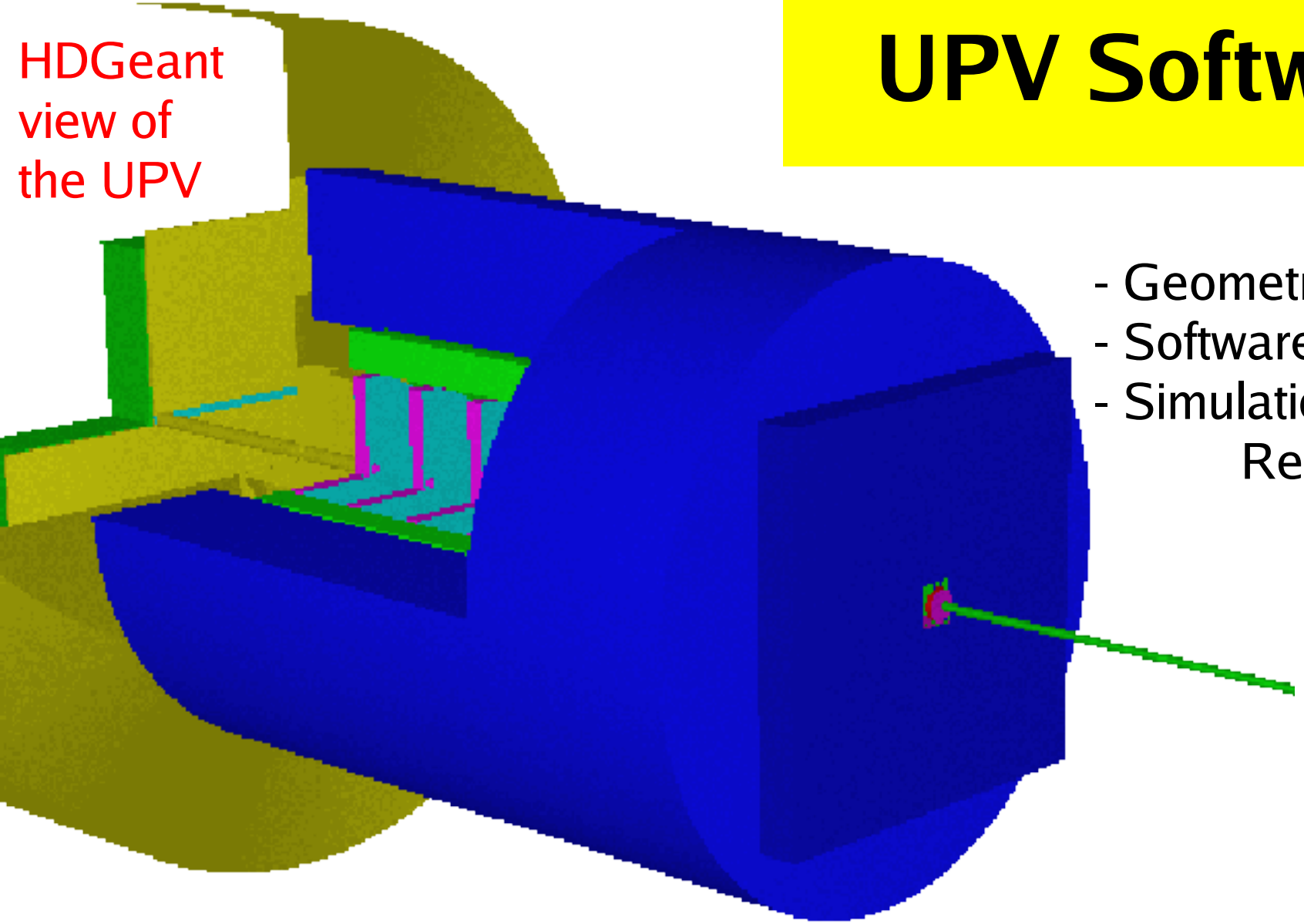
UPV Software



- Geometry
- Software Updates
- Simulations &
  Reconstruction

Paul Eugenio
Florida State University

UPV Geometry

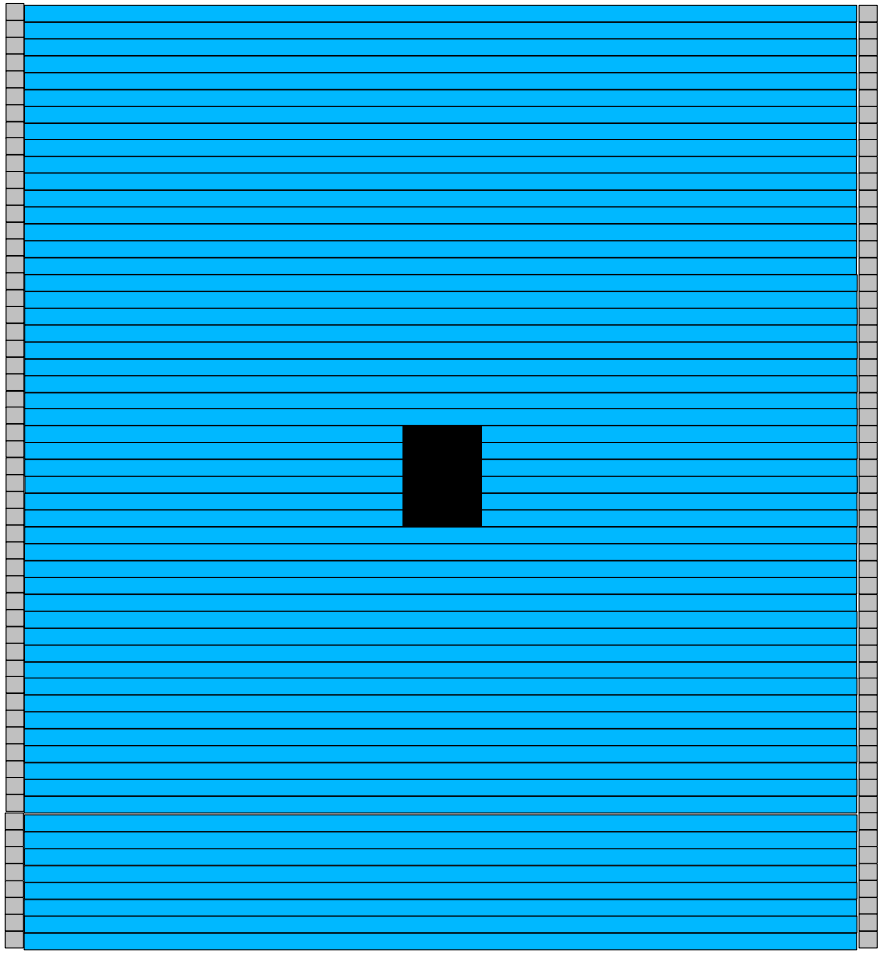
Sampling Calorimeter Design

Alternating lead/scintillator planes
- 18 layers of 1cm thick scintillator
- 12 layers of 0.25cm thick lead sheets
- 6 layers 2*0.25cm thick lead sheets (upstream)

segmentation
- 56-4.25cm x 238cm rows
- 112 FADC readout channels

beam hole
- 25.5cm square hole(6 central paddles)
- left-right continuous fiber connections

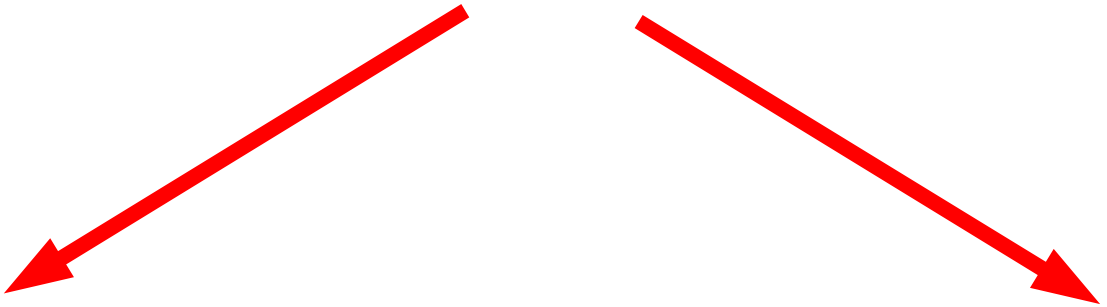
light readout on both ends
- Exploring alternate readouts



25.5cm x 25.5cm beam hole

Total UPV volume:
240cm x 240cm x 26cm

UPV Software Updates



UPVSim

Geant4 based

simulation tool for detailed UPV specific R&D studies

HDGeant

Geant3 based

HDDS

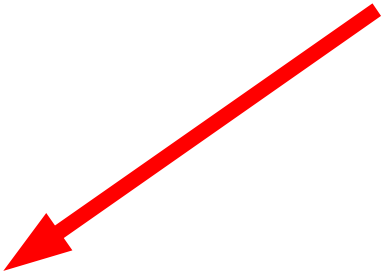
Added UPV geometry description

HDDM

Added UPV hits to the data I/O

HDGEANT

Implemented UPV hit description & hddm output



UPV Reconstruction

tools for simulation & reconstruction studies

---

goal to implement in JANA

UPV in HDDS

A new backwardPackage was created in the file main_HDDS.xml

```
<composition name="backwardPackage">
  <posXYZ volume="UpstreamEMveto" X_Y_Z="0.0 0.0 -60.8" />
