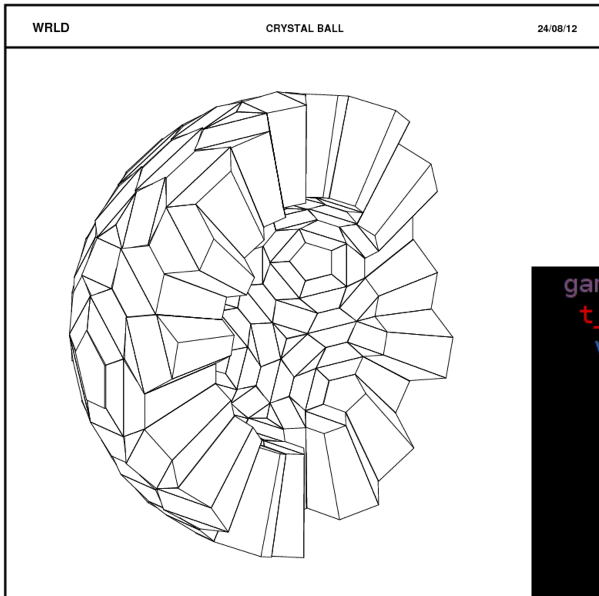
ggland examples




```
gamma:    [1] pxyz=(   -8.537,    -1.231,    -5.059) T=    10.000
 t_delay=    0.000 pdg=22     te=    0.032 se=     0.968 # pid=1
  vertex: t=    0.032 xyz=(    -0.827,    -0.119,    -0.490) s=     0.968
          PAIR     pxyz=(    -8.537,    -1.231,    -5.059) STOP T=    10.000 {
    e+:        [2] pxyz=(    -2.856,    -0.591,    -1.332) T=     2.736
     t_delay=    0.000 pdg=-11     te=    0.054 se=     0.604 # pid=2
      vertex: t=    0.042 xyz=(    -0.973,    -0.200,    -0.582) s=     0.272
              BREM     pxyz=(     0.842,    -1.470,     0.158) T=     1.266 {
        gamma:    [3] pxyz=(     0.273,    -0.253,     0.185) T=     0.416
         t_delay=    0.000 pdg=22     te= 101.588 se=    escape # pid=1
      }
      vertex: t=    0.054 xyz=(    -0.949,    -0.231,    -0.561) s=     0.604
              ANNI     pxyz=(     0.000,    -0.000,    -0.000) STOP T=     0.000 {
        gamma:    [4] pxyz=(    -0.065,    -0.125,     0.491) T=     0.511
         t_delay=    0.000 pdg=22     te=  69.472 se=    escape # pid=1
        gamma:    [5] pxyz=(     0.065,     0.125,    -0.491) T=     0.511
```

Håkan T. Johansson, Chalmers, Göteborg

January 2013

Getting started (@ chalmers/sub-linux)

Download & compile:

```
BASE=/net/data1/htj/ggland/

cd $BASE
cvs -d :ext:land@lx-pool.gsi.de:/u/johansso/CVS co land02
cd land02/scripts/ggland
make land_geant4
make -C ../../util/hbook root_writer
```

Running:

```
. geant4.sh         # Setup environment (planck-o)
/net/data1/htj/ggland/land02/scripts/ggland/land_geant4

GG=/net/data1/htj/ggland/land02/scripts/ggland/
```

Example 1 – e^- in a box

```
GG=/net/data1/htj/ggland/land02/scripts/ggland/
```

```
$GG/land_geant4 --test=d=1cm,z0=2cm,type=Ge,out_col=0 \
  --gun=T=1MeV,e- | \
  grep TSTDMP | sed -e "s/#.*//" -e "s/.*://"
```

Output formatting

Particle gun

Particle: e^-
Kinetic energy: 1 MeV

(default direction: z)

Test volume

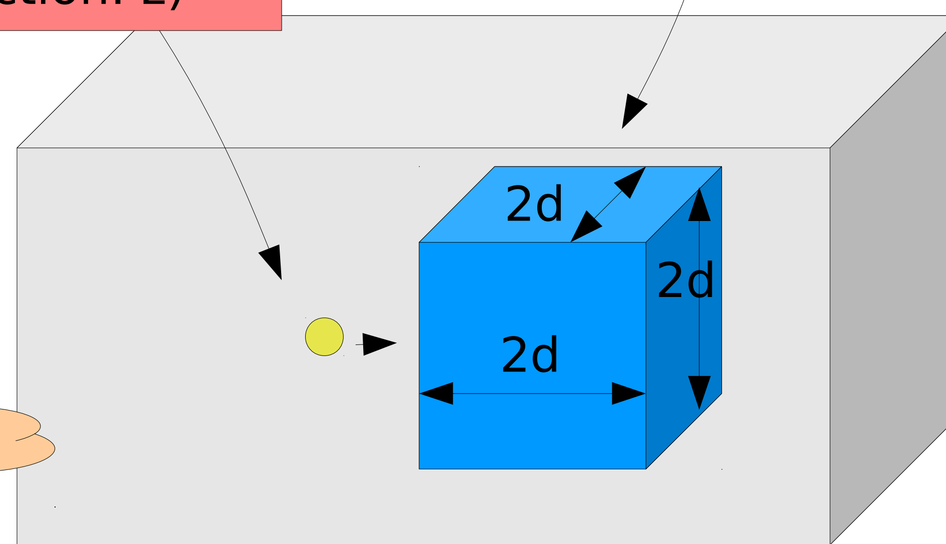
Half-side: d = 1 cm
Material: Ge
Center at: z = 2cm

Output:

```
0.99881
0.99906
0.49163
0.99759
0.99386
0.37398
0.99842
0.99763
0.29450
0.99929
```

Units: MeV

Why not 1.00?



Example 2 – world vacuum

```
$GG/land_geant4 --test=d=1cm,z0=2cm,type=Ge,out_col=0 \
  --gun=T=1MeV,e- \
  --world=type=vacuum | \
  grep TSTDMP | sed -e "s/#.*//" -e "s/.*://"
```

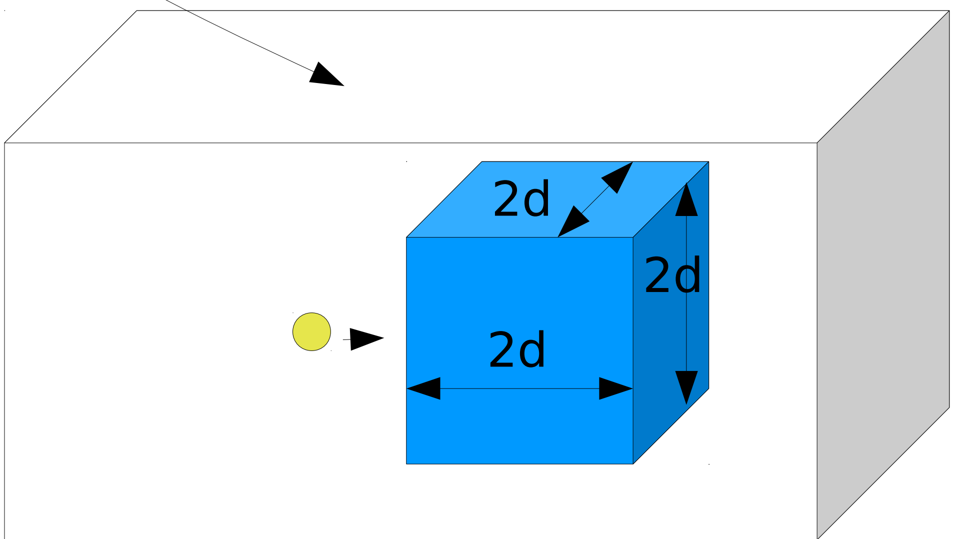
World volume

Material: vacuum

(default size: d = 20 m)

