energy 10 GeV
Idle> /run/beamOn 100
```

- It is difficult to know all the Geant4 commands
  - The availability changes for different Geant4 status
  - The terminal help is tedious
  - A GUI to explore the tree would help a lot

- There are several graphical user interfaces in G4 but:
  - Xm (Motif) is pretty simple
  - OPACS was difficult to integrate in a program
    - Classes with ROOT
  - GAG (Java) was slow (inter-process communication)
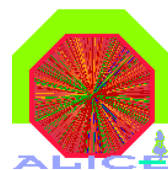
Motivations

- ROOT users (usually) like ROOT:
 - They would like something like
    ```
    [root] myGun->SetParticle("pi-");
    [root] myGun->SetEnergy(10*GeV);
    [root] myRun->BeamOn(100);
    [root] //Plot my histograms, fit them, do all sort of
    things using possibly a macro
    ```
  - Unfortunately Geant4 cannot be CINTified yet ☹
    - We still need some way to access the Geant4 commands
    - Which, in fact, sometimes are very convenient
  - However the histogramming part may be used from ROOT
- I wanted to learn ROOT GUI capabilities
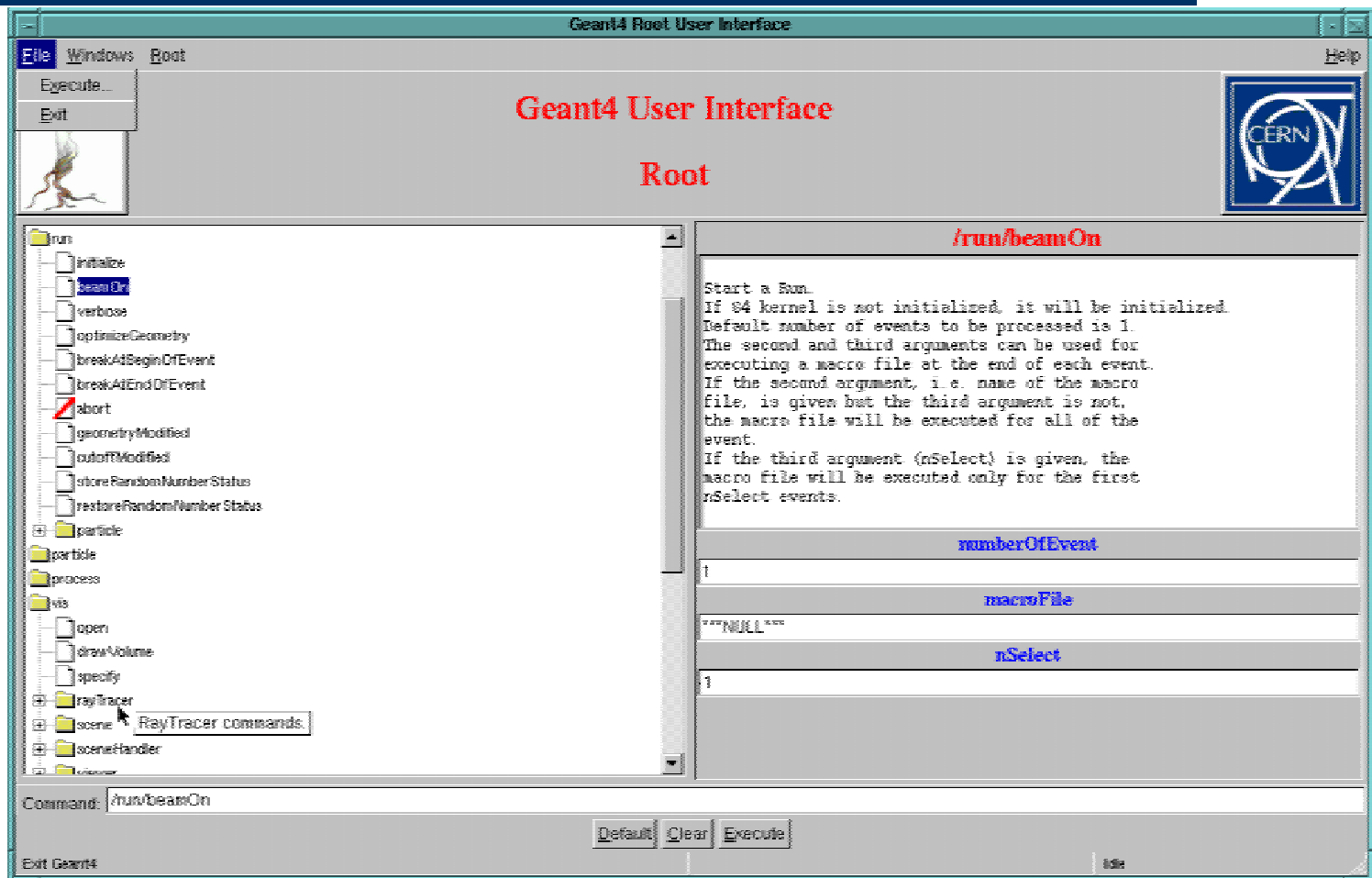- It might be a bridge between the two worlds
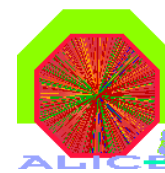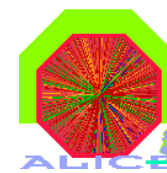
What are we talking about

- A GUI for Geant4:
  - Built with ROOT

- …providing:
  - an easy way to explore G4 command tree
  - a quick inspection of standard/error output

- A C++ Interpreter (CINT)
  - That may allow run time access to G4 classes
  - That certainly allows access to all ROOT functionallity
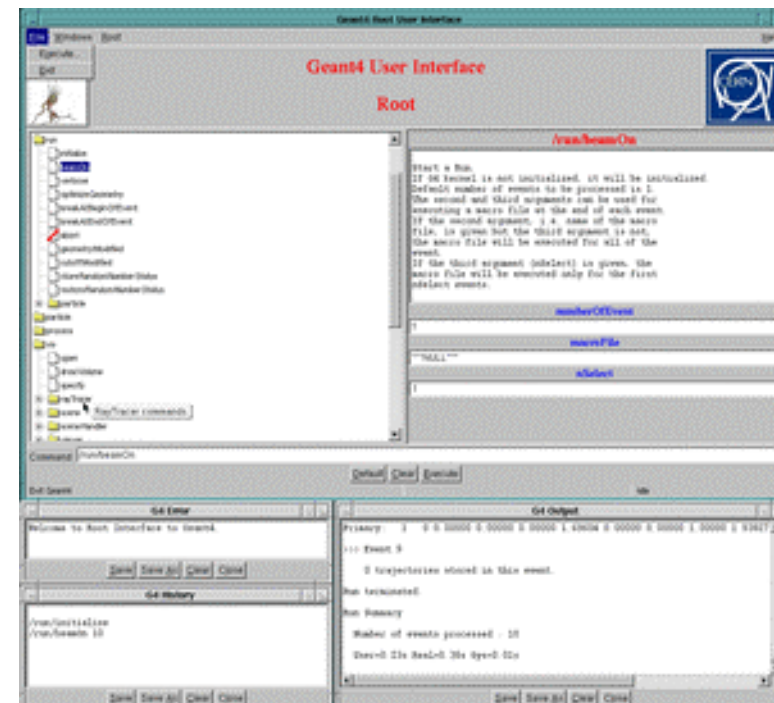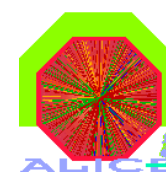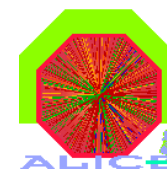
Features

Features (Command Tree)

Full Geant4 command tree displayed in a "file system" like structure

- Availability clearly marked
- Non available commands are identified and cannot be selected.
- The availability is correctly updated with Geant4 status
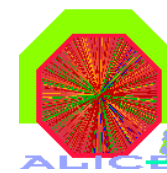
Features (Normal typing)

Normal Geant4 command typing is also possible

- Selecting a command in the tree will automatically update the command line input widget and vice-versa.