</composition>

<composition name="GlueXdetector">
  <posXYZ volume="Target" X_Y_Z="0.0 0.0 50.0" />
  <posXYZ volume="barrelPackage" />
  <posXYZ volume="forwardPackage" />
  <posXYZ volume="backwardPackage" />
</composition>
 ...
```

UpstreamEMveto_HDDS.xml

```
<!-- UPV body has 12 single-width layers downstream and
    6 double-width layers upstream -->
 <composition name="UPVbody">
  <mposZ volume="UPVsingleLayer" ncopy="12" X_Y="0.0 0.0" Z0="11.375" dZ="-1.25">
   <layer value="1" step="1" />
  </mposZ>
  <mposZ volume="UPVdoubleLayer" ncopy="6" X_Y="0.0 0.0" Z0="-3.75" dZ="-1.50">
   <layer value="13" step="1" />
  </mposZ>
 </composition>

<!-- UPV single layer has 1 plane of scintillators and 1 sheet of lead -->
 <composition name="UPVsingleLayer">
  <posXYZ volume="UPVleadSheet"         X_Y_Z="0.0 0.0 +0.5"   />
  <posXYZ volume="UPVscintillatorPlane"  X_Y_Z="0.0 0.0 -0.125" />
 </composition>

<!-- UPV double layer has 1 plane of scintillators and 2 sheets of lead -->
 <composition name="UPVdoubleLayer">
  <posXYZ volume="UPVleadSheet"         X_Y_Z="0.0 0.0 +0.625" />
  <posXYZ volume="UPVleadSheet"         X_Y_Z="0.0 0.0 +0.375" />
  <posXYZ volume="UPVscintillatorPlane"  X_Y_Z="0.0 0.0 -0.25"  />
 </composition>
                                        ...
```