Output:

```
1.00000
0.72242
1.00000
1.00000
0.42274
1.00000
1.00000
1.00000
1.00000
1.00000
```



Example 3 – root tree

```
$GG/land_geant4 --test=d=1cm,z0=2cm,type=Ge,tree=1 \
  --gun=T=1MeV,gamma --world=type=vacuum \
  --events=10000 --tree=example3.root
```

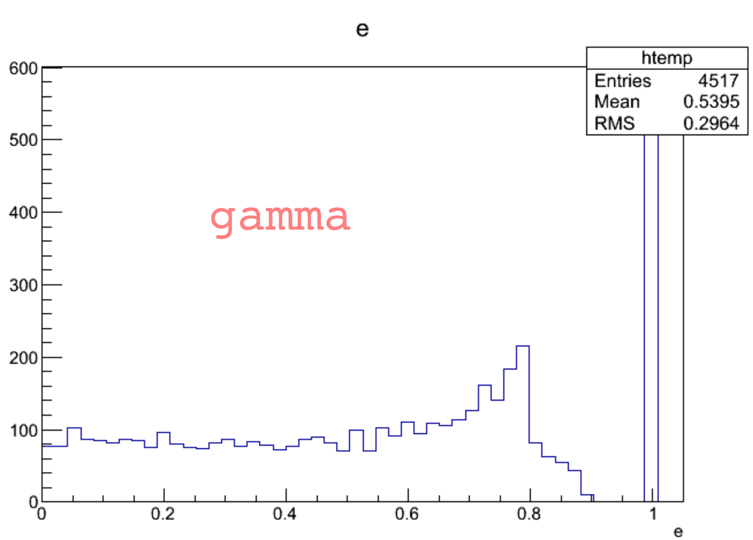
More events!

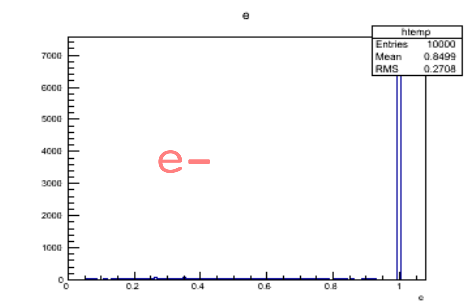
Output root tree

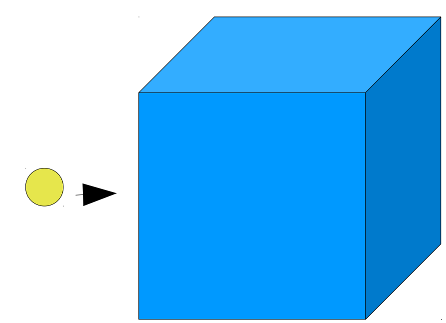
Filename: example3.root

```
root -l example3.root
```

```
_file0->ls()
h101->Print()
h101->Draw("e")
```

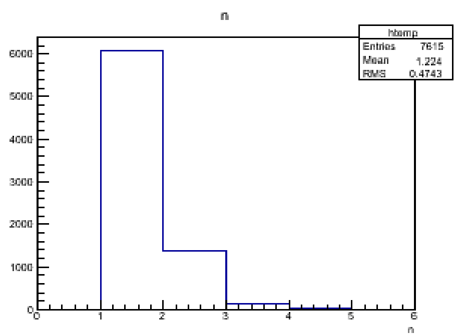






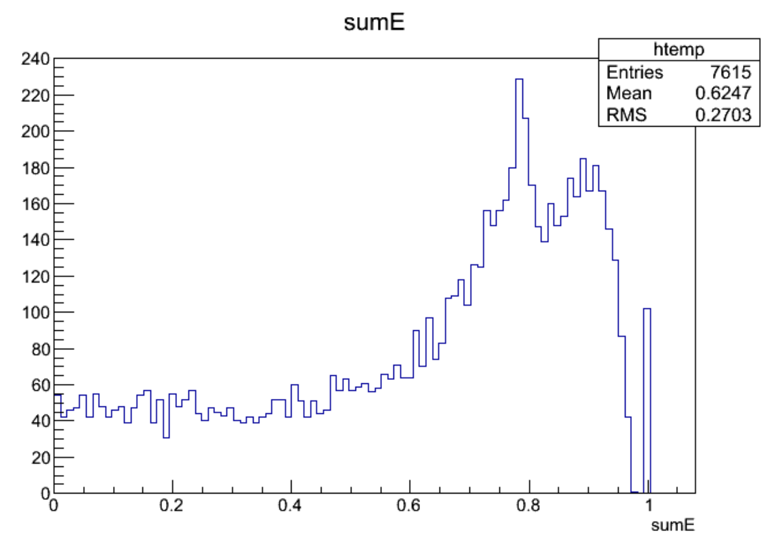
Example 4 – zero-suppression

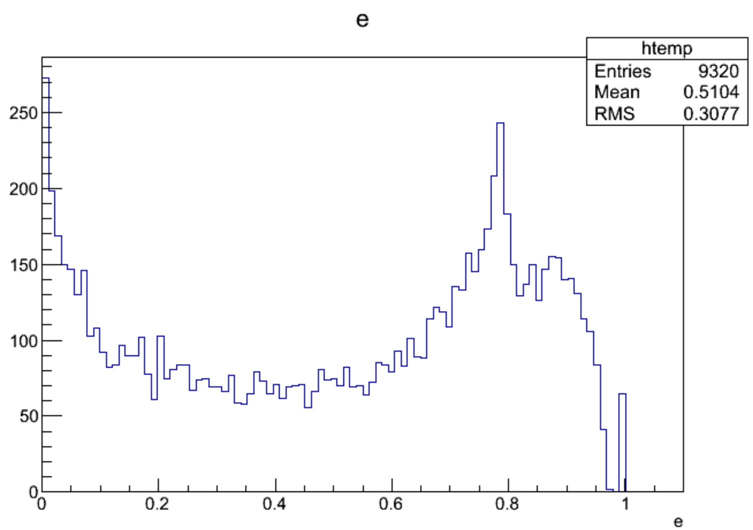
```
$GG/land_geant4 --test=d=10cm,z0=20cm,type=plastic,tree=1 \
  --gun=T=1MeV,gamma --world=type=vacuum \
  --events=10000 --tree=example4.root
```




```
root -l example4.root
```

```
h101->Print()
h101->Draw("n")
h101->Draw("e")
h101->Draw("sumE")
```

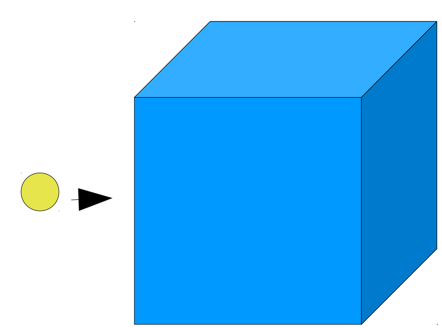





```
sumE
n
x[n]
y[n]
z[n]
e[n]
t[n]
```

Total

'Clusters'
(max 10)

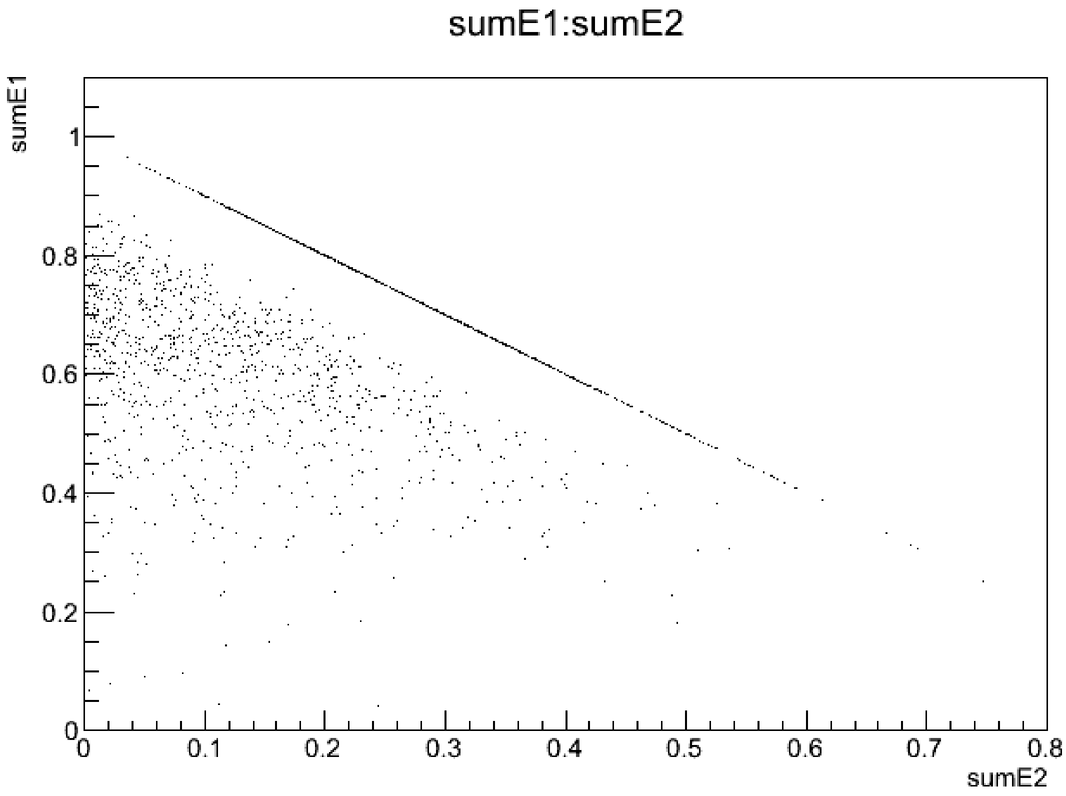


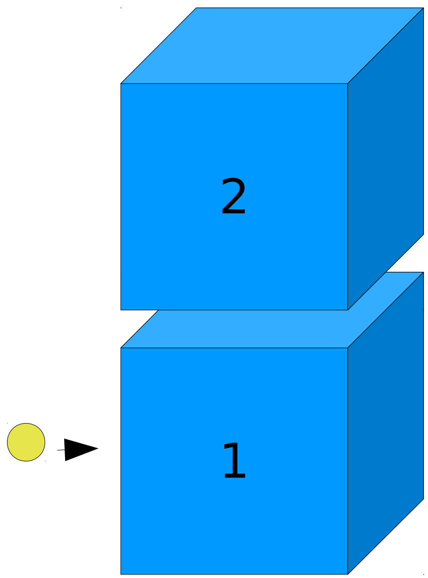
Example 5 – coincidences

```
$GG/land_geant4 --events=100000 --tree=example5.root \
  --test-1=d=1cm,z0=2cm,type=Ge,tree=1 \
  --test-2=d=1cm,z0=2cm,y0=2cm,type=Ge,tree=1 \
  --gun=T=1MeV,gamma --world=type=vacuum
```

```
root -l example5.root
```

```
h101->Draw("sumE1:sumE2")
```





Data flow

Reconstruction flow: ↓

Simulation starts at the level where analysis ends;