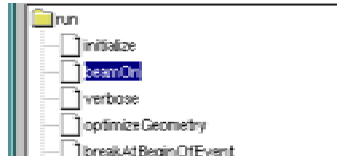
- Automatic command completion using the TAB key.
- The navigation through the successful commands executed before may be done using the arrow keys.
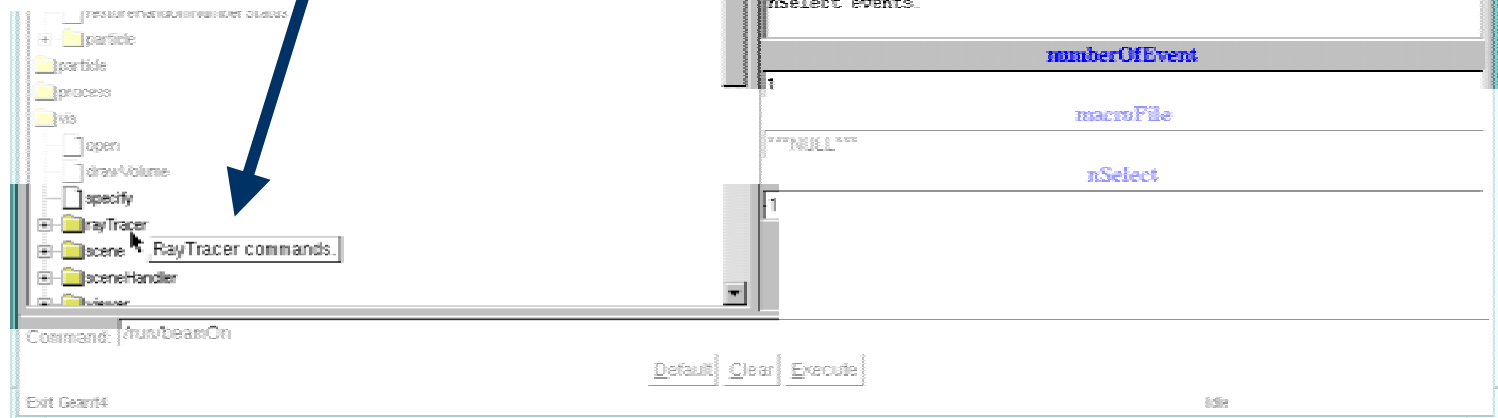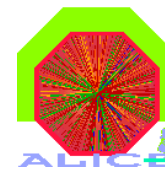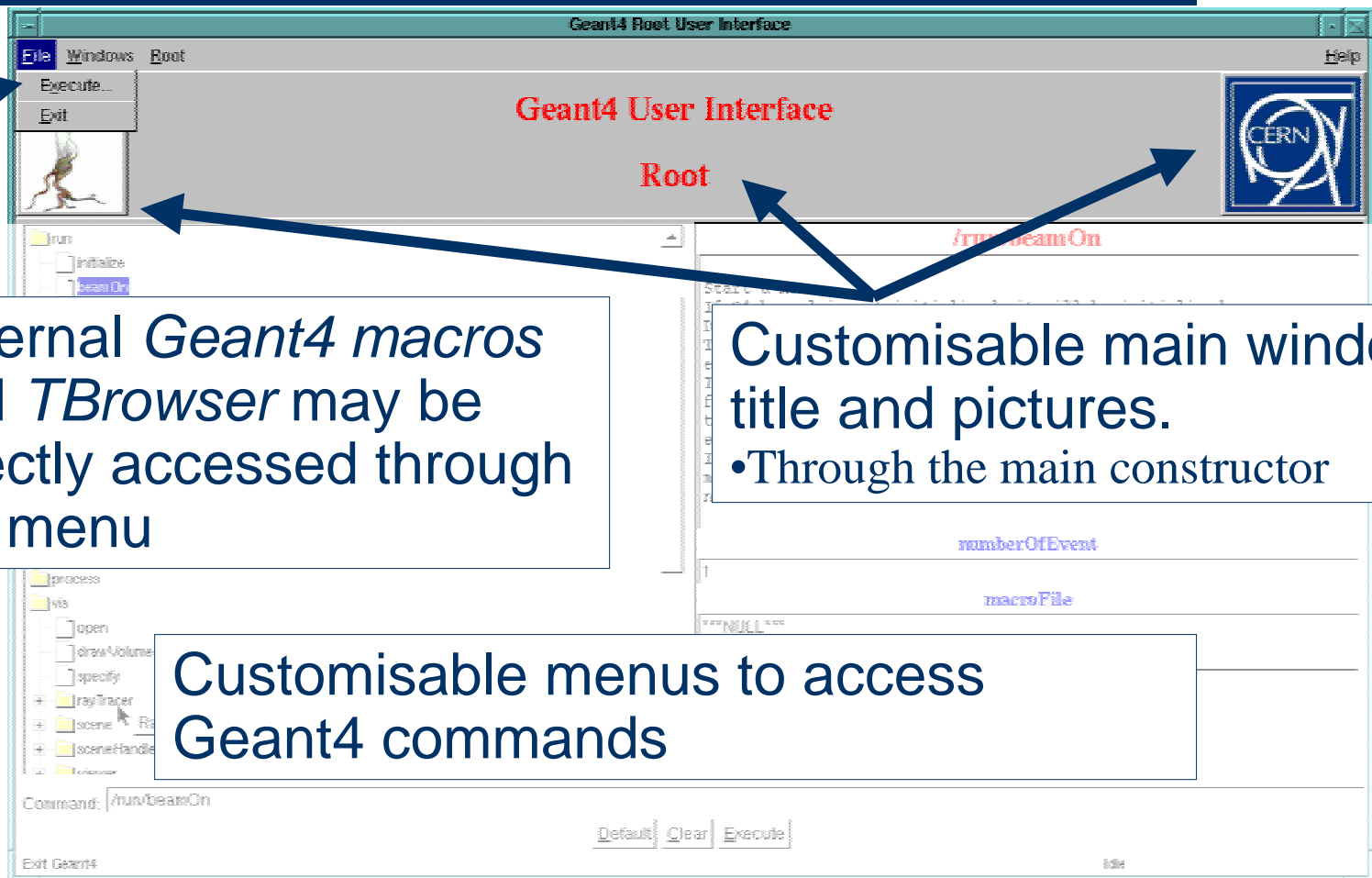
Features (Command Help)

Full and short command help

- Full command help for selected command
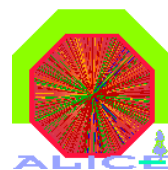- Short command help in pop up messages and in the status bar.

Features (Menus & pictures)

External *Geant4 macros* and *TBrowser* may be directly accessed through the menu

Customisable main window title and pictures.
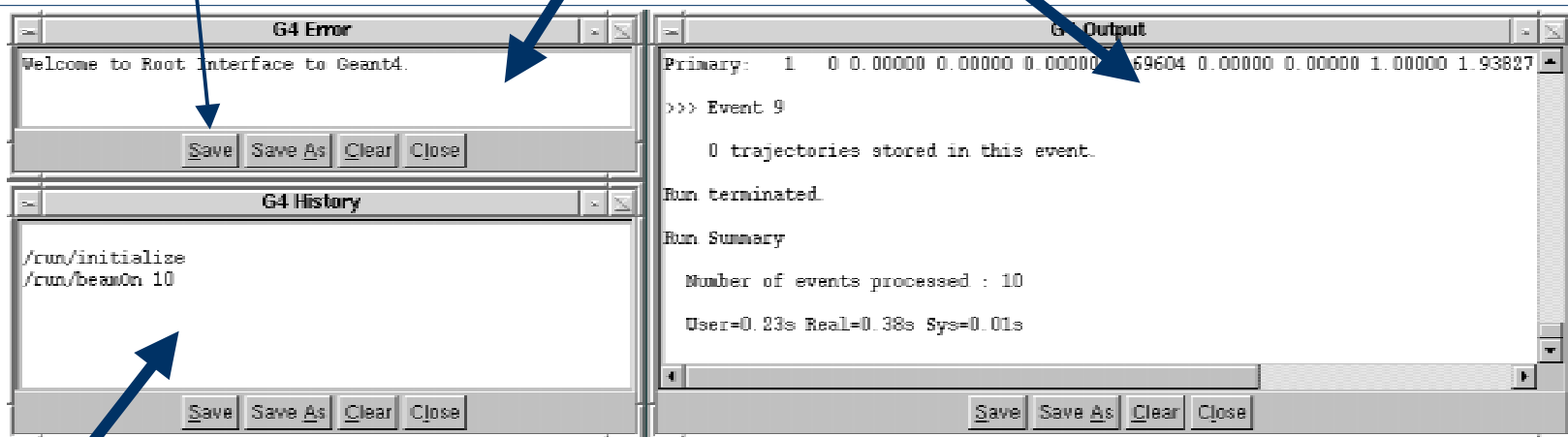•Through the main constructor

Customisable menus to access Geant4 commands
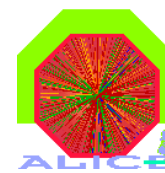
Features (Output Windows)

Different windows for error and normal output
- With saving capabilities.

