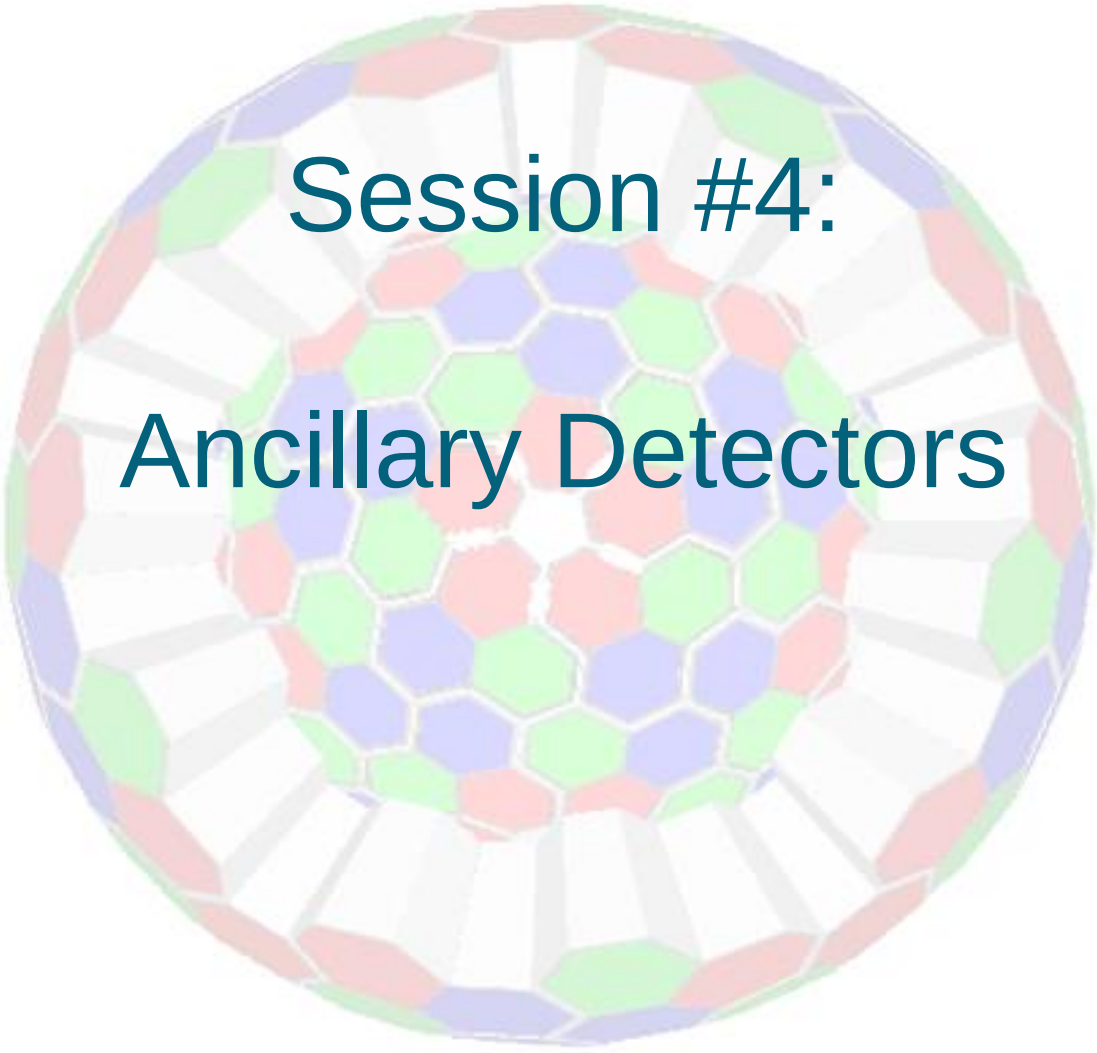
The AGATA Simulation code

Session IV

Marc Labiche
(marc.labiche@stfc.ac.uk)

BCGS Intensive week course,
IIKP Köln- 15th Oct 2015

Science & Technology Facilities Council
Nuclear Physics Group



Session #4:

Ancillary Detectors

Outline

- Session #4: Ancillary Detectors
  - Integration of an ancillary
  - Running the AGATA code with an ancillary
  - Examples: LNL reaction Chamber, PRISMA

Why using ancillaries ?