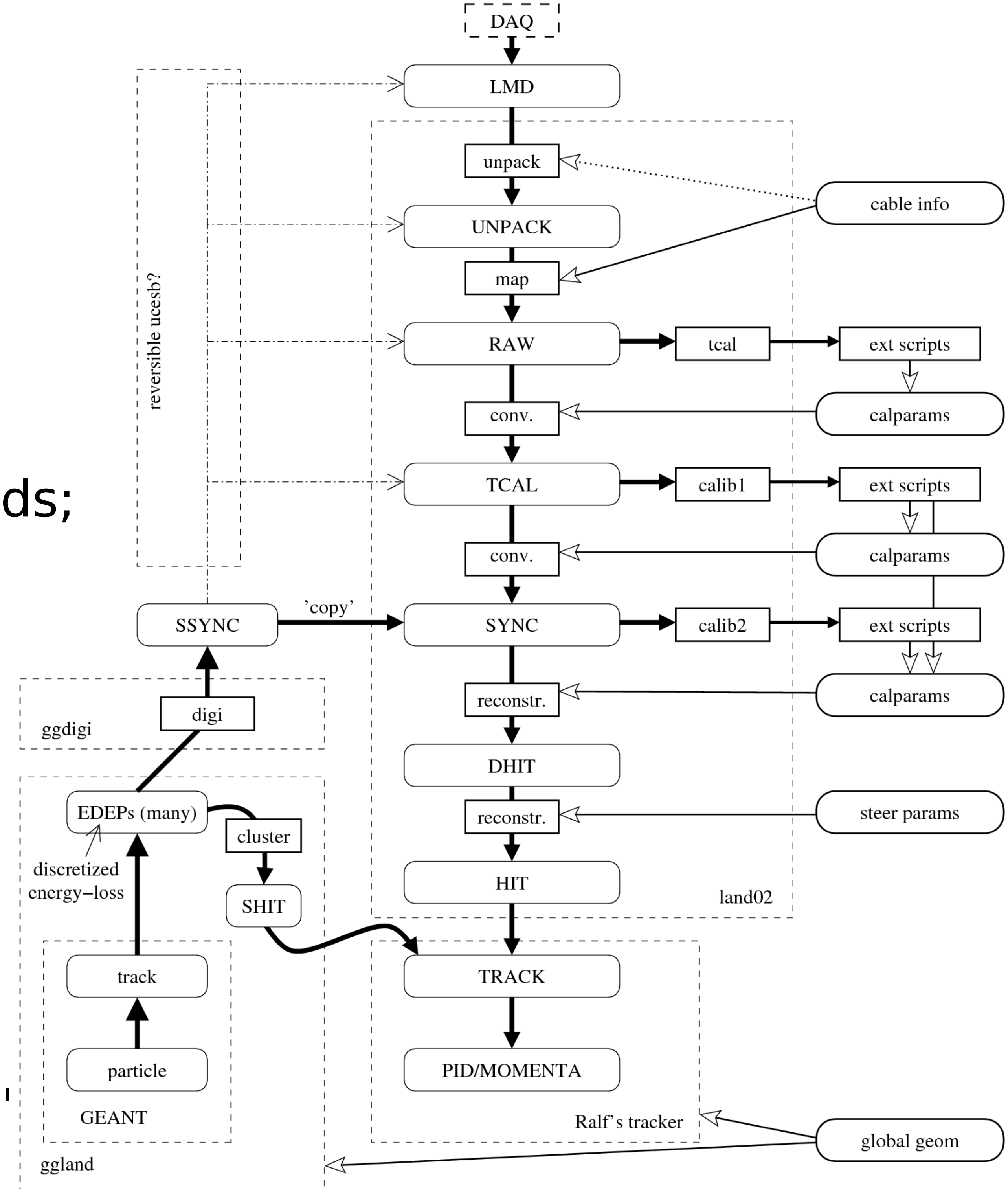
Simulation flow: ↑

Flow ↑ until suitable handover point:

(land02-speak:) SYNC level

`ggland`: wrap GEANT3/4

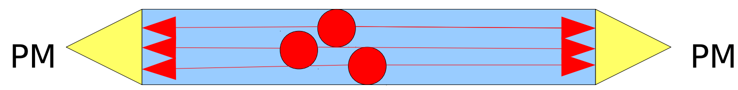
`ggdigi`: combine 'δE-hits'



ggland & ggdigi

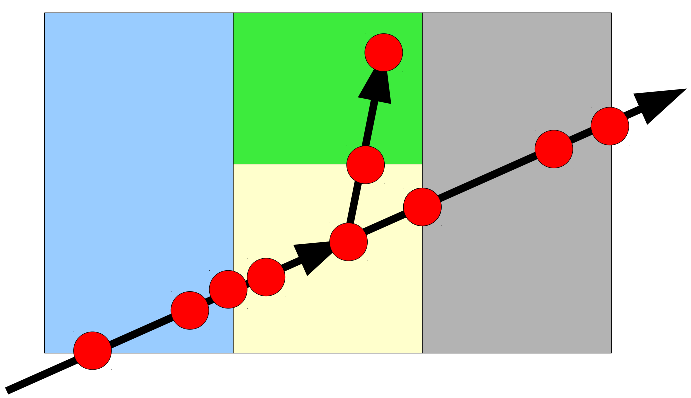
Read this way

SYNC level data (t,e) → land02



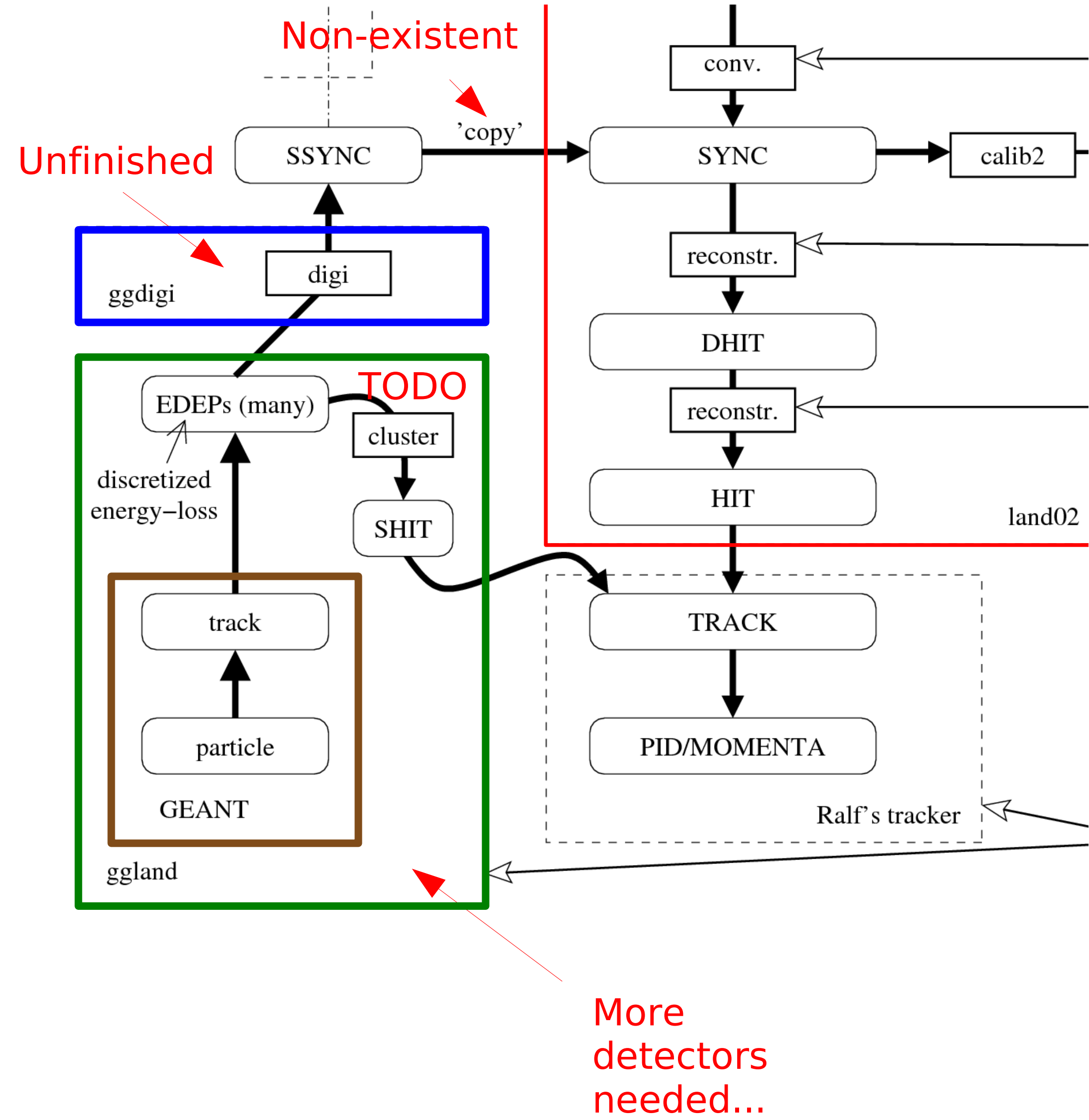
ggdigi: propagate
δE-signals to readout,
sum & discriminate

GEANT: gives 'δE-hits' (●)



