# The Athena off-line sotware

**Valerio Filippini**

Istituto Nazionale di Fisica Nucleare,
Sezione di Pavia, Italy

III Root User Meeting,
FermiLab 2001

**The Athena off-line sotware is based on:**

**root** **geant 3.21** **linux**

**starting from the TGeant3 class of the ALICE group**

**many thanks to ALICE collaboration (F. Carminati)**

**The Athena off-line uses these classes:**

**TGeant3** *(from Alice, with some corrections)*
**THIGZ** *(from Alice, with some corrections)*

**TAthenaEvent** *(event description)*
**TAthenaDisplay** *(detector geometry (root) and graphics)*
**TAthenaMC** *(detector geometry (HIGZ) and geant 3.21 functions)*

## The macro to run a Monte Carlo looks like

```
{
TAthenaEvent *gEv = new TAthenaEvent();

TAthenaDisplay *Dis = new TAthenaDisplay(&gEv);
TAthenaMC *gMc = new TAthenaMC(&gEv);

gMc→SetCUTGAM (10 E−6); // 1 Kev
gMc→SetCUTELE (10 E−6); // 1 Kev
gMc→SetMULS (2);
gMc→SetLOSS (2);
gMc→SetHADR (1);
gMc→UgInit(kFALSE);  // kTRUE = 3 T magnetic field

gMc→UgEvent();       // generate the event

gEv→RecPoints();     // reconstruct the hit
gEv→RecTrakcs();     // reconstruct the tracks
}
```

# The residual Fortran interface

```
#include "TAthenaEvent.h"
#include "TAthenaMC.h"
ClassImp(TAthenaMC)

TAthenaMC *gAthenaMC;                                    // FORTRAN interface
extern "C" void gukine_() { gAthenaMC→GuKine(); }   // FORTRAN interface
extern "C" void gustep_() { gAthenaMC→GuStep(); }   // FORTRAN interface
extern "C" void guout_()  { gAthenaMC→GuOut();  }   // FORTRAN interface

//_____________________________________________________________________
TAthenaMC::TAthenaMC(TAthenaEvent **e, Bool_t Inter) : TGeant3("AthenaMC") {
  gAthenaMC = this;   // FORTRAN interface
  //  .....
}
```

# The “UgInit”

```
void TAthenaMC::UgInit(Bool_t field) {
  Gzinit();
  DefineParticles();

  //===>  define tracking media
  Gmate();

  Int_t Nalu=0;
  Medium(Nalu,"ALU",  9, 0,Bt,Bz ,0,0,0,0,0 ,NULL,0);
  // ....

  //===> define and locate volumes
  Float_t p01[] = { 1.25, 1.55, 10};
  Gsvolu("TRAP","TUBE",Nalu, p01 ,3);              // trap
  Gspos ("TRAP", 1, "CHAN", 0, 0, 0, 0);
  //....

  FinishGeometry();
  BuildPhysics();
}
```

# The "GuKine"

```
void TAthenaMC::GuKine() {
  // ===> clear the previuos event
  fEvent→DeleteEvent();

  // ===> define a vertex
  Int_t nv = Gsvert(v,0,0,0,0);

  // ===> pbar − p annihilation
  TGenPhaseSpace ps;
  ps.SetDecay(P,NumPar[Nfs],ParMas[Nfs]);
  for (Int_t n=0; n<NumPar[Nfs]; n++)
    Gskine(p ,j ,nv ,NULL ,0);

  // ===> e+ − e− annihilation
  Float_t p0[3] = { PGa*Ste*Cph, PGa*Ste*Sph, PGa*Cte };
  Gskine(p0 ,1 ,nv ,NULL ,0);

  Float_t p1[3] = {−p0[0], −p0[1], −p0[2] };
  Gskine(p1, 1 ,nv ,NULL ,0);
}
```

# The "GuStep"

```
void TAthenaMC::GuStep() {
  // =====> save the points for display
  TLorentzVector x;
  TrackPosition(x);
  TVector3 *v = new TVector3 (x.Vect());
  v→SetUniqueID(TrackPid());
  v→SetBit(1,IsNewTrack());
  fEvent→AddMCPoint(v);

  // =====> hit on silicon
  if (strcmp(CurrentVolName(),"SPDD")≡0) {
    //...
  }

  // =====> hit on cristals
  if (strcmp(CurrentVolName(),"CSIX")≡0) {
    // ...
  }

}
```

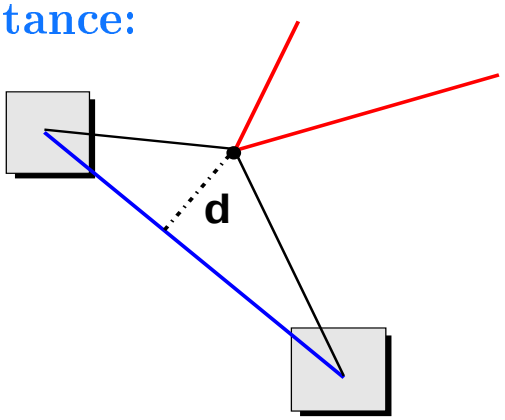
# The event reconstruction steps

-reconstruct the points on silicon

-find the 3D vertex

-identify the two crystals hit by a 511 Kev gamma

-compute the 3D distance:




-search a peak around zero

## What's rigth with root:

- **Geant 3.21 interface works fine,**
- **compact code,**
- **very fast writing,**

## What's wrong with root:

- **Two geometries** *(root and HIGZ)*