- Many reaction studies require exclusive measurements
  - For good identification/selection of reaction channel
  - X-section measurements
  - ➢ γ coincidences with light charged particles, Heavy Fragments, and neutrons
- To measure the γ-emitter velocity and direction and perform accurate Doppler corrections.
- To increase efficiency while building the rest of the AGATA array (AGATA+PARIS, +FATIMA, …)

Why adding ancillary to the code ?

- Estimate the energy resolution after Doppler correction
- Simulate γ/particle coincidences
  - Investigate the acceptance effects of the experimental setup and estimate its full efficiency
- Investigate γ-attenuation/scattering in these ancillary detectors and the effect on AGATA performances
  - SPIDER inside reaction chamber
  - Back scattering in ancillaries' surrounding structure
    - Proximity of PRISMA/ VAMOS dipoles

The list of AGATA ancillaries so far

- More than 30 different “ancillaries” already defined
  - Full list displayed with: ./build/Agata -h
  - Most of those expected at LNL (NEDA, …)
  - Some geometries are defined in GEANT4
  - Others are imported from a gdml file.

- Not all are detectors:
  - Reaction chambers
  - VAMOS/PRISMA Dipoles (Brick)

- Not all are completed
  - Current implementation of Mcp/DANTE crashes
  - Geometry maybe define but analysis code is often missing or under development.

The list … so far … (1/2)

Ancillaries	Index	Sensitive Det. instances	Offset
Koeln Si det.	1	1	1000
Shell (default)	2	1	2000
MCP/DANTE	3	1	3000
EUCLIDES	4	2	4000,5000
Brick (PRISMA "dipole")	6	0	6000
N-Wall	7	1	7000
DIAMANT	8	1	8000
EXOGAM	9	1	9000
HELENA	10	1	10000
RFD	11	1	11000
NEDA (gdml)	12	2	12000
Cassandra	16	1	16000

