Timing Detector at Panda(root):

The Scintillator Tile detector (SciTil)

Alicia Sanchez Lorente

Helmholtz Institut Mainz

Panda Computing Workshop

Torino, 23 – 27 July 2012

Timing Detector for PANDA

- **PANDA no hardware Trigger**

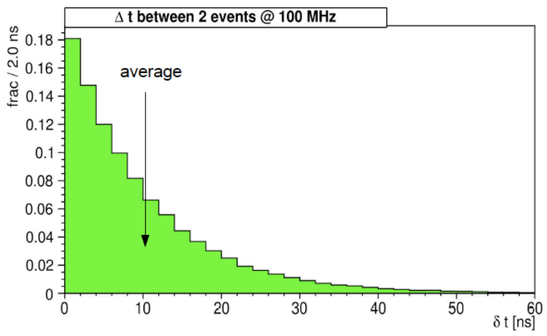
- ~100MHz event rate

- Complete processing of all events no possible
- Software trigger: high rejection factors
  by means of simple signatures

- Event building (*See Tobias Stockmanns's talk*)

Event timing

Time between successive events are **not equally spaced** but follow a **exponential distribution**



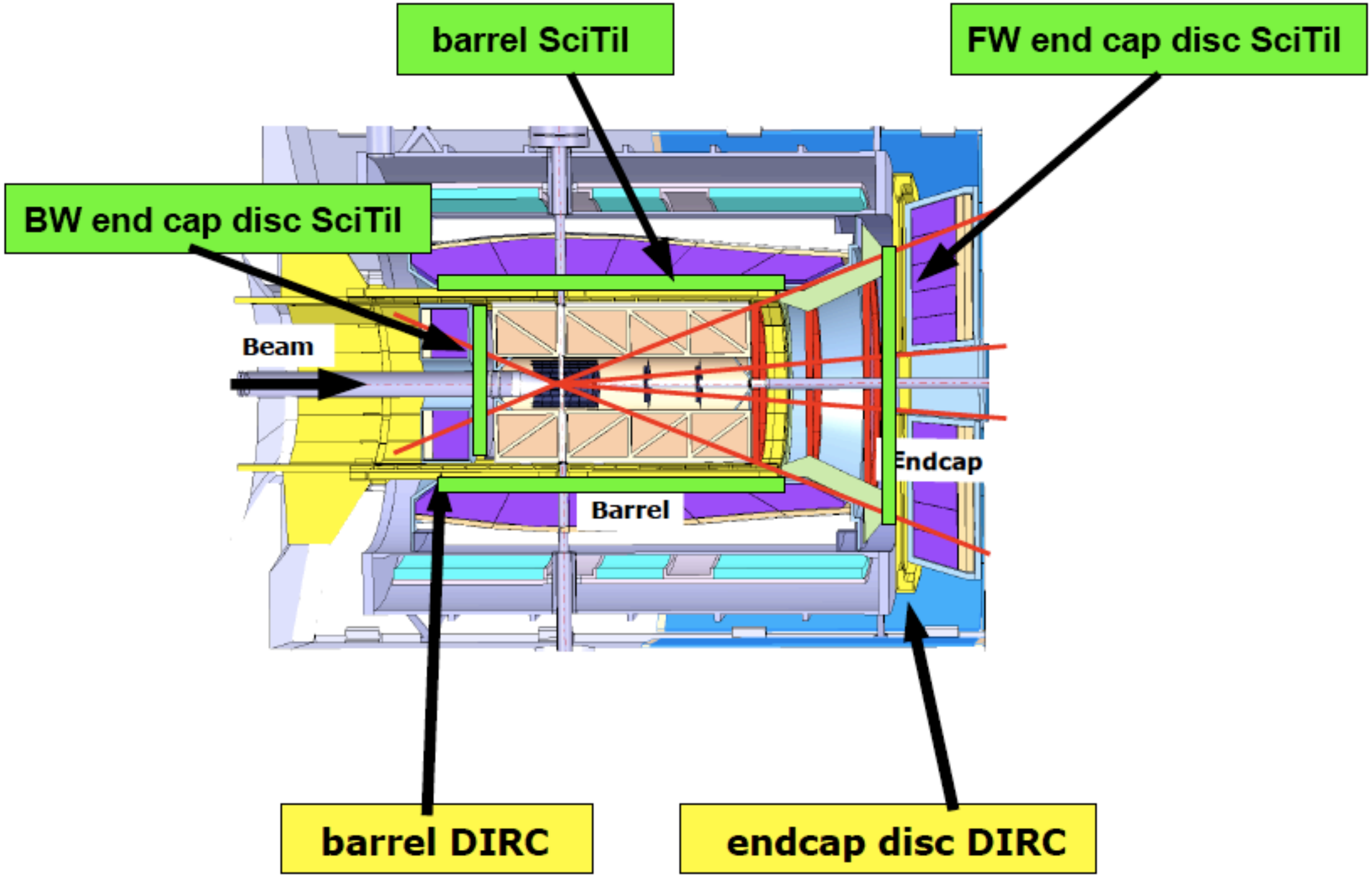
•Fast timing (100 ps) : essential for event building algorithm