UPV in HDDM

Addition to the file hits.xml

Data

```
<upvRow y="float" minOccurs="0" maxOccurs="unbounded">
  <upvLeft minOccurs="0">
    <shower t="float" E="float" maxOccurs="unbounded" />
  </upvLeft>
  <upvRight minOccurs="0">
    <shower t="float" E="float" maxOccurs="unbounded" />
  </upvRight>
</upvRow>
```

cheats

```
<upvPaddle y="float" z="float" minOccurs="0" maxOccurs="unbounded" >
  <upvLeft minOccurs="0">
    <shower t="float" E="float" maxOccurs="unbounded" />
  </upvLeft>
  <upvRight minOccurs="0">
    <shower t="float" E="float" maxOccurs="unbounded" />
  </upvRight>
</upvPaddle>
```

Only in MC

truths

```
<upvShower t="float" x="float" y="float" z="float"
        E="float" track="int" primary="boolean"
        minOccurs="0" maxOccurs="unbounded" />
```

UPV in HDGEANT

File: hitUPV.c

```
void hitUpstreamEMveto (float xin[4], float xout[4],
                        float pin[5], float pout[5], float dEsum,
                        int track, int stack)
{
                              ...

  if (dEsum == 0) return;              /* only seen if it deposits energy */

  x[0] = (xin[0] + xout[0])/2;
  x[1] = (xin[1] + xout[1])/2;
  x[2] = (xin[2] + xout[2])/2;
  t    = (xin[3] + xout[3])/2 * 1e9;
  transformCoord(x,"global",xlocal,"UPV");
  transformCoord(zeroHat,"local",xupv,"UPV");

  int layer = getlayer_();
  int row = getrow_();
                        ...
```

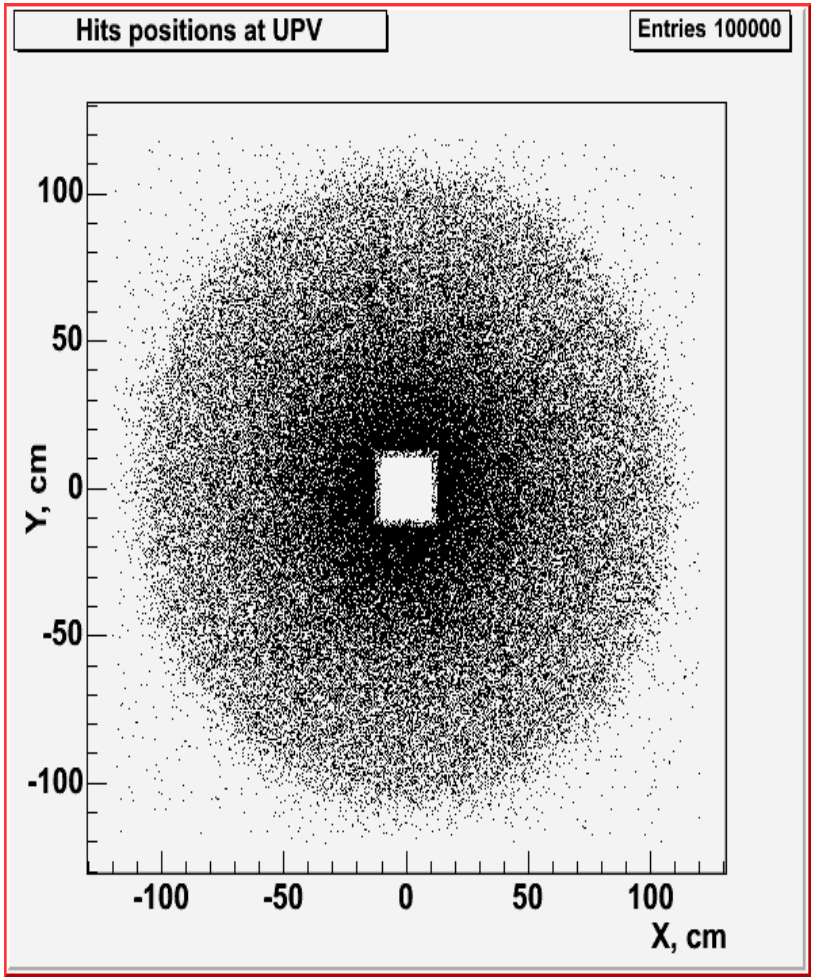
```
float dxleft = xlocal[0];
float dxright = -xlocal[0];
float tleft  = t + dxleft/C_EFFECTIVE;
float tright = t + dxright/C_EFFECTIVE;
float dEleft  = dEsum * exp(-dxleft/ATTEN_LENGTH);
float dEright = dEsum * exp(-dxright/ATTEN_LENGTH);
float ycenter = (fabs(xupv[1]) < 1e-4) ? 0 : xupv[1];
float zcenter = (fabs(xupv[2]) < 1e-4) ? 0 : xupv[2];
                              ...
```