GEANT: particles & tracks

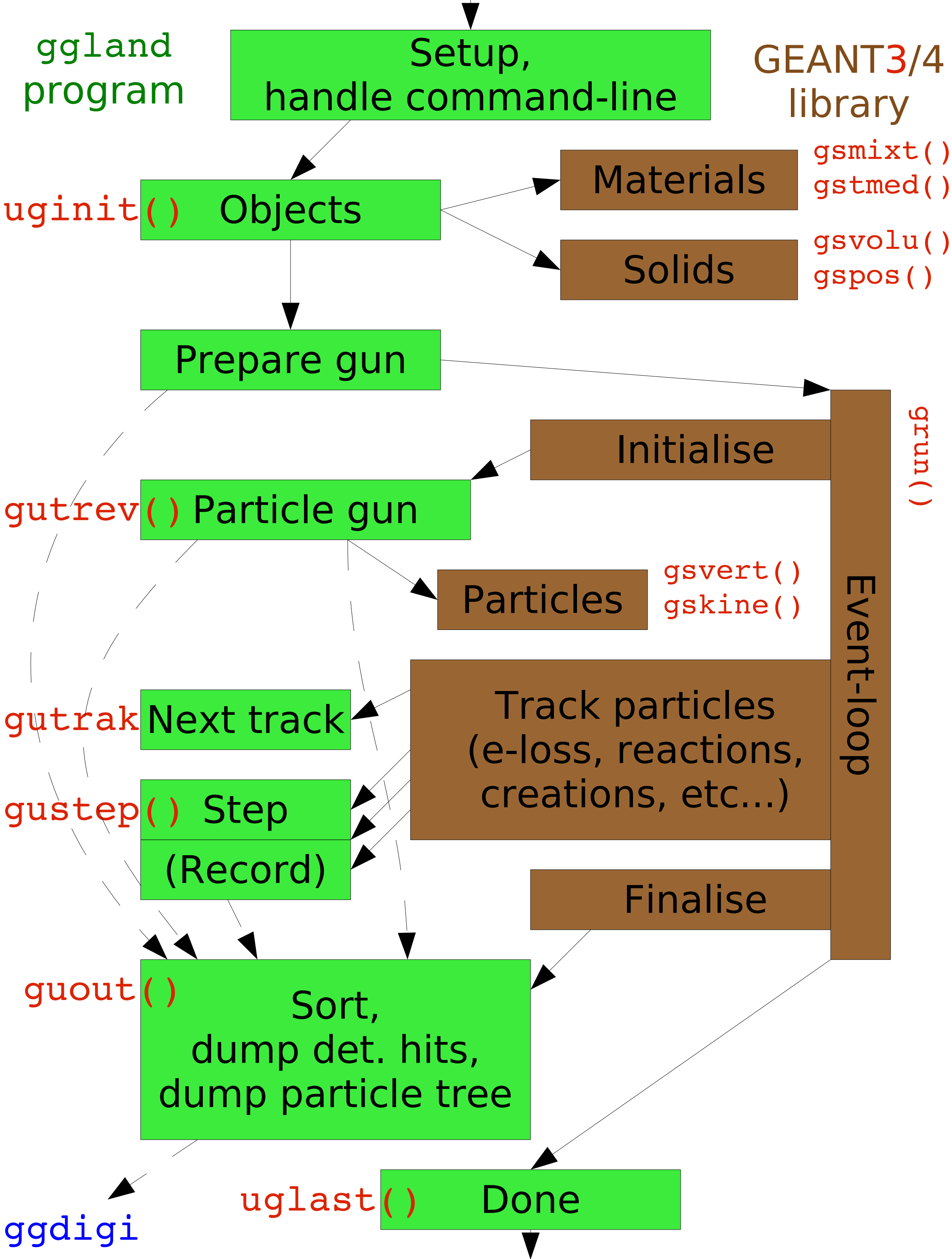
ggland: minimalistic interface



ggland: Interfacing with GEANT3/4

Interface to GEANT3:
6 user-defined functions (FORTRAN, easy to call From C).

GEANT4 has similar 'mechanics', but more flexible and C++.



ggdigi: Generic digitisation

Simulation generates list of energy deposits in volumes

```
*** EVENT 5 *** # seeds: 885310761,1134865903
hits: LAND: #       3,      2 sections
  1/18: # LAND, 2 hits (E=18.792)
        t=    41.189 p=-99.97 dE=  10.906 ds= 0.865
        t=    41.209 p=-99.29 dE=   7.886 ds= 1.125
  1/19: # LAND, 1 hits (E=3.674)
        t=    41.280 p=-99.35 dE=   3.674 ds= 0.490
# total E=22.466
```

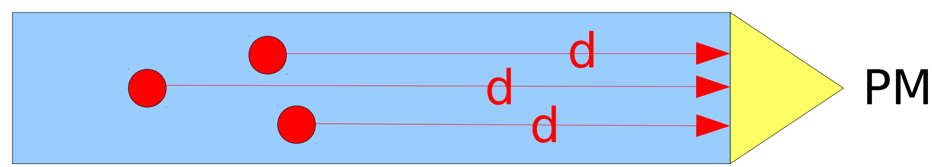
Often many!
Depends on discretization.

```
hits: TEST: #    49,    1 sections
  1: # TEST, 49 hits (E=9.073)
        t=    0.032 p= -0.49 dE=   0.000 ds= 0.000 # 2
        t=    0.032 p= -0.49 dE=   0.000 ds= 0.000 # 7
        t=    0.032 p= -0.49 dE=   0.000 ds= 0.968 # 1
        t=    0.035 p= -0.51 dE=   0.316 ds= 0.072 # 2
        t=    0.037 p= -0.56 dE=   0.173 ds= 0.057 # 2
        t=    0.038 p= -0.58 dE=   0.185 ds= 0.051 # 2
        t=    0.038 p= -0.58 dE=   0.852 ds= 0.183 # 7
        t=    0.040 p= -0.59 dE=   0.197 ds= 0.043 # 2
        t=    0.041 p= -0.58 dE=   0.161 ds= 0.039 # 2
        t=    0.042 p= -0.58 dE=   0.022 ds= 0.009 # 2
        t=    0.043 p= -0.58 dE=   0.113 ds= 0.035 # 2
        t=    0.044 p= -0.57 dE=   0.716 ds= 0.163 # 7
        t=    0.044 p= -0.58 dE=   0.129 ds= 0.035 # 2
        t=    0.045 p= -0.57 dE=   0.126 ds= 0.035 # 2
        t=    0.047 p= -0.56 dE=   0.108 ds= 0.035 # 2
        t=    0.048 p= -0.54 dE=   0.107 ds= 0.035 # 2
        t=    0.048 p= -0.55 dE=   0.000 ds= 0.000 # 8
        t=    0.048 p= -0.55 dE=   0.622 ds= 0.127 # 7
        t=    0.049 p= -0.55 dE=   0.024 ds= 0.014 # 8
        t=    0.049 p= -0.57 dE=   0.118 ds= 0.035 # 2
        t=    0.050 p= -0.58 dE=   0.000 ds= 0.000 # 9
        t=    0.050 p= -0.58 dE=   0.388 ds= 0.063 # 7
        t=    0.051 p= -0.57 dE=   0.113 ds= 0.035 # 2
        t=    0.051 p= -0.59 dE=   0.038 ds= 0.011 # 9
        t=    0.052 p= -0.56 dE=   0.123 ds= 0.035 # 2
        t=    0.054 p= -0.56 dE=   0.000 ds= 0.000 # 4
        t=    0.054 p= -0.56 dE=   0.000 ds= 0.000 # 5
        t=    0.054 p= -0.56 dE=   0.143 ds= 0.012 # 2
```