• Event timing: disentangle signals of different events
•Photon Conversion Detection (DIRC)

•charged/ neutral discrimination for EMC
•High granularity -> Tracking
•PID : time-Of-Flight

Panda Detector

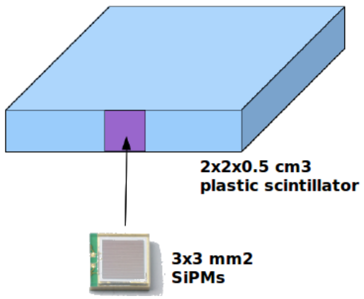
PANDA interaction rate:
Average 20MHz
Peak 50-100MHz

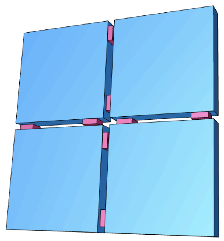


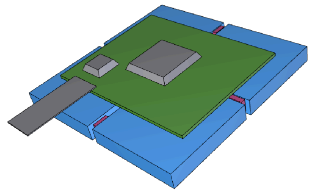
Scintillator Tile Detector Geometry

- Low material budget :1% radiation length

1. Four tiles arranged with their SiPMs for densest packing:
2. A quad module with a R&D PCB based on 8-channel readout ASIC and a data transfer chip.