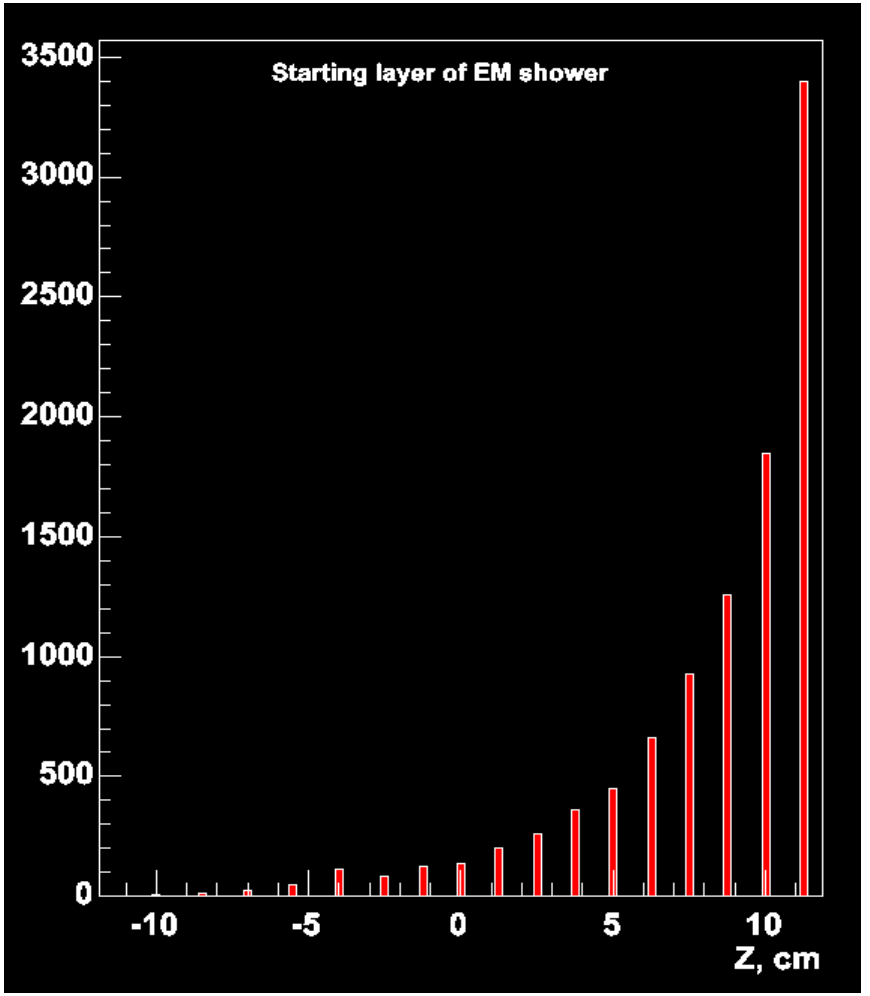
Simulations & Reconstruction

HDGeant Monte Carlo Study

10 MeV < Eγ < 500 MeV

random angles covering all of UPV

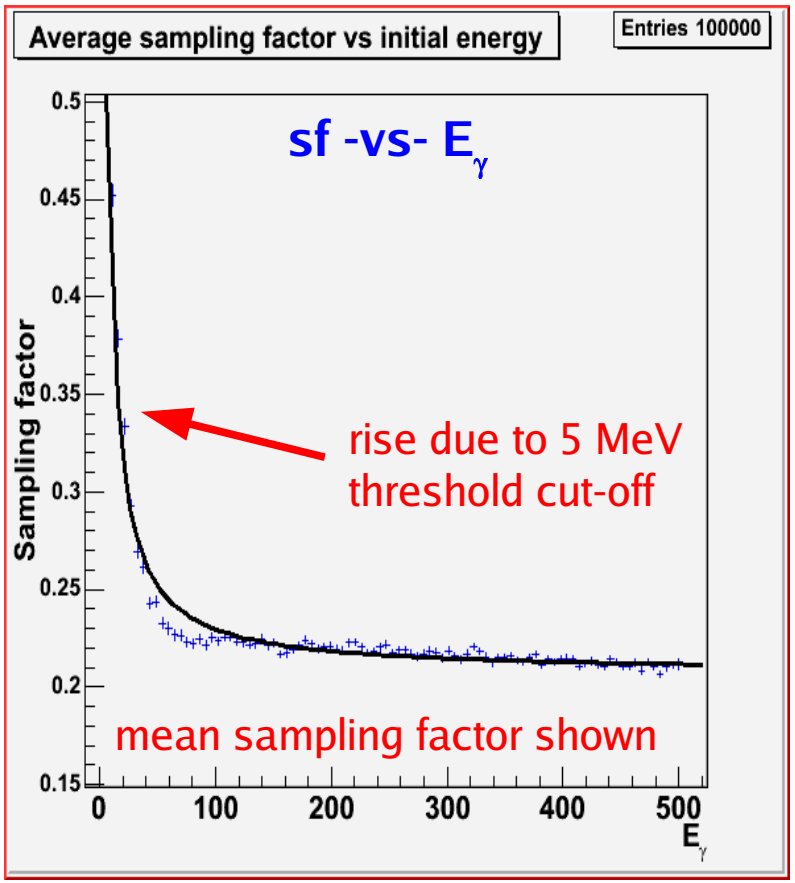


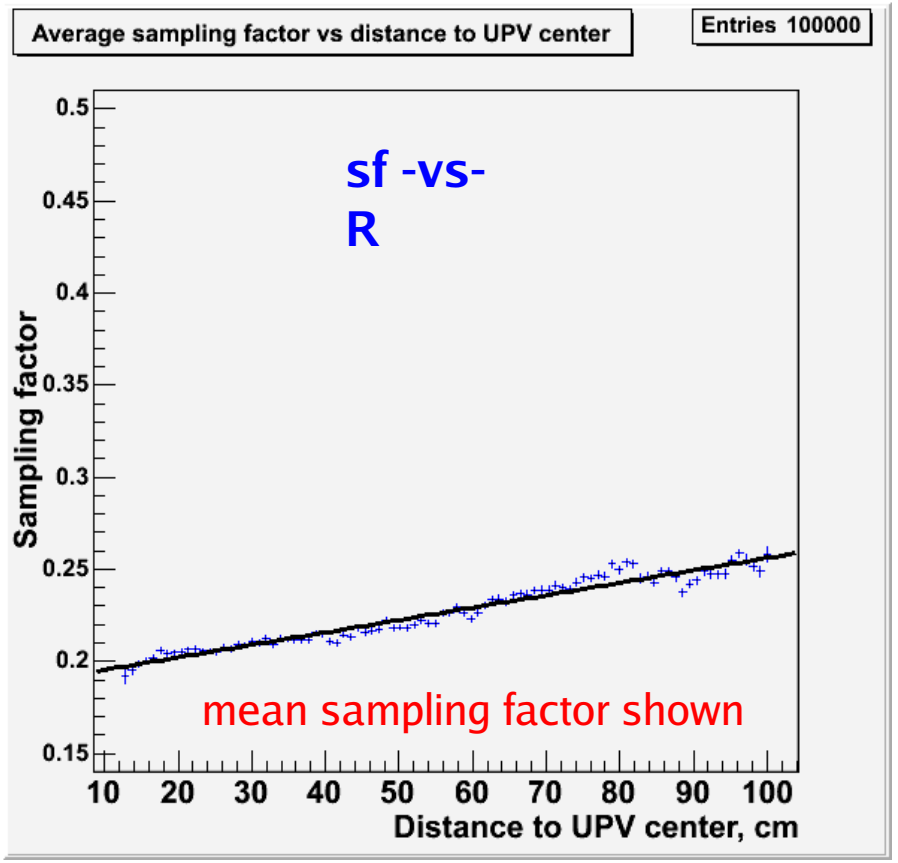


Sampling Factor

Energy & Angular Dependence

$$sf = E_{seen}/E_{\gamma}$$