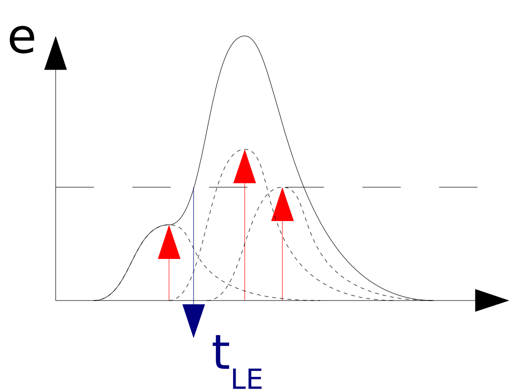
Analysis needs single data per readout channel (like real data)

Digitiser propagates to readout:

- Time-delay `t += d * v`$_{eff}$
- Attenuation `e *= exp (-d/l`$_{att.len.}$`)`



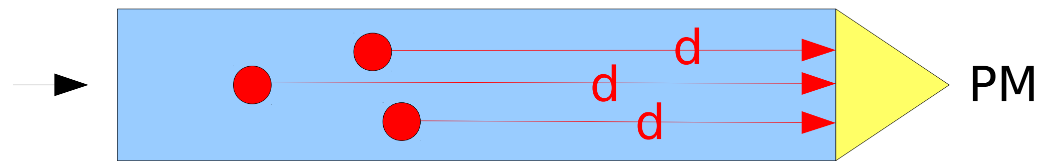
- Each hit contributes to readout pulse.
- Final time-signal after CFD (or LE) mimic



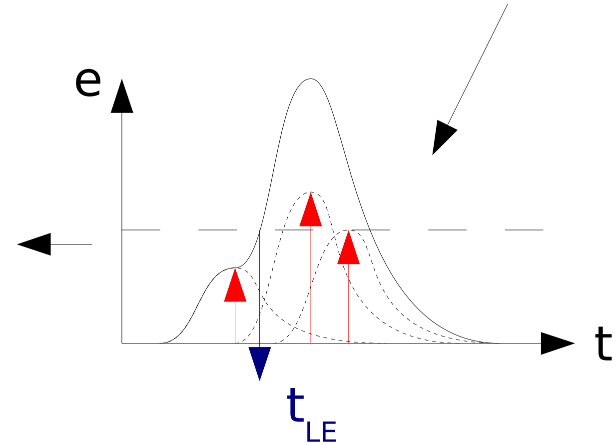
Propagation and CFD based on
`R3BConstantFraction.{h,cxx}`
By Johan Gill & Staffan Wranne

ggdigi: Generic digitisation II

```
*** EVENT 5 *** # seeds: 885310761,1134865903
hits: LAND: #      3,      2 sections
  1/18: # LAND, 2 hits (E=18.792)
         t=    41.189 p=-99.97 dE=   10.906 ds= 0.865
         t=    41.209 p=-99.29 dE=    7.886 ds= 1.125
  1/19: # LAND, 1 hits (E=3.674)
         t=    41.280 p=-99.35 dE=    3.674 ds= 0.490
# total E=22.466
```




```
*** EVENT 5 ***
# LAND 2 sections 3 hits
 1/18: t1=   35.189 e1=   10.079 t2=   47.202 e2=   35.039
 1/19: t1=   35.295 e1=    1.975 t2=   47.265 e2=    6.836
```



ggdigi: Separate process, same routine for all detectors!

Customised by:

- Relevant propagation coordinate
- Speed of signal, e.g. v_{eff} of scintillation photons
- Attenuation length
- (To handle Birk's law – light output saturation)

Parametrised detectors

Existing:

```
--land
--sciland
--xb
--csi
--cave
--boxcave
--test
--world
```

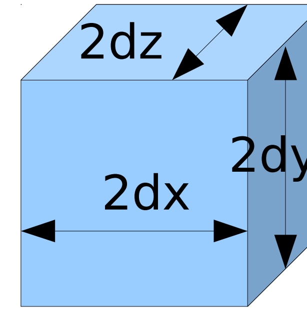
Detector volumes created by C code,
by calling wrappers for GEANT3/4.

Dimensions and aspects variable controlled.
Defaults give 'normal' detectors.
Override on command line.

```
--xb=help

rot{x,y,z}|x0|y0|z0                 0 1|cm  Rotation | Location.
r1                                 25 cm    Inner radius.
r2                                 45 cm    Outer radius.
right                               1       Right half (1-81,!77,!81).
left                                1       Left half (83-162,!82).
forward                             0       Forward crystal (81).
backward                            0       Backward crystal (82).
bottom                              0       Bottom crystal (77).
all                                 0       Enable all crystals.
crystal                           NaI       Crystal material.
```

TEST volume




```
--test=help

rot{x,y,z}|x0|y0|z0      0 1|cm   Rotation | Location.
d                          1 cm    Half width (if none of dx,dy,dz given).
dx                         1 cm    Half width (x).
dy                         1 cm    Half width (y).
dz                         1 cm    Half width (z).
dm                       nan g/cm2 Thickness (if none of dx,dy,dz given).
dmx                      nan g/cm2 Thickness (x).
dmy                      nan g/cm2 Thickness (y).
dmz                      nan g/cm2 Thickness (z).
out_col                   -1       Output column.  (Text dump.)
stepprint                  0       Debug printing from the step function.
type               undefined       Test material.
```

Material must be given
for test volume:

`--list-materials`

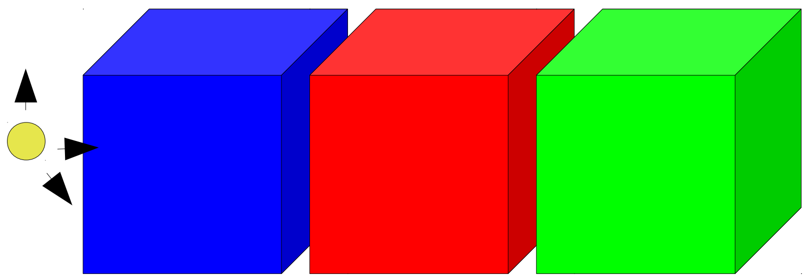
Energy sum per event.
First column is 0.
(Only non-0 events printed.)
For ROOT's `Ttree::ReadFile()`

test3.txt:

```
0.71876
0.05768
0.00663
0.68386
0.06915
0.79547
0.61802
0.69747
0.07892
...
```

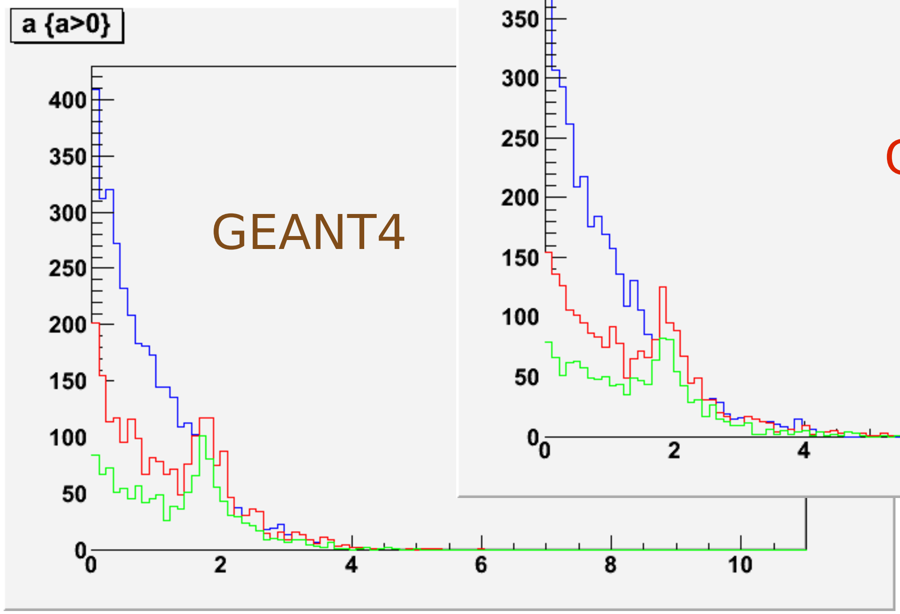
Multiple TEST volumes

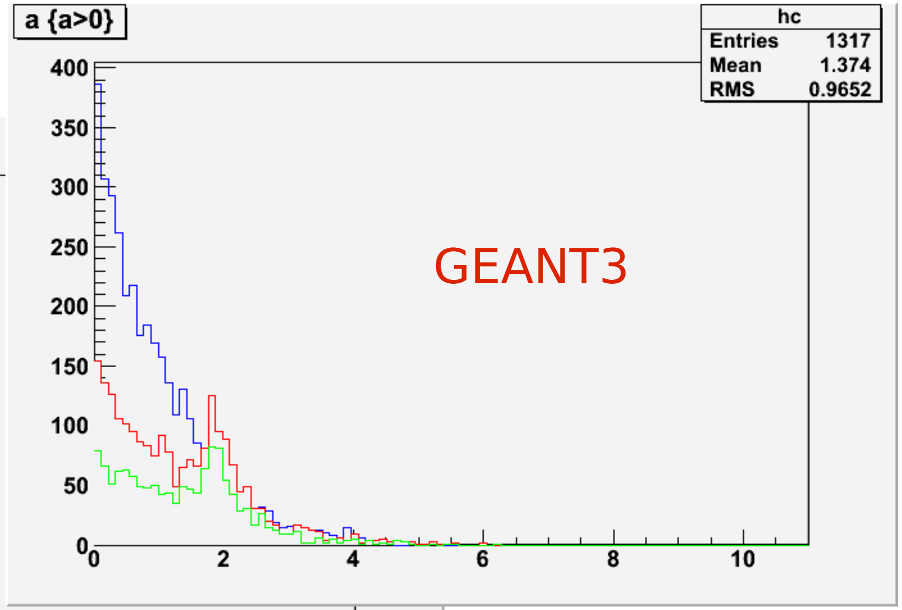
```
./land_geant3
  --test-0=d=0.5cm,z0=1cm,type=plastic,out_col=0
  --test-1=d=0.5cm,z0=2cm,type=plastic,out_col=1
  --test-2=d=0.5cm,z0=3cm,type=plastic,out_col=2
  --gun=gamma,T=10MeV,dtheta=0.5pi --events=1000000 |
  grep TSTDMP | sed -e "s/.*://" > 3vols.txt
```



3vols.txt:

```
    2.26846     0.00000     0.00000
    2.14312     1.33287     0.00000
    0.95297     0.00000     0.00000
    0.07750     0.00000     0.00000
    0.27885     0.00000     0.00000
    0.00000     1.23321     0.00000
    0.53523     0.93503     0.00000
    1.06333     0.00000     0.00000
    0.00000     0.79065     0.00000
    3.89404     1.62902     0.00000
    0.14590     2.33030     1.07062
    0.00000     0.00000     0.47722
    0.00000     1.79729     2.36696
    0.93408     0.00000     0.00000
...
```





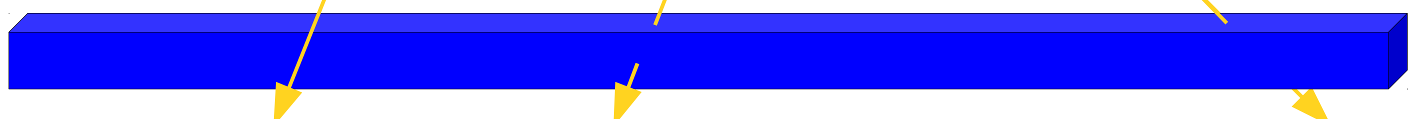
NEW

Instant .root-file

Back-of-the-envelope simulations

Create root-file with data from test volumes:
`sumE`

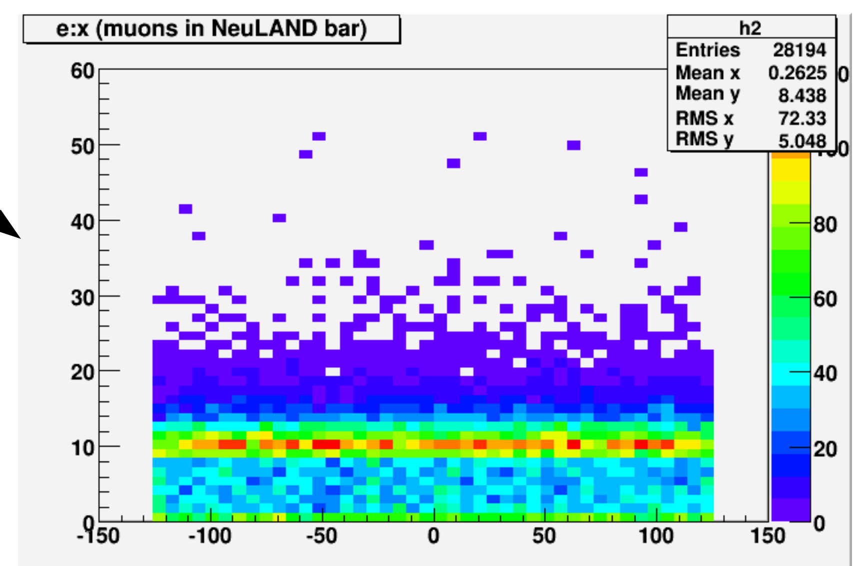
By rough 'digitizer': cluster by distance (currently sqrt(10) cm):
`n, x[n], y[n], z[n], t[n], e[n]`



```
./land_geant4 --test=dx=125cm,dy=2.5cm,dz=2.5cm,type=plastic,tree=1
   --gun=cosmmuonthetaE=250MeV,sr=130cm --tree=muons2.root
   --events=1000000
```

Include volume

`h101->Draw("e:x")`



Realistic muon source now
also gives corresponding real time.

```
Muon source (lowEcut=250.0 MeV, r=130.0 cm):
   avg time between ev. 0.0009705 s.
Muon source with 1000000 events, total time:
   970.5 s (= 0 h 16 m).
```

Vertex & hit list – markup: ggmark

```
--hits=vertex,- |
  ./ggmark |
  less -R
```