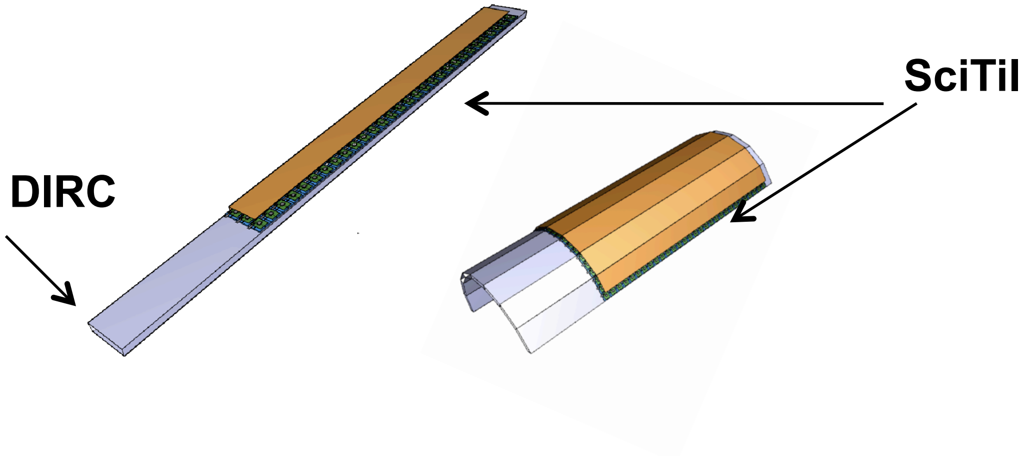




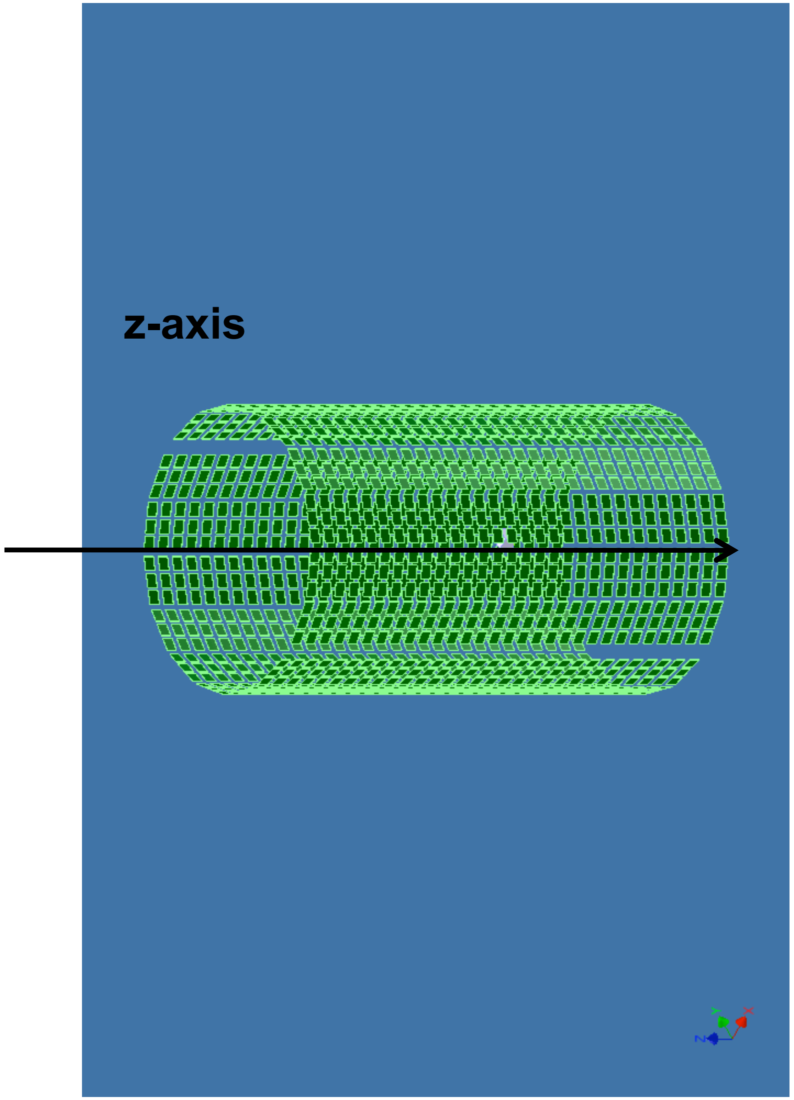


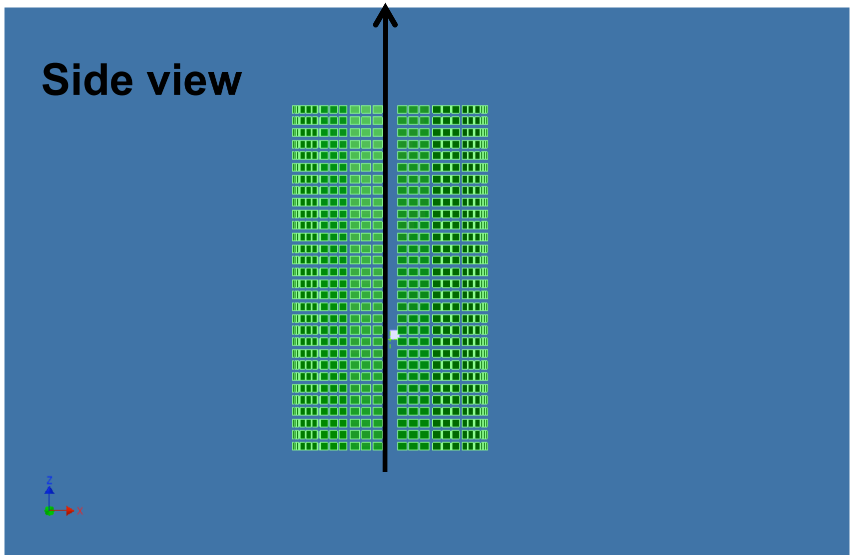
3. Supermodule : (3X30) 90 quad modules on top of a DIRC bar box.