…

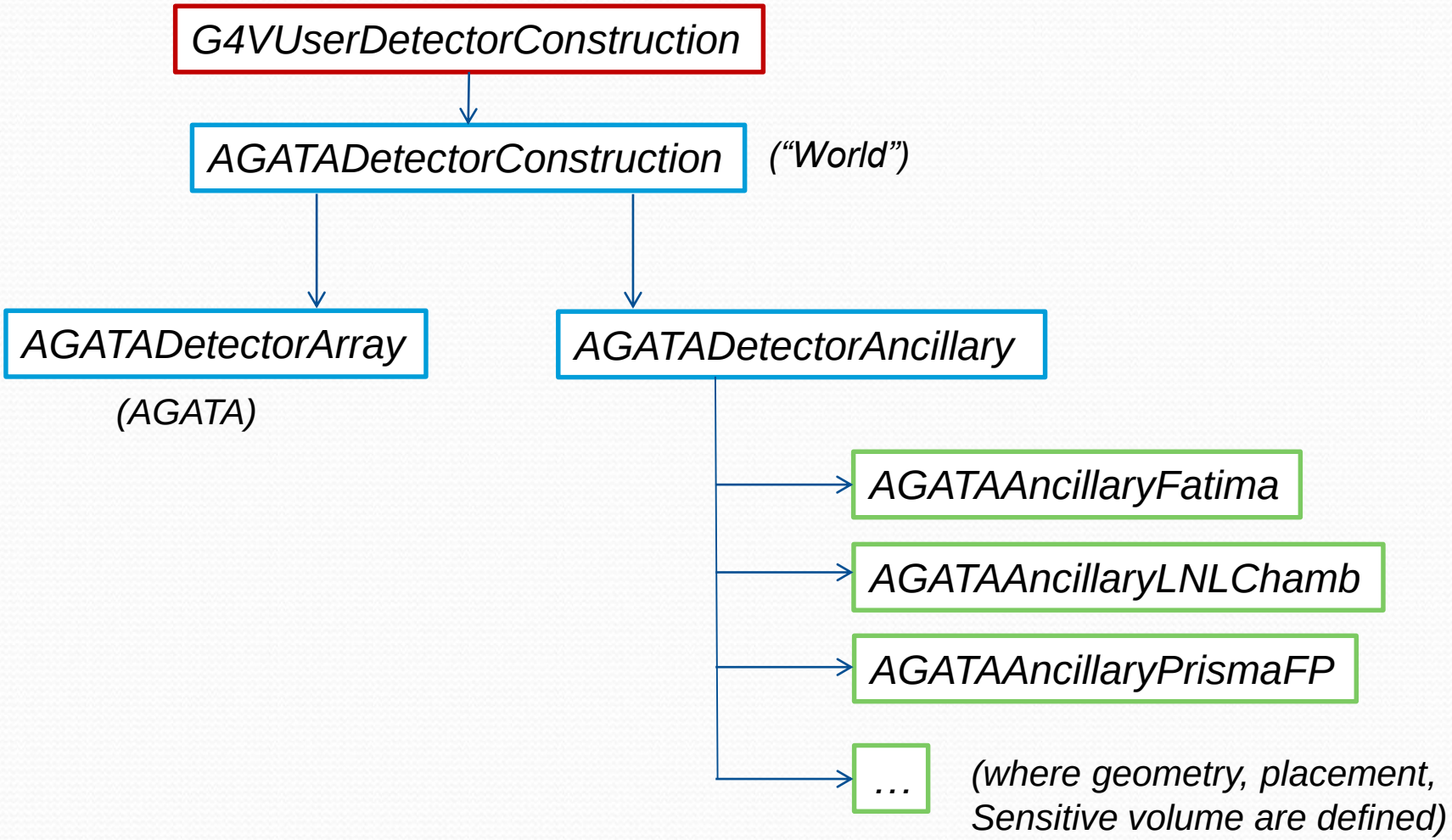
The list … so far … *(2/2)*

Ancillaries	Index	Sensitive Det. instances	Offset
AIDA	17	1	17000
FATIMA	18	1	18000
PARIS	19	1	19000
GSI Chamber	20	n/a	n/a
SPIDER (gdml)	21	1	21000
LYCCA	22	1	22000
NordBallNDet	23	1	23000
Plastic Orsay	24	1	24000
Miniball	25	1	25000
Honeycomb (gdml)	26	n/a	n/a
VAMOS Chamber (gdml)	27	n/a	n/a
OOPS (gdml)	28	n/a	n/a
GALILEO plgr (gdml)	29	n/a	n/a
SIGMA	30	1	30000

The list … so far … *(2/2)*

Ancillaries	Index	Sensitive Det. instances	Offset
DIAMANT plgr (gdml)	31	n/a	n/a
DIAMANT Ftgt (gdml)	32	n/a	n/a
OPSA Orsay	33	1	33000
PRISMA magnets	34	n/a	n/a
PRISMA FP	35	1	tbc
LNL chamber	36	n/a	n/a

Detector construction classes in ASC



How to implement a new one ? (1/5)

Relatively simple:

Use existing example and copy them:

- For ancillary built with G4 native geometry classes

  src/AgataAncillaryFatima .cc & include/AgataAncillaryFatima.hh

- For passive ancillary built from gdml file:

  src/AgataAncillaryLNLChamb.cc & include/AgataAncillaryLNLChamb.hh

- For active ancillary built from gdml file:

  src/AgataAncillarySpider.cc & include/AgataAncillarySpider.hh

How to implement a new one ? (2/5)

- Create your new ancillary detector geometry files: .cc & .hh
  - In the .cc file, 2 parameters ancOffset , ancName which are data members of AgataDetectorAncillary must be initialised:

```
#include "AgataDetectorAncillary.hh"
AgataAncillaryDummy::AgataAncillaryDummy()
{
…
  ancName   = G4String(“DUMMY");
  ancOffset = XX*1000;
}
…
```

How to implement a new one ? (3/5)

Also, for a new detector material:

- Then, in AgataAncillaryDummy.cc , define the FindMaterial method and use a G4Material pointer “mat_csi” to retrieve the material through its name

```
G4Material* FindMaterial (G4String Name){

 // search the material by its name
    G4Material* ptMaterial = G4Material::GetMaterial(Name);

    return ptMaterial;
}


G4int AgataAncillaryFatima::FindMaterials()
{
     mat_csi= FindMaterial(“CsI”);      if (!mat_csi)     return 1;
    return 0;
}
```

How to implement a new one ? (4/5)

Also, for a new detector material:

- Declare and define any new material in: AgataDetectorConstruction.cc

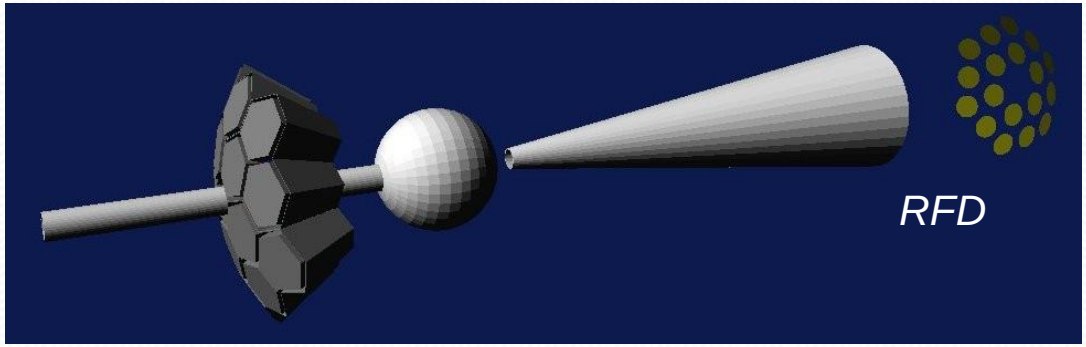
```
void AgataDetectorConstruction::DefineAdditionalMaterial()
{ …
    G4Element* elCs= newG4Element(name="Cesium", symbol="Cs",
                                                    z=55,a=132.90545*g/mole);
    myelement.push_back(elCs);
    G4Element* elI= newG4Element(name="Iodine", symbol="I",
                                                    z=53,a=126.90477*g/mole);
    myElement.push_back(elI);

    G4Material* CsI=newG4Material(name="CsI",density=4.51*g/cm3,Nelements=2);
    CsI->AddElement( elCs , natoms=1);
    CsI->AddElement( elI , natoms=1);
    myMaterial.push_back(CsI);
    …
```

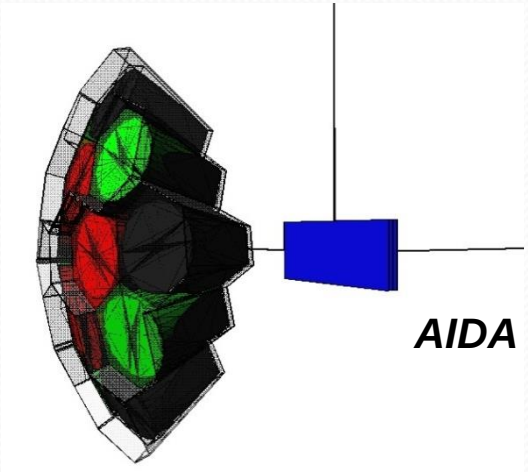
How to implement a new one ? (5/5)

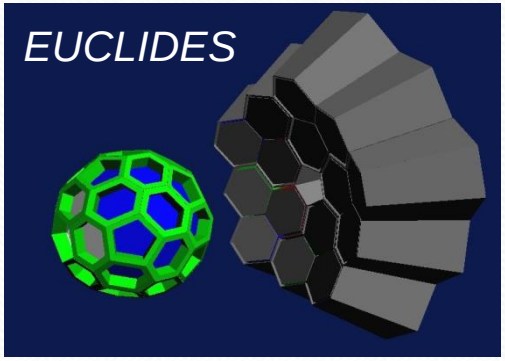
- Instantiate the new ancillary in src/AgataDetectorAncillary.cc:

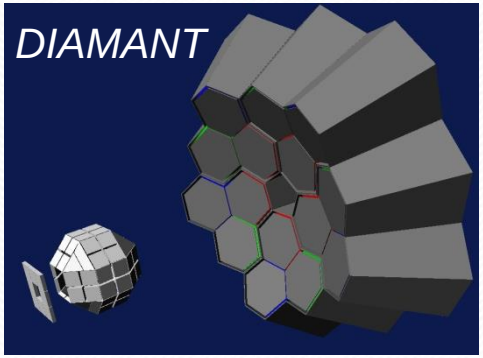
```
#include “AgataAncillaryDummy.hh”
…
AgataAncillaryDummy* theDummy=Null;
…
switch(index)
…
  case XX:
    theDummy = new AgataAncillaryDummy(path,name);
    theAncillary.pushback(AgataAncillaryScheme* theDummy);
    the Constructed.pushback(AgataDetectorConstructed* theDummy)
    break;
    …
```

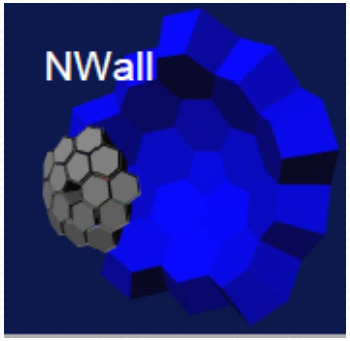
Example of Ancillaries:

