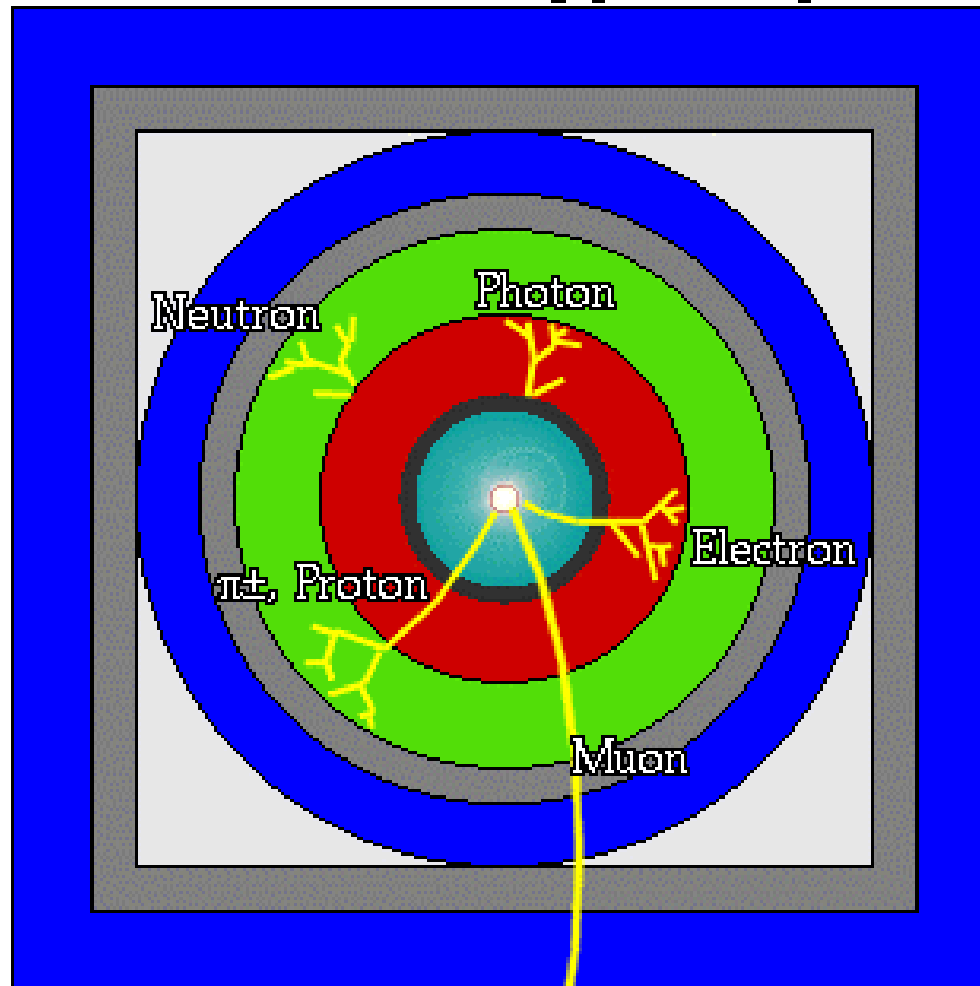


Why ROOT?

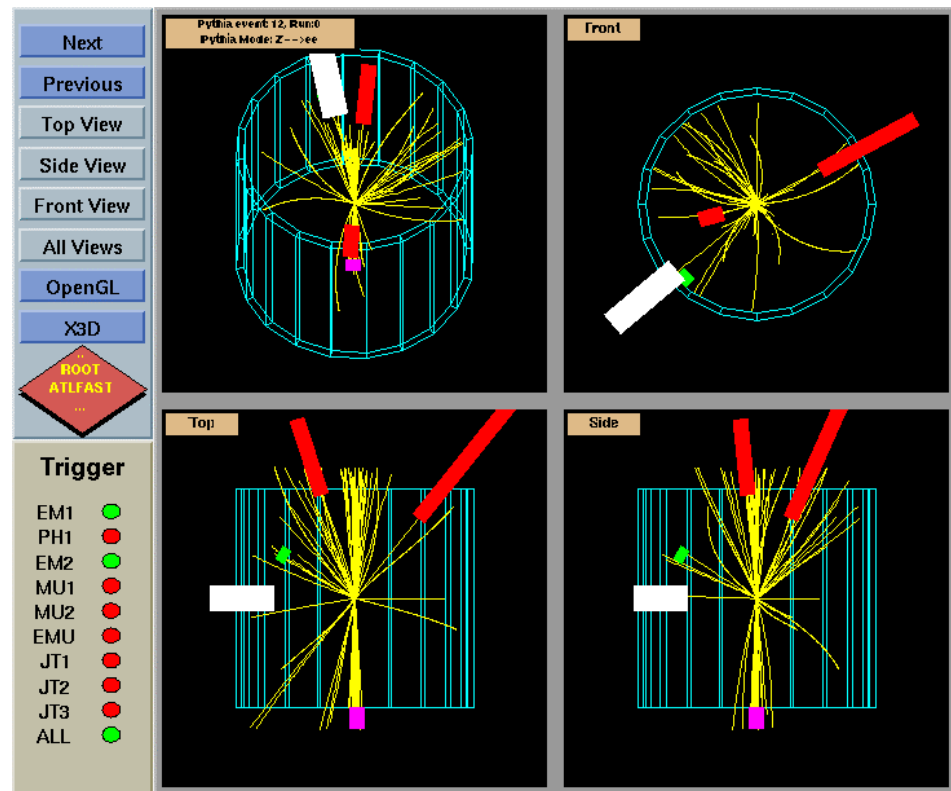
A detector cross-section, showing particle paths

-  Beam Pipe (center)
-  Tracking Chamber
-  Magnet Coil
-  E-M Calorimeter
-  Hadron Calorimeter
-  Magnetized Iron
-  Muon Chambers



ROOT

- ROOT:
- is an object_oriented frame work aimed at solving the data analysis challenges of high energy physics
- Object_oriented: by encapsulation , class hierarchies complexity is reduced
- Framework: contain the basic utilities and services
- Such I/O and 2D,3D graphics,Histograms ...



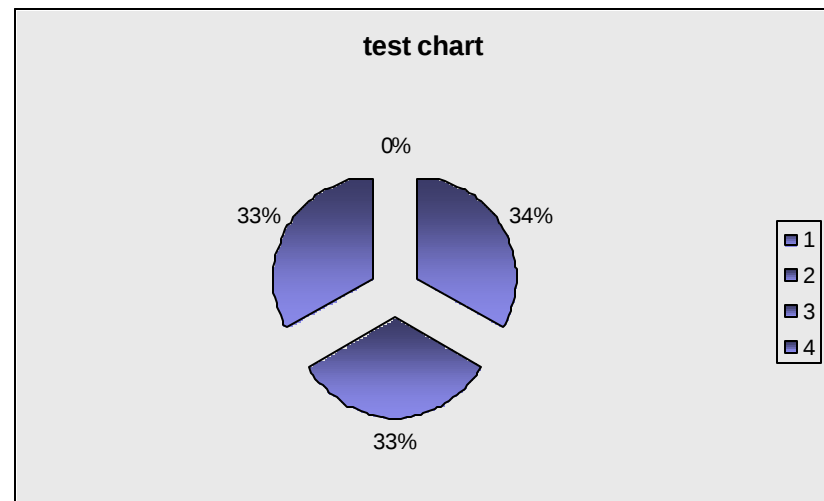
WAYS TO USE ROOT

Three ways to use root:

-GUI: graphic use interface
(windows ,buttons ,menus..)

-Macros&Programs

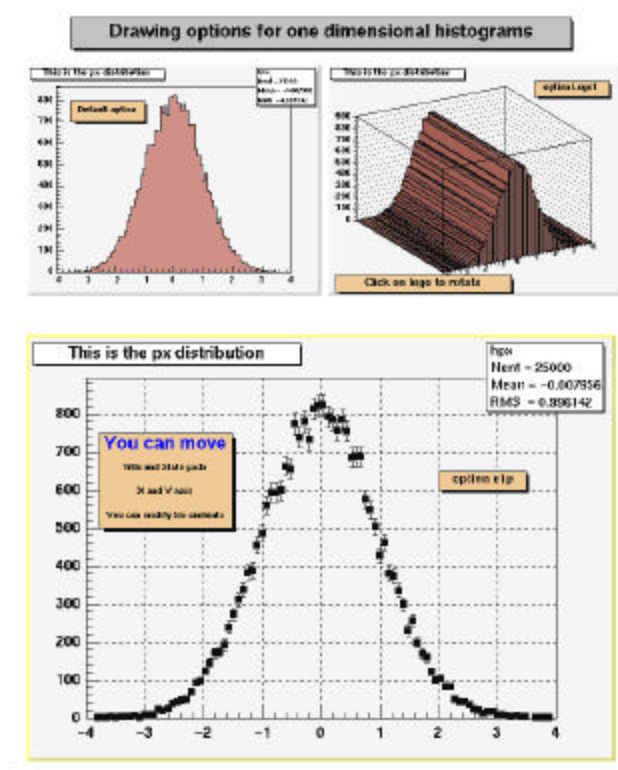
-Command line interpreter
(CINT) C++ command



HISTOGRAMS

Root support the following histograms:

- .1-D histograms
- .2-D histograms
- .3-D histograms
- .profile histograms



Creating Histograms

- Histograms are created with constructor:

```
TH1F * h1 = new TH1F("h1","h1 title" , 100, 0, 4) ;
```

```
TH2F * h2 = new TH2F("h2","h2 title", 40, 0, 4, 30, -3, 3);
```

- for 1-D histogram : the name of the histogram, the title,
the number of bins, the X min, and the X max

-for 2-D histogram: the bins along the X axis....
the bins along the Y axis....

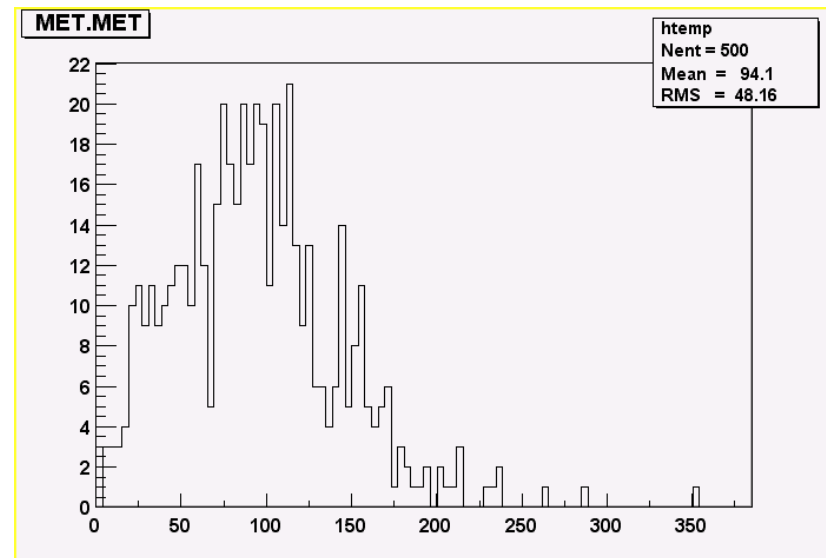
Filling Histograms

A histogram is filled typically with statement like this:

-h1 -> Fill (x) ;

-h2 -> Fill(x,y);

The Fill method computes the bin number corresponding to the given x ,y ,z



The options

1-the SCATter plot option

2-the ARRow option

3-the BOX option

4-the COLor option

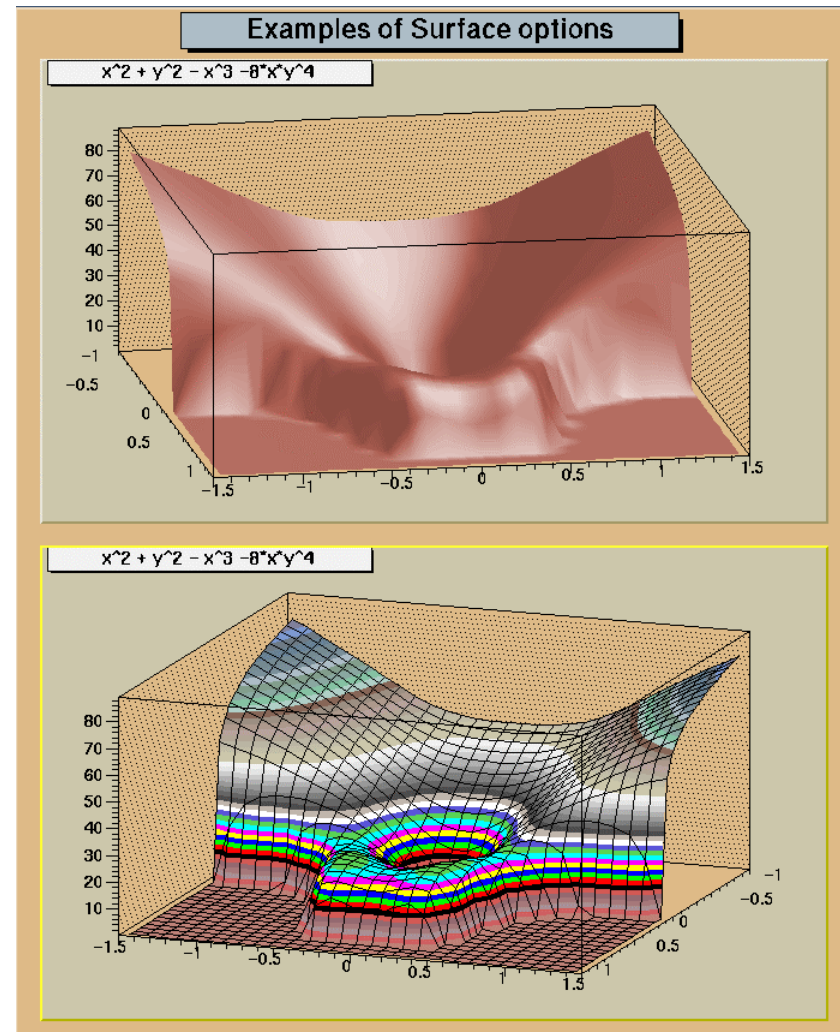
5- the LEGO option

.....

.....

...

..



Graphs

A graph is graphics object made of
two array X and Y holding

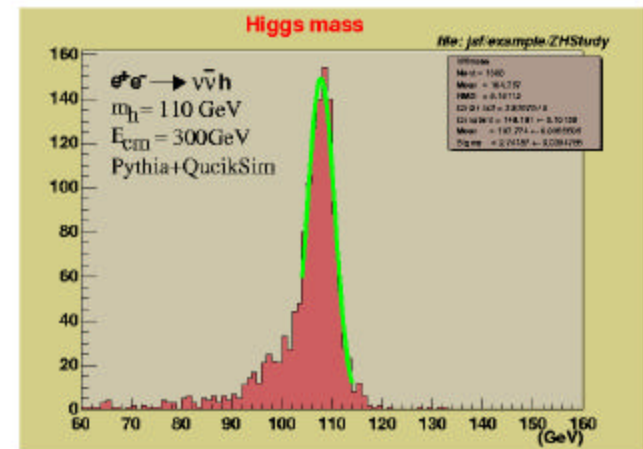
The x and y coordinates of n points

Graphs are created with a constructor

Graph Draw options:

- 1- “L” a simple poly –line
- 2- “*” a star is plotted at each point
- 3- “B” a bar chart is drawn at each
point.

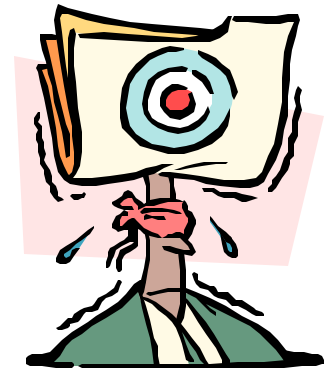
.....



Graphics and the Graphical user interface

- Graphical capabilities of root range from 2-D objects (lines ,arrows,...) to various plots, histograms and 3-D graphical objects.
- BY PROGRAMING: all the objects are supposed to be drawn.
- When an object is drawn one can interact with it :
(moving , resizing , modifying ...)

Examples feynman Diagrams:



Input/ Output

- It explain how the saving and reading of objects to and from Root files.
- A Root file is a UNIX file directory

It can contain directories and object organized in unlimited number of levels

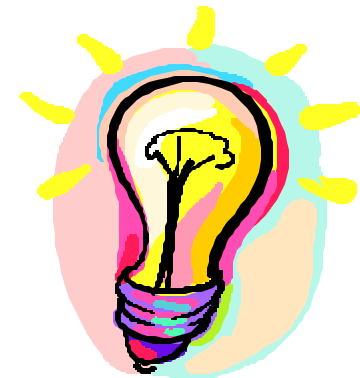
In the manual you can find :

The compression, file recovery, the logical file

The class T file and it's methods.

How to navigate in the file ,

How to save objects and read them back.



TREES

why Trees ?

1-storing large number of entries

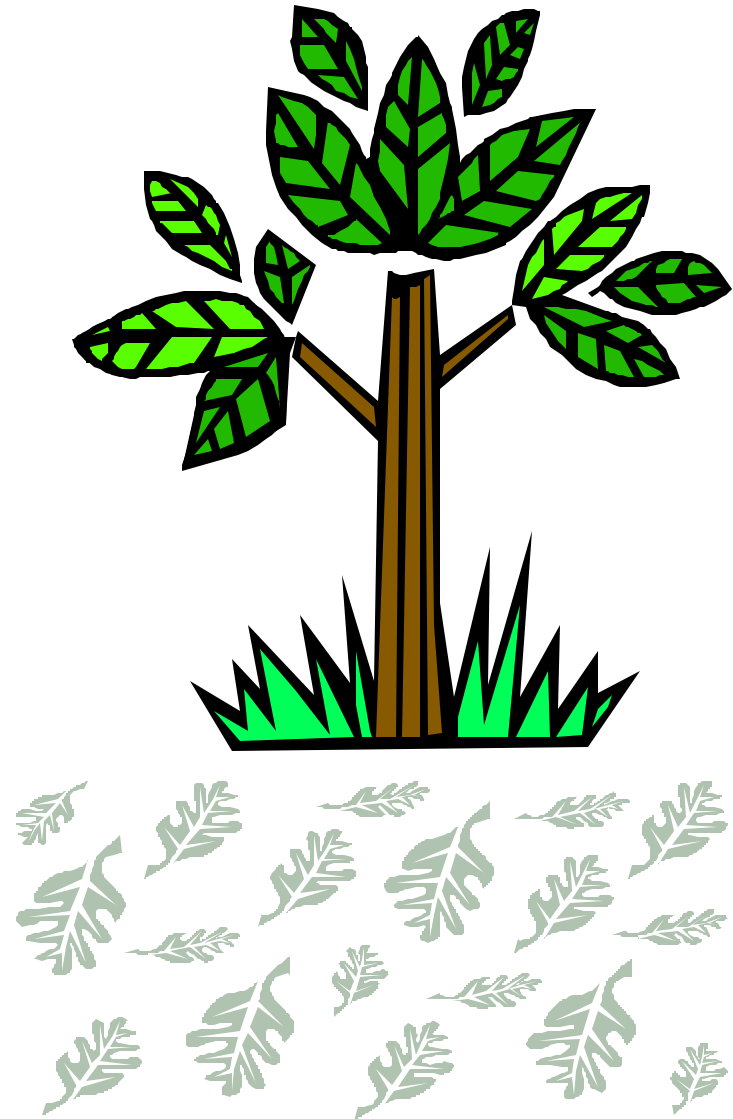
The T tree class is designed for storing large number of events
it minimize the space.

2-Hierarchy of branches and leaves
which allows a flexible organization of data.

_a tree may have one or many branches
and a branch may have one or many leaves

_the branches are independent and may
be written to different files

3-reading selective branches : each
branch can be read independently of
any other branch.



Final word about Root

- It' a powerful tool to
Analysis the current
standard data in HEP

And let's Do our best to
learn it.