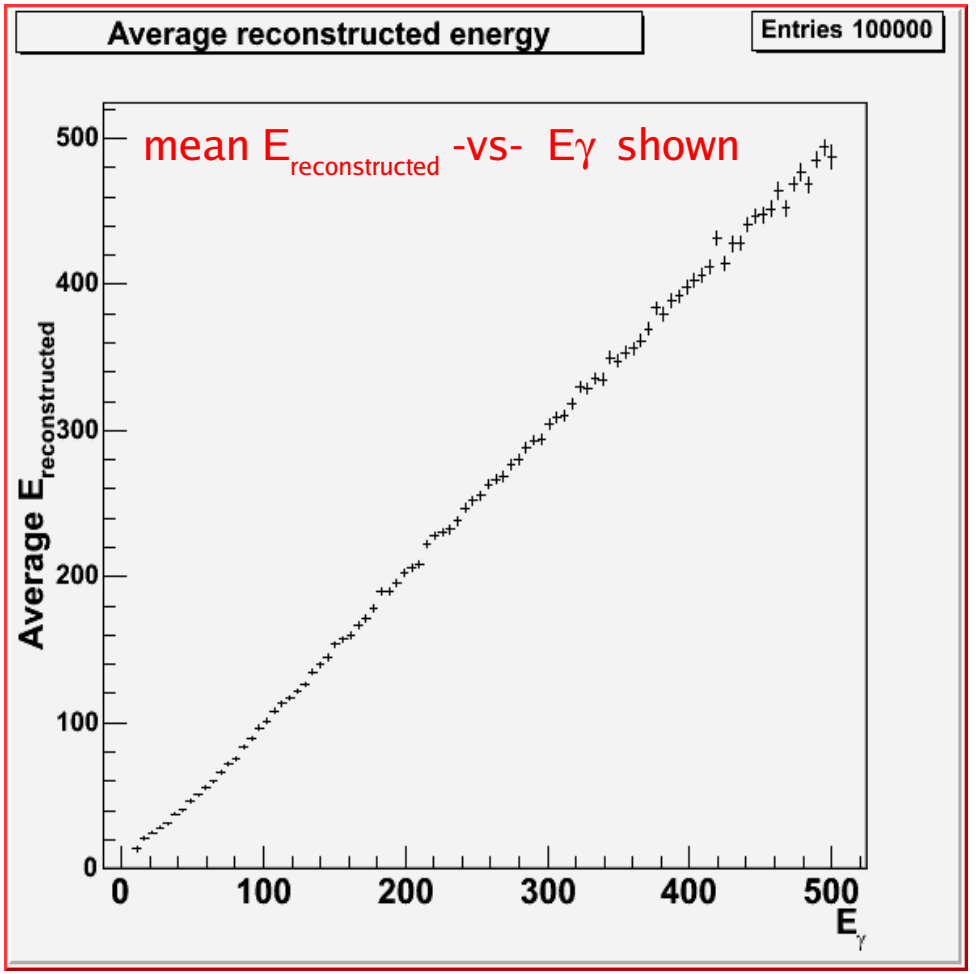




Reconstructed Cluster Energy

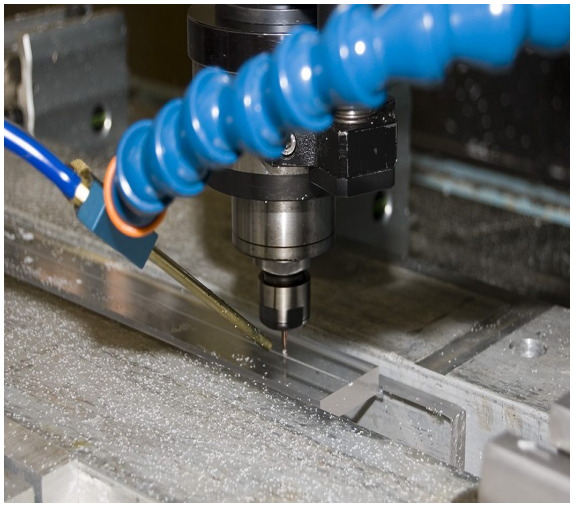
To account for both energy and angular dependence of the sampling factor, the reconstructed cluster energy $E_{cluster}$ was determined from both the detected energy E_{seen} and position of the hit R according to the formula:

$$E_{cluster} = \frac{(a1 * E_{seen} + a2)}{(1 + a3 * R)}$$

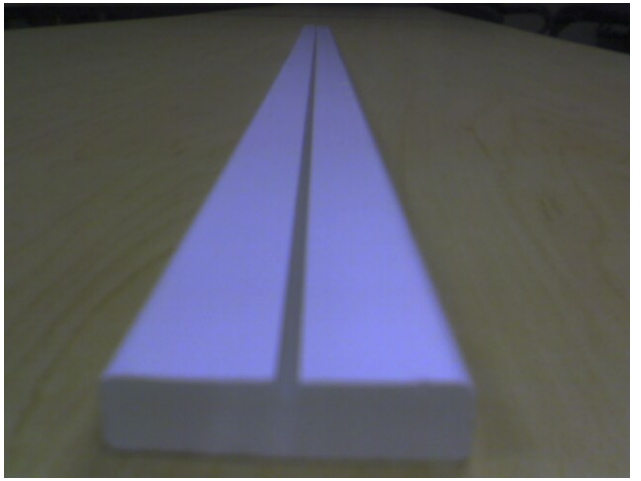


a1, a2, & a3 were determined by fitting σ_E/E

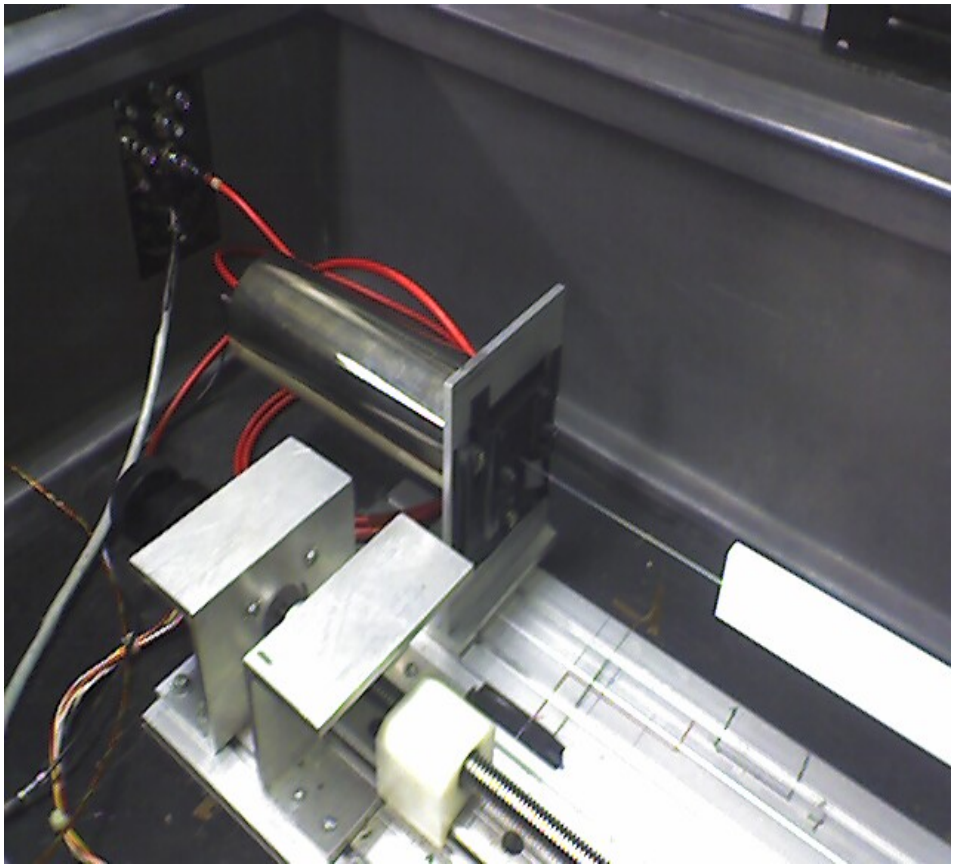
Some Issues

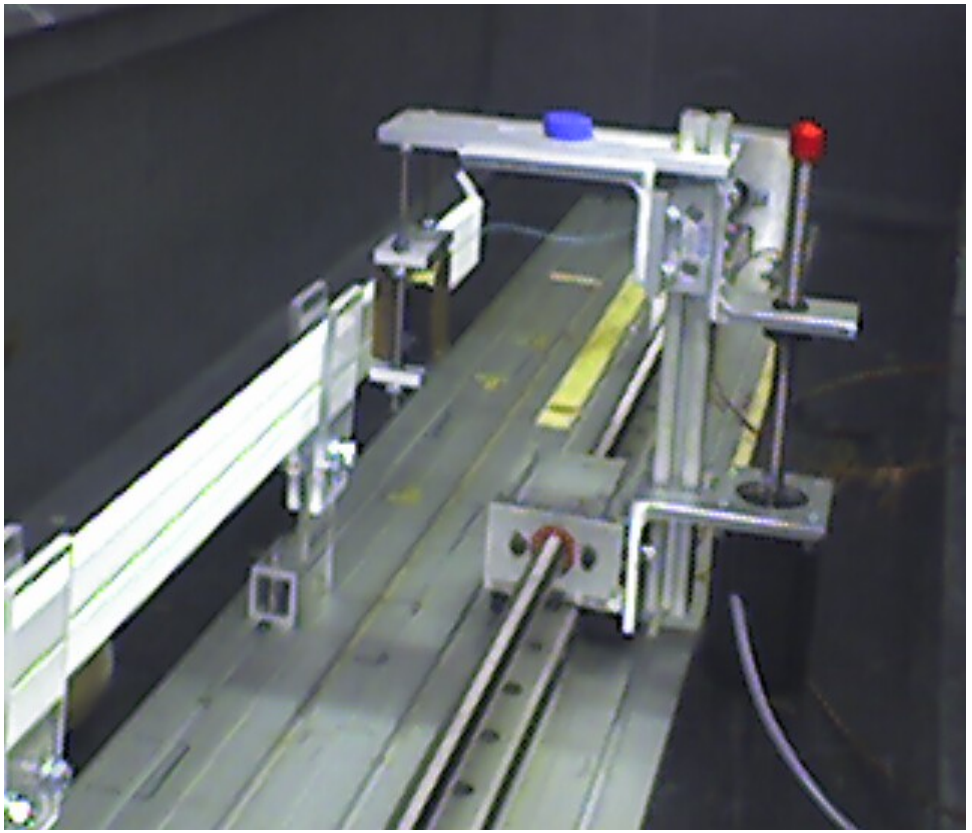


- **Continued simulation and reconstruction studies**
