GlueX Monte Carlo

Benedikt Zihlmann

Geant steps

Track primary particles in STEPS:

GUSTEP.F

- GEANT generates secondary particles at each step (physics processes)
- What to do with secondaries? track them? give them their own track number?

```
      if (nosecondaries.eq.0) then
        do i=1,NGKINE
          itypa  = GKIN(5,i)
          if (itypa.ne.4) then

c     make primary except if in calorimeter volume and not hadronic interaction
            iflgk(i) = 1
            cint = KCASE
c            print *, cchar
            rx = sqrt(VERT(1)**2+VERT(2)**2)
            if ( (((rx>65.) .and.((VERT(3)>-17.) .and.
     >           (VERT(3)<390.))).or.(VERT(3)>625.)).and.
     >           (cchar.ne.'HADR')) then
              iflgk(i) = 0
            endif
            call GSKING(i)
          endif
        enddo
      endif
```

When to record track data?

At each step in GUSTEP.F

- Check if if track leaves current volume.
- If yes check if that volume was a detector
- If yes call savehits.F
  - Decide weather to record the track data
  - If yes call appropriate function hitXXX.c
  - XXX = FDC, CDC, FCAL, BCAL, TOF, Start, Tag
  - Note: from this point on C-code!

Jefferson Lab

Register hits

Example: hitFDC.c

- C-function to store data:

```
/* register hits during tracking (from gustep) */

void hitForwardDC (float xin[4], float xout[4],
                   float pin[5], float pout[5], float dEsum,
                   int track, int stack, int history, int ipart)
{ ....
  if (history == 0) // REGISTER TRUTH INFO IF NOT IN CALO
  { ...
    if (*twig == 0) // LIKE TRACK NUMBER, PARTICLE_ID ETC.
    { ...
      s_FdcTruthPoints_t* points = make_s_FdcTruthPoints(1);
      ....
    }
  }
  ....
  /* FIRST RECORD ANODE WIRE HIT */
  if (dE > 0 && valid_hit)
  { ....
  ahits = fdc->fdcChambers->in[0].fdcAnodeWires->in[0].fdcAnodeTruthHits;
  ...
  /* keep the earlier hit and discard the later one */
  /* Feb. 11, 2008 D. L. */
   ahits->in[nhit].t       = tdrift;
   ahits->in[nhit].d       = dradius;    //ADDED SIMON
   ahits->in[nhit].dE      = dEsum;
   ahits->in[nhit].itrack  = track;      //ADDED BENI
   ahits->in[nhit].ptype   = ipart;      //ADDED BENI
  ...
```

Jefferson Lab

MC data model

MC data structures are defined in sim-recon/src/libraries/HDDM/event.xml

```
...
<centralDC minOccurs="0">
    <cdcStraw maxOccurs="unbounded" minOccurs="0" ring="int" straw="int">
        <cdcStrawHit dE="float" maxOccurs="unbounded" t="float" itrack="int" ptype="int"/>
        <cdcStrawTruthHit dE="float" maxOccurs="unbounded" t="float" d="float" itrack="int" p
    </cdcStraw>
    <cdcTruthPoint dEdx="float" dradius="float" maxOccurs="unbounded" minOccurs="0" phi="floa
                   primary="boolean" ptype="int" px="float" py="float" pz="float" r="float" t
                   track="int" z="float"/>
</centralDC>

<forwardDC minOccurs="0">
    <fdcChamber layer="int" maxOccurs="unbounded" module="int">
        <fdcAnodeWire maxOccurs="unbounded" minOccurs="0" wire="int">
            <fdcAnodeHit dE="float" maxOccurs="unbounded" t="float" itrack="int" ptype="int"/>
            <fdcAnodeTruthHit dE="float" maxOccurs="unbounded" t="float" d="float" itrack="in
        </fdcAnodeWire>
        <fdcCathodeStrip maxOccurs="unbounded" minOccurs="0" plane="int" strip="int">
            <fdcCathodeHit maxOccurs="unbounded" q="float" t="float" itrack="int" ptype="int"/
            <fdcCathodeTruthHit maxOccurs="unbounded" q="float" t="float" itrack="int" ptype=
        </fdcCathodeStrip>
        <fdcTruthPoint E="float" dEdx="float" dradius="float" maxOccurs="unbounded" minOccurs
                       primary="boolean" ptype="int" px="float" py="float" pz="floa
                       track="int" x="float" y="float" z="float"/>
    </fdcChamber>
</forwardDC>
```

Jefferson Lab

DANA container classes

Create first the container classes to hold the MC data in the DANA world.