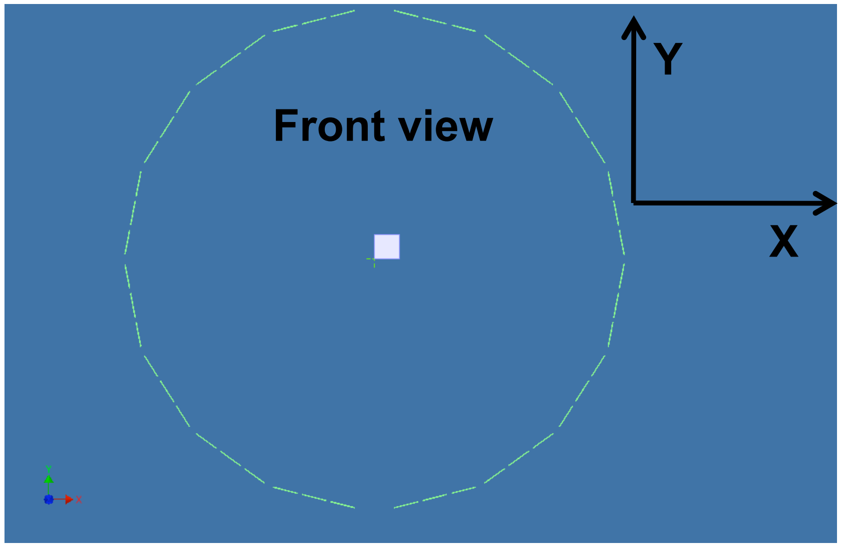
4. Entire SciTil half barrel composed of 8 super-modules



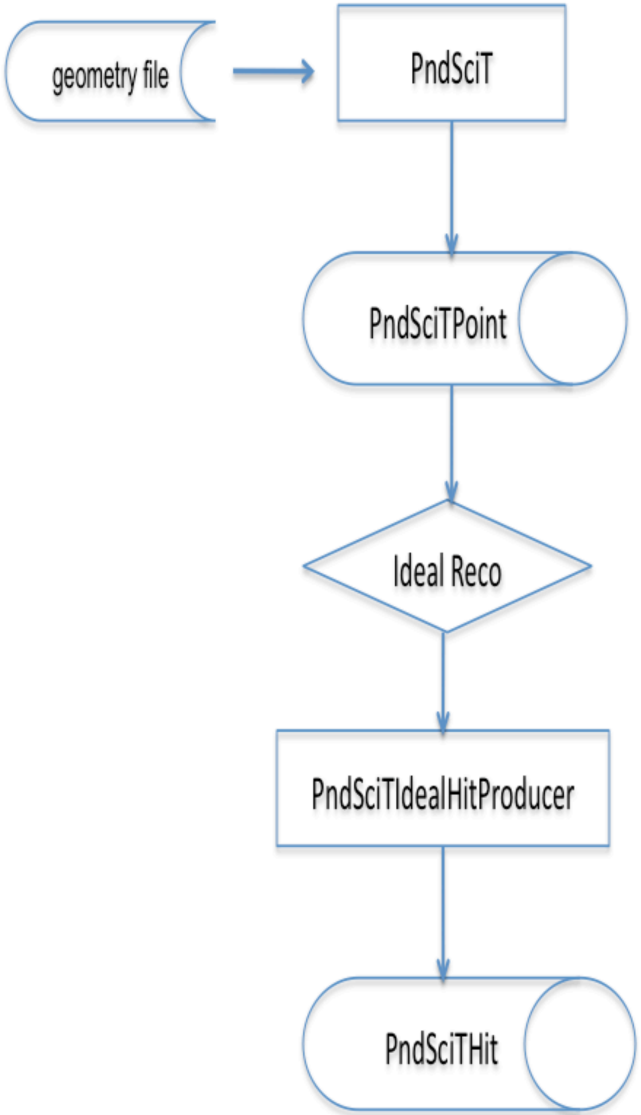
Tech. Design to be converted as Root Geometry







Implementation in the Pandaroot framework



PndSciTPoint

(Int_t trackID, Int_t evtID, Int_t detID,
TString detName, TVector3 posin,
TVector3 momin, TVector3 posout,
TVector3 momout, Double_t tof,
Double_t length,
Double_t eLoss,Double_t charge,
Double_t mass, Int_t pdgCode)

PndSciTHit

(Int_t trackID,
Int_t detID,
TString detName,
Double_t time,Double_t dt,
TVector3& pos,TVector3& dpos,
Int_t index,
Double_t charge);