  - Implement into JANA
- **Light Collection**
  - Simulate WLS fiber readout
  - Add proper WLS fiber attenuations
- **Signal Electronics**
  - Implement FADC data
- **R&D with Extruded -vs- Cast Scintillators**
  -



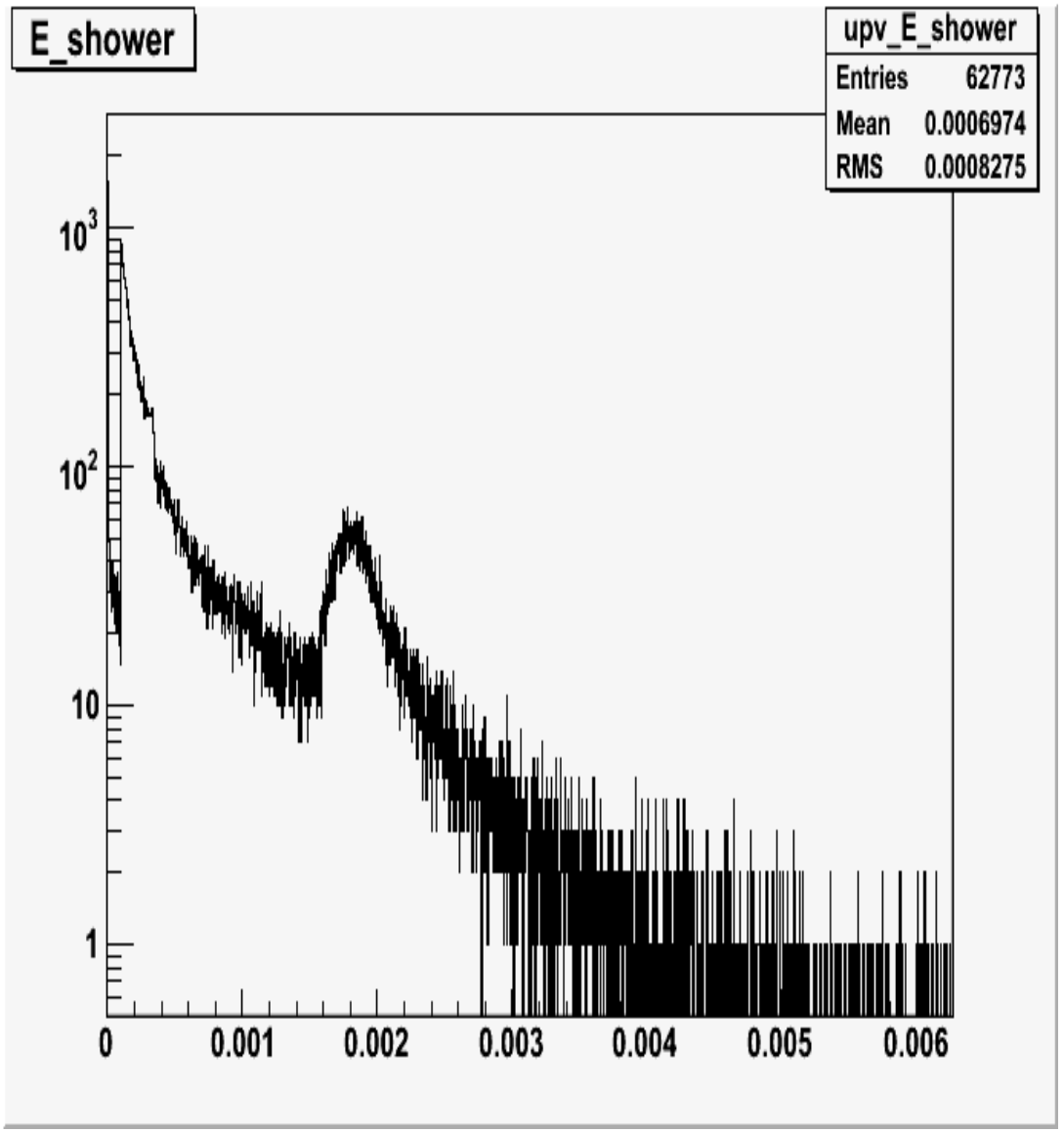
R&D on WLS Fibers & Extruded Scintillator