History window with saving capabilities.
- History is always tracked.
- Successful commands may be recalled at any point by simply hitting the up arrow at the command line.

Features (Root Interpreter)

Root interpreter (CINT) included.

- It runs in the terminal.
- If Geant4 is CINTified it will give run-time access to the MC
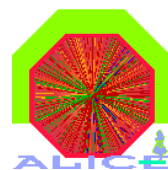- Interactive histograms, etc…

A few implementation details

- 4 classes + 1 dictionary = 5 KLOC
- ROOT and Geant4 separated (as far as it was possible) in different classes
  - Geant4: `G4UIRoot`
  - ROOT: `G4RootMainFrame`, `G4RootTitleFrame`, `G4RootCommandFrame`, `G4RootTextFrame`
- Signal/Slot mechanism used for widget event handling
- Fully integrated into the Geant4 (complex) compilation framework
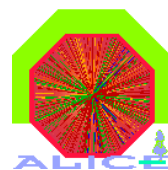
Known Problems

- Inconvenient interaction with the output windows
  - Limitation in the ROOT GUI implementation
    - Arrow keys, page up/down keys do not work
    - Reported to ROOT
  - Slower output compared to standard terminal output
    - Related to the fact that we are using graphics
    - Possible solution using timers to schedule updates
- Small problems with the layouts
- Commands with too many parameters may not fit in the window
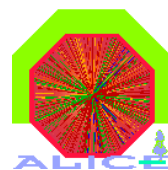
To be done

- Improve synchronisation between command line and tree selection input
- Histograms integrated in the GUI
  - Scan for newly created histogram after every command
  - Provide a basic set of tools to interact with them
  - We need a set of examples on how to use this
- Move to CVS:
  - Currently released through tar files
  - Versioning is difficult (Root/Geant4 versions)
- Improve the overall look
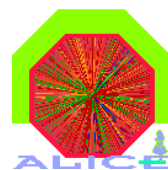- Installation script
- Suggestions, requests ?
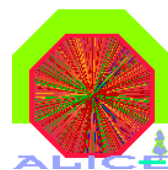
Conclusions

- A GUI was built with ROOT in a very short time
    - Main development was done in 2 months (working partially on this)
    - The additions and improvements were done in spare days
- You get all the advantages of Geant4 and ROOT
- Easy way to explore the Geant4 command tree
- Can be extended adding more complex or personal capabilities
- Seems to be people out there trying it ☺
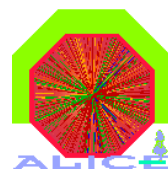
Acknowledgments

- ALICE (F. Carminati) who supported this "personal" project

- Jacek M. Holeczek for his clever ideas and help in solving some problems, as well as for providing the modified examples.

- GAG user interface whose look and feel inspired this GUI

Interesting addresses

- **ROOT**:
  - *http://root.cern.ch*

- **Geant4**:
  - *http://cern.ch/geant4/*

- **G4 & ROOT** (from Root HowTo)
  - *http://root.cern.ch/root/HowtoMC.html*

- **G4UIRoot**:
  - *http://home.cern.ch/iglez/alice/G4UIRoot*