Ideal reconstruction Hit Collection

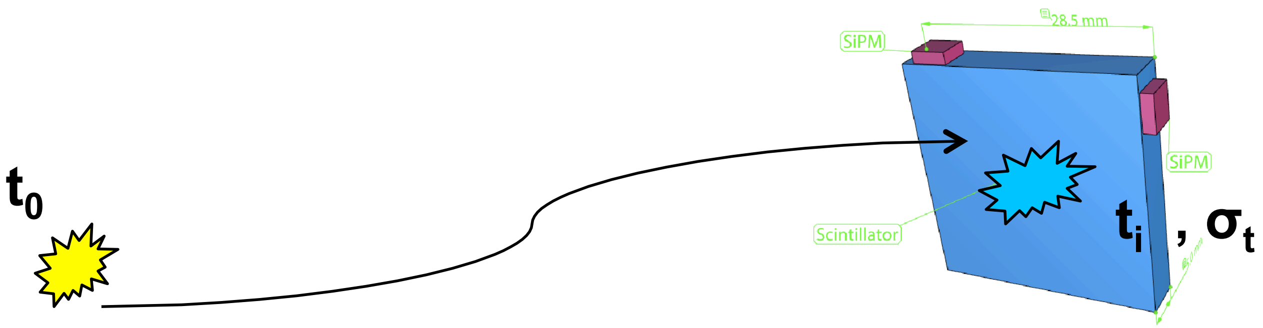
An event takes place at a certain t_o . The generated particles travel for some time with a certain velocity v before they hit a timing detector.

Each particle creates a signal at wall clock time t_i.
Assuming that the track length s can be provided the time of origin $t_{0,i}$ is equal to $t_i - s/v$. Uncertainties can be assumed gaussian, so these values are distributed around event to with a total time resolution σ_t .

<u>PndSciTHit.h</u>

Hit Position corresponds to the center point of each scintillating tile in the submodule

Time is smeared by the expected resolution of the detector ~ 100ps



Implementation on Pandaroot framework

Root Geometry : provided by T. Stockmanns
by using CadConverter program

Material : Polypropylene (defined at media.geo)

Source code : **/trunk/scitil**

- PndSciT.cxx /h
- PndSciTHitProducerIdeal.cxx/h

/trunk/pnddata/SciTData

- PndSciTPoint.cxx /h
- PndSciTHit.cxx/ h

Simulation Macro : **/trunk/macro/scitil**

- simulation macro : *sim_scit.C*
- fast analysis macro : *anaMCOpt.C*
- ideal hit production macro : *hit_scit.C*

SciTil source code : Detector Class

From the root geometry file : /geometry/SciTil_Barrel_woPCB.root

<u>Source code</u> : **/trunk/scitil/**PndSciT.cxx

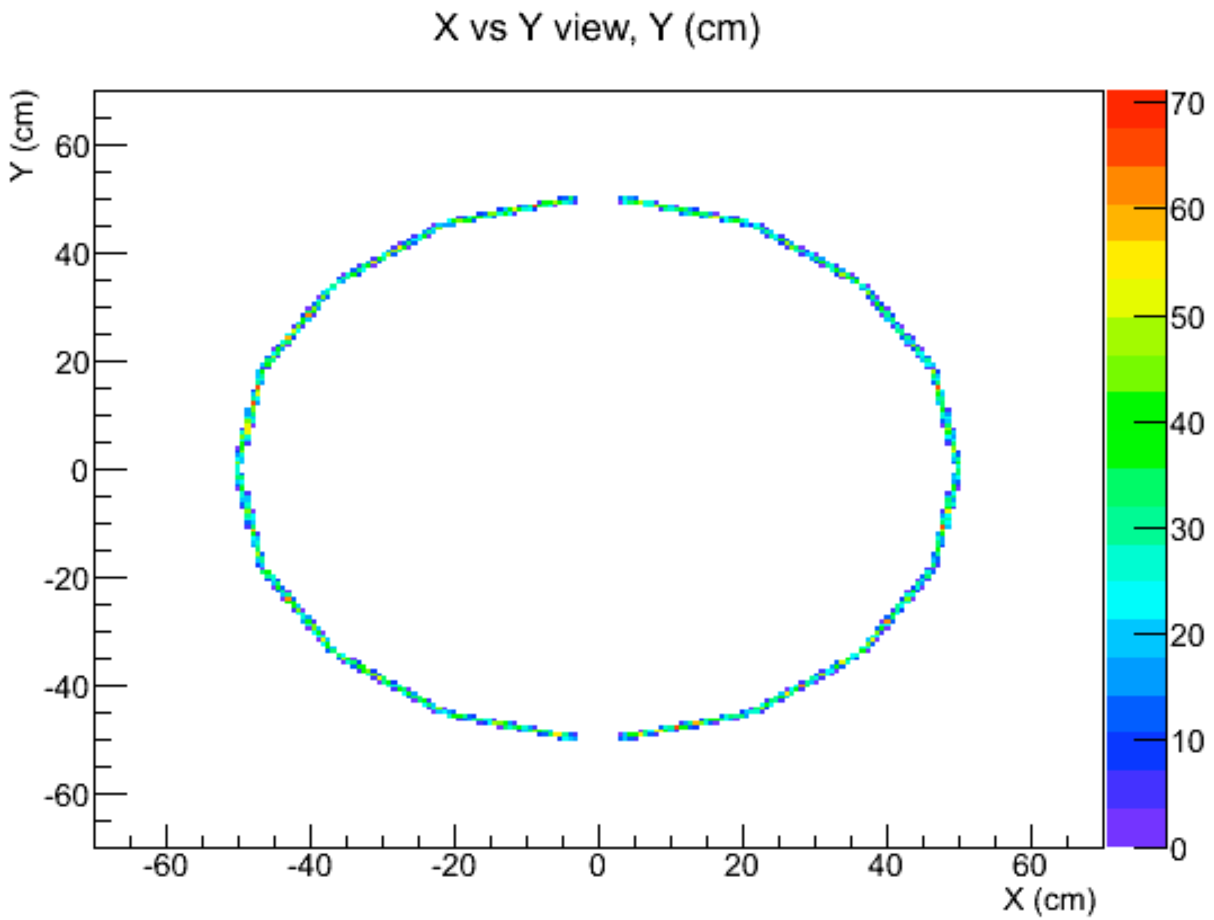
```
// -----   Standard constructor   ----------------------------------------
       PndSciT::PndSciT(const char* name, Bool_t active)
         : FairDetector(name, active) {
          fSciTCollection        = new TClonesArray("PndSciTPoint");
          fVolumeID = -1;
          pvId=0;

          fListOfSensitives.push_back("SciTil");
```

<u>Simulation code</u> : **/trunk/macro/scitil/sim_scit.C**

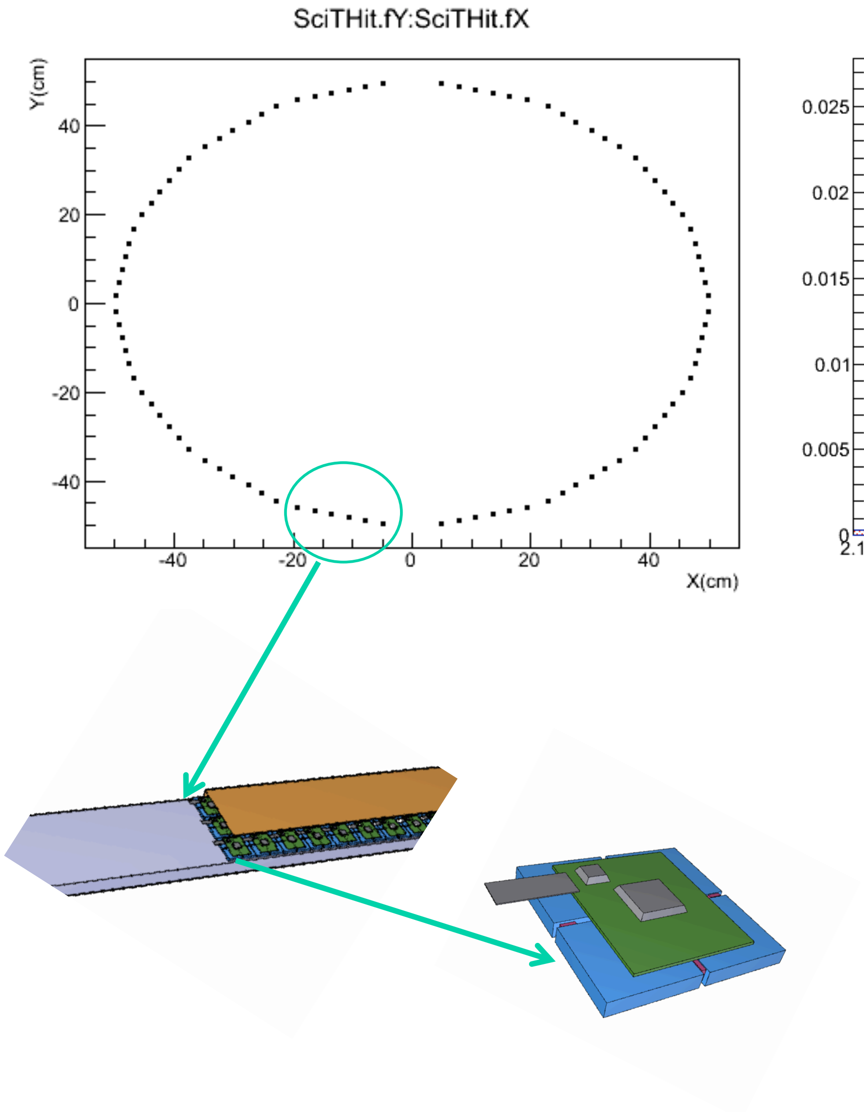
```
  FairDetector *SciT = new PndSciT("SCIT",kTRUE);
   SciT->SetGeometryFileName("SciTil_Barrel_woPCB.root");
   fRun->AddModule(SciT);
```

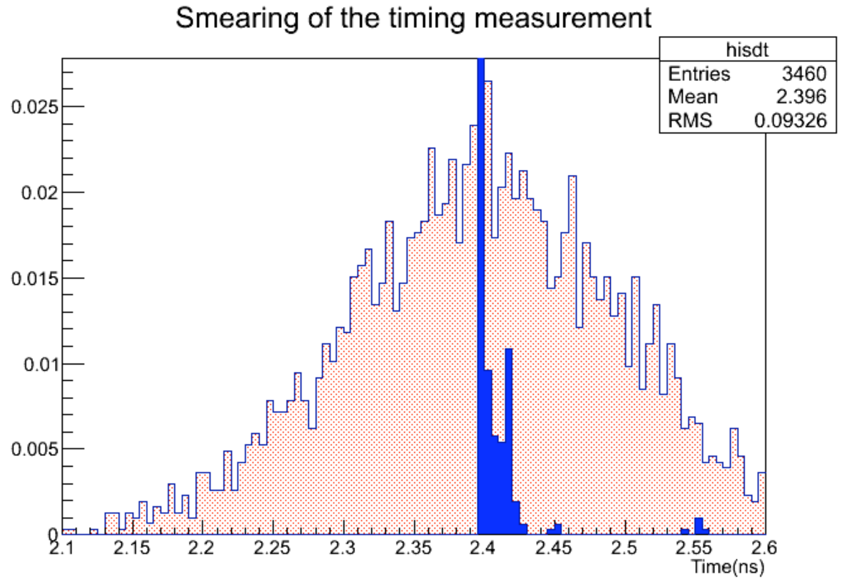
Raw MC information : SciTPoint

UrQMD generator :300 events, mult ~ 10 particles/event



Ideal Hit Information





PndBoxGenerator ~3000 p events

P = 1 GeV/c, theta = 70 ° , phi = 40°

Smearing of MC time t by a TMath::Gauss (t_0 + s/v, σ_t = 100ps)

TODO : Real reconstruction Hit Collection

Light output of scintillator :

Time resolution depends on
number of photons

Signal response of a SiPM
has to be accordingly parametrized.

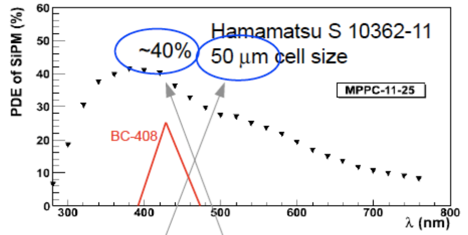
Photon detection
efficiency

≈ 80%   > 80%

PDE = QE * ε * $P_{trigger}$

Geometric factor
100 μm cell size ≈ 90%

Smaller pixel =
more dead area



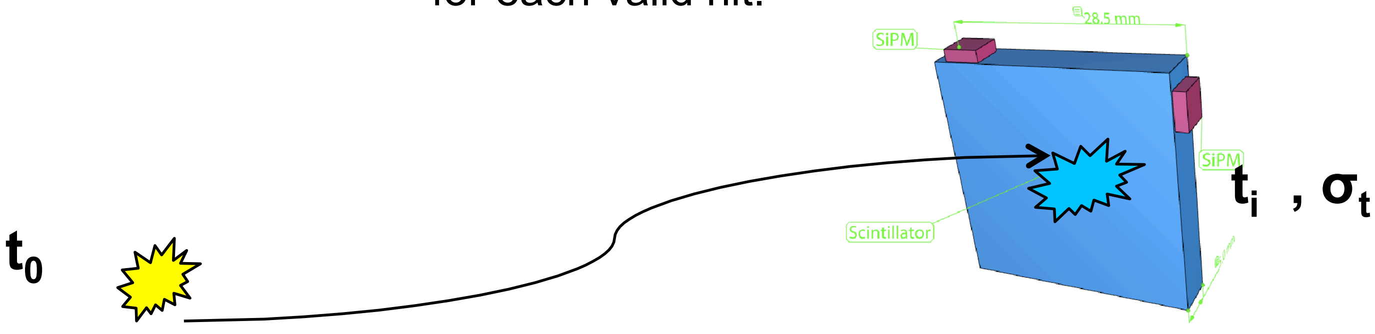
H.Orth, GSI
A. Wilms, GSI
B. Roy, BARC
D. Dutta, BARC

SiPM	Micro cell size μ	Micro cells /mm^2	Bias voltage V	PDE at λ=430 nm	Gain	Dark Rate Mcps at 0.5 p.e.	Signal rise time ns	Material
Hamamatsu S10931-100P	100	100	72	55	$7x10^5$	7	< 1	P on N
Zecotek MAPD3N	8	15000	90	30	$5x10^4$	8	< 2	P on N
KETEK 50D2	50	400	30	40	$5x10^5$	10	< 1	P on N
AdvansID (FBK-irst)	40	400	30	15	$1.5x10^6$	18	< 1	N on P

<u>PndSciTHit.h</u>

Hit Position corresponds to the center point of each scintillating tile in the submodule

Time : precise timestamp (after Amp. and Disc.) is generated for each valid hit.



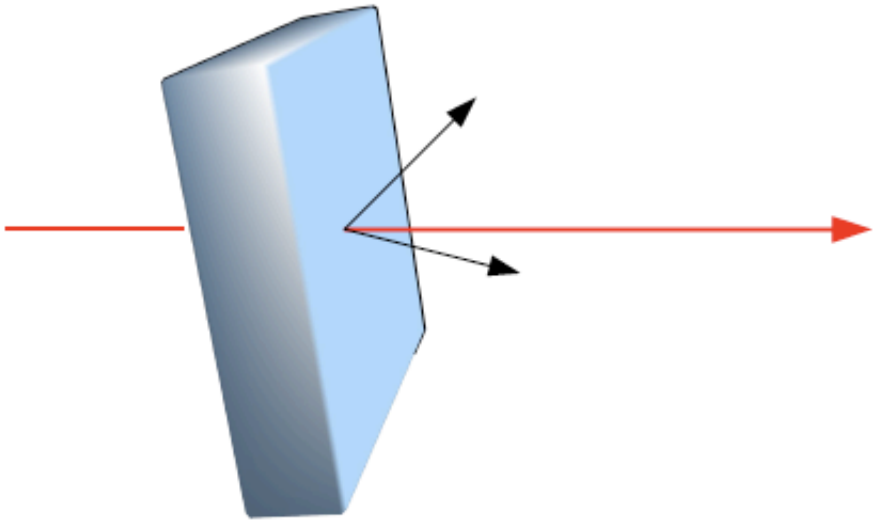
Summary and Outlook

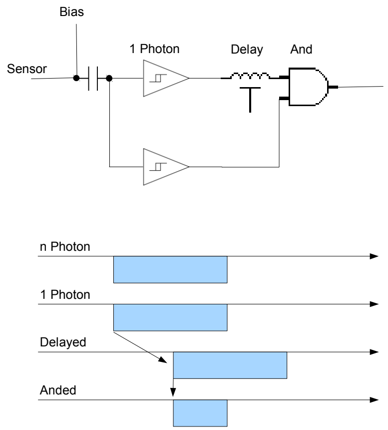
- A simple (only Scintillator tile, no electronic) geometry detector based on the Scintillator tile hodoscope detector proposal has been implemented within the Pandaroot Framework

- A more complex hit reconstruction including R&D is in progress

Photon number

Tile 30 x 30 x 5 mm^3





Minimum ionizing particle

ΔE = 1 MeV
= 10^4 photons — generated

70% hit rim
= 7000 photons — on rim

PD area = 18 mm^2
rim area = 600 mm^2

= 210 photons — geometry

55% PD efficiency — PDE

= 115 photons

30 x 30 x 5 mm^3 → 115 photons