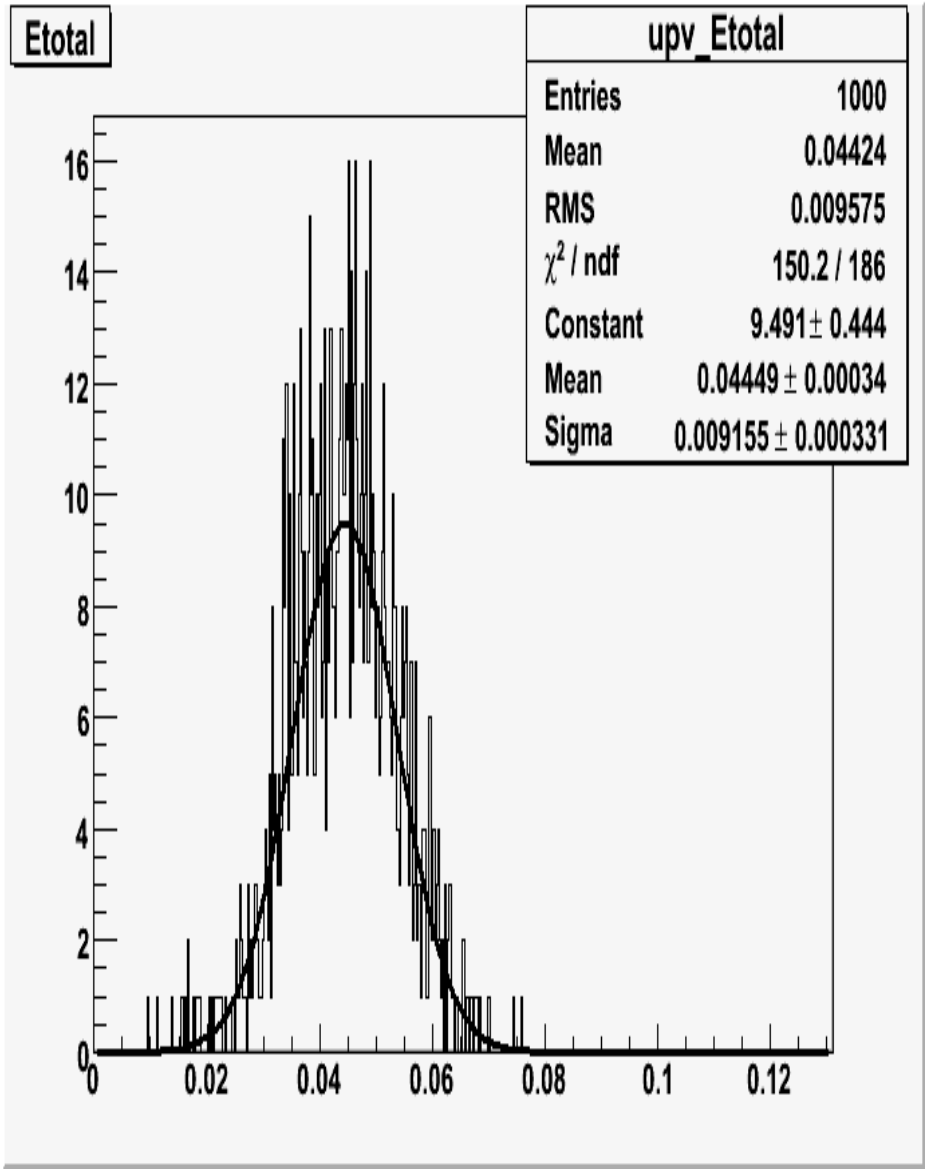




UPV Reconstruction in JANA

Plots from David Lawrence





UPV Reconstruction in JANA

Plots from David Lawrence