- Container classes holding hit data are defined in sim-recon/src/libaries/XXXX
- Example FDC: sim-recon/src/libraries/FDC/DFDCHit.h
  factory registered in FDC_init.cc twice (once with tag TRUTH)
- Note: All factories are registered in XXX_init.cc

```
...
//DEFINE FACTORY FROM CONTAINER CLASS
#include "DFDCHit.h"
typedef JFactory<DFDCHit> DFDCHit_factory;
...
// REGISTER FACTORY TWICE WITH DIFFERENT TAG
        loop->AddFactory(new DFDCHit_factory());
        loop->AddFactory(new DFDCHit_factory("TRUTH"));
....
```

- TruthPoints of all detectors are in sim-recon/src/libraries/TRACKING/DMCTrackHit.h
  → vector<DMCTrackHit> is a list of all TruthPoins in all detectors

Jefferson Lab

The Translation

Translation hddm to DANA done in sim-recon/src/libraries/HDDM/DEventSourceHDDM.cc

```
...
 if(dataClassName =="DCDCHit" && (tag=="" || tag=="TRUTH") )
   return Extract_DCDCHit(my_hddm_s, dynamic_cast<JFactory<DCDCHit>*>(factory) , tag );

 if(dataClassName =="DFDCHit" && (tag=="" || tag=="TRUTH") )
   return Extract_DFDCHit(my_hddm_s, dynamic_cast<JFactory<DFDCHit>*>(factory), tag);
...
    if (tag=="") {
     for(unsigned int j=0; j<cdcstraw->cdcStrawHits->mult; j++){
       s_CdcStrawHit_t *strawhit = &cdcstraw->cdcStrawHits->in[j];
       DCDCHit *hit = new DCDCHit;
       hit->ring     = cdcstraw->ring;
       hit->straw    = cdcstraw->straw;
    ...
     if (tag=="TRUTH") {
      for(unsigned int j=0; j<cdcstraw->cdcStrawTruthHits->mult; j++){
        s_CdcStrawTruthHit_t *strawhit = &cdcstraw->cdcStrawTruthHits->in[j];
        DCDCHit *hit = new DCDCHit;
        hit->ring     = cdcstraw->ring;
        hit->straw    = cdcstraw->straw;
     ...
```

Untagged structures contain smeared data from mcsmear!

Purpose

Get track ID for each hit in the FDC and CDC!

Identify hits with tracks and vice versa.

Example: hd_dump -DDCDCHit:TRUTH hdgeant_pion_nsmeared.hddm

```
...
DCDCHit:TRUTH
```

ring:	straw:	dE:	t:	d:	itrack:	ptype:
1	14	9.0125e−06	3.2	1.2353e−02	1	8
1	15	2.1986e−05	125.9	6.7444e−01	6	14
1	16	3.7269e−04	28.5	1.3114e−01	2	46
2	15	5.8314e−05	88.4	4.6991e−01	6	14
2	15	2.4093e−04	54.5	2.7847e−01	2	46
3	18	6.7389e−06	115.1	6.2608e−01	1	8
3	19	4.7112e−05	93.5	5.0002e−01	6	14
3	19	1.9679e−04	13.4	5.6562e−02	2	46
4	17	8.7143e−06	66.5	3.5819e−01	1	8
4	18	1.7872e−04	12.3	5.3515e−02	2	46
5	20	3.5586e−06	121.3	6.5791e−01	5	9
5	21	6.9230e−05	59.5	3.1863e−01	1	8
5	21	3.4998e−05	3.3	8.1471e−03	6	14
5	23	2.4548e−06	1245.6	4.9736e−01	69	2
5	25	1.1171e−03	618.4	1.4239e−01	68	5
5	25	6.4023e−06	1155.5	9.2943e−03	69	2
5	26	2.2641e−06	1237.7	4.5589e−01	69	2
6	15	2.5197e−06	107.9	5.8261e−01	5	9
6	16	5.6681e−06	18.4	9.0512e−02	5	9

```
.....
```