Particle & vertex list

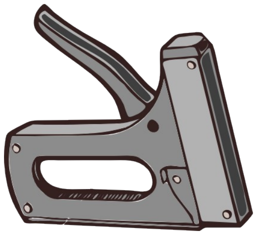
```
*** EVENT 5 *** # seeds: 55312060,333625511
# Vertices 7, particles 9
# bullet pdg=22    T=   10.000 theta=120.394deg phi=-171.797deg gamma
origin: t=   0.000 xyz=(    0.000,    0.000,    0.000) {
  gamma:    [1] pxyz=(   -8.537,   -1.231,   -5.059) T=   10.000
   t_delay=   0.000 pdg=22     te=   0.032 se=    0.968 # pid=1
    vertex: t=   0.032 xyz=(   -0.827,   -0.119,   -0.490) s=    0.968
            PAIR    pxyz=(   -8.537,   -1.231,   -5.059) STOP T=   10.000 {
      e+:       [2] pxyz=(   -2.856,   -0.591,   -1.332) T=    2.736
       t_delay=   0.000 pdg=-11    te=   0.054 se=    0.604 # pid=2
        vertex: t=   0.042 xyz=(   -0.973,   -0.200,   -0.582) s=    0.272
                BREM    pxyz=(    0.842,   -1.470,    0.158) T=    1.266 {
          gamma:    [3] pxyz=(    0.273,   -0.253,    0.185) T=    0.416
           t_delay=   0.000 pdg=22     te= 101.588 se=   escape # pid=1
        }
        vertex: t=   0.054 xyz=(   -0.949,   -0.231,   -0.561) s=    0.604
                ANNI    pxyz=(    0.000,   -0.000,   -0.000) STOP T=    0.000 {
          gamma:    [4] pxyz=(   -0.065,   -0.125,    0.491) T=    0.511
           t_delay=   0.000 pdg=22     te=  69.472 se=   escape # pid=1
          gamma:    [5] pxyz=(    0.065,    0.125,   -0.491) T=    0.511
           t_delay=   0.000 pdg=22     te=   0.069 se=    0.451 # pid=1
            vertex: t=   0.062 xyz=(   -0.916,   -0.169,   -0.804) s=    0.253
                    COMP    pxyz=(    0.192,    0.135,    0.078) T=    0.247 {
              e-:       [6] pxyz=(   -0.126,   -0.010,   -0.569) T=    0.264
               t_delay=   0.000 pdg=11     te=   0.062 se=    0.035 # pid=3
            }
        }
      e-:       [7] pxyz=(   -5.603,   -0.628,   -3.681) T=    6.242
       t_delay=   0.000 pdg=11     te=   0.071 se=    1.161 # pid=3
        vertex: t=   0.048 xyz=(   -0.794,   -0.314,   -0.546) s=    0.473
                BREM    pxyz=(    1.088,    3.664,   -2.395) T=    4.028 {
          gamma:    [8] pxyz=(    0.007,    0.018,   -0.014) T=    0.024
           t_delay=   0.000 pdg=22     te=   0.049 se=    0.014 # pid=1
        }
        vertex: t=   0.050 xyz=(   -0.780,   -0.268,   -0.576) s=    0.535
                BREM    pxyz=(    1.443,    0.913,   -3.707) T=    3.602 {
          gamma:    [9] pxyz=(    0.014,    0.010,   -0.034) T=    0.038
           t_delay=   0.000 pdg=22     te=   0.051 se=    0.011 # pid=1
        }
    }
}
```

Particle-hit association

Hit (δE) list

```
hits: TEST: #     49,     1 sections
  1: # TEST, 49 hits (E=9.073)
        t=    0.032 p= -0.49 dE=    0.000 ds= 0.000 # 2
        t=    0.032 p= -0.49 dE=    0.000 ds= 0.000 # 7
        t=    0.032 p= -0.49 dE=    0.000 ds= 0.968 # 1
        t=    0.035 p= -0.51 dE=    0.316 ds= 0.072 # 2
        t=    0.037 p= -0.56 dE=    0.173 ds= 0.057 # 2
        t=    0.038 p= -0.58 dE=    0.185 ds= 0.051 # 2
        t=    0.038 p= -0.58 dE=    0.852 ds= 0.183 # 7
        t=    0.040 p= -0.59 dE=    0.197 ds= 0.043 # 2
        t=    0.041 p= -0.58 dE=    0.161 ds= 0.039 # 2
        t=    0.042 p= -0.58 dE=    0.022 ds= 0.009 # 2
        t=    0.043 p= -0.58 dE=    0.113 ds= 0.035 # 2
        t=    0.044 p= -0.57 dE=    0.716 ds= 0.163 # 7
        t=    0.044 p= -0.58 dE=    0.129 ds= 0.035 # 2
        t=    0.045 p= -0.57 dE=    0.126 ds= 0.035 # 2
        t=    0.047 p= -0.56 dE=    0.108 ds= 0.035 # 2
        t=    0.048 p= -0.54 dE=    0.107 ds= 0.035 # 2
        t=    0.048 p= -0.55 dE=    0.000 ds= 0.000 # 8
        t=    0.048 p= -0.55 dE=    0.622 ds= 0.127 # 7
        t=    0.049 p= -0.55 dE=    0.024 ds= 0.014 # 8
        t=    0.049 p= -0.57 dE=    0.118 ds= 0.035 # 2
        t=    0.050 p= -0.58 dE=    0.000 ds= 0.000 # 9
        t=    0.050 p= -0.58 dE=    0.388 ds= 0.063 # 7
        t=    0.051 p= -0.57 dE=    0.113 ds= 0.035 # 2
        t=    0.051 p= -0.59 dE=    0.038 ds= 0.011 # 9
        t=    0.052 p= -0.56 dE=    0.123 ds= 0.035 # 2
        t=    0.054 p= -0.56 dE=    0.000 ds= 0.000 # 4
        t=    0.054 p= -0.56 dE=    0.000 ds= 0.000 # 5
        t=    0.054 p= -0.56 dE=    0.143 ds= 0.012 # 2
```

Versatile particle gun

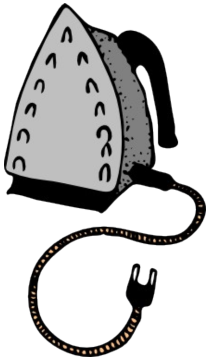


```
--gun=isotropic,gamma,E=5MeV

--gun=z0=10cm,sr=0.1cm,sphere,theta=45deg,
      dtheta=0.5pi,C12,T=200MeV/u
```

```
--gun=help
```

- Tree / conditional emission
- Source location & shape
- Direction and emission opening
- Energy / momentum
- Particle selection
- Phase-space / Lorentz boost



ggland - TODO

More detectors (POS, SST, GFI, NTF, PDC, TFW)

Mimic of QFS - (p,2p) / (p,np) – in particle gun

Quick-n-dirty digitizer/tree writer for detectors

Magnetic field (ALADiN)

Needed for project

Nice to have

Gun from file (bullets)

Auto-plot of visualisation around (interesting) vertices

???

Detector defintion sources

`dets, land, xb, box_cave, cave`

`geom_<DET>.{hh,cc}` // detector definitions

`{gg,g3,g4}_geom_help.{hh,cc}` // used by the defintions

Geant3 geometry package:

http://wwwasdoc.web.cern.ch/wwwasdoc/geant_html3/node94.html

Geant3 shapes:

http://wwwasdoc.web.cern.ch/wwwasdoc/geant/node109.html

Geant4 geometry is very similar:

http://geant4.web.cern.ch/geant4/UserDocumentation/UsersGuides/ForApplicationDeveloper/html/ch04.html

We will mostly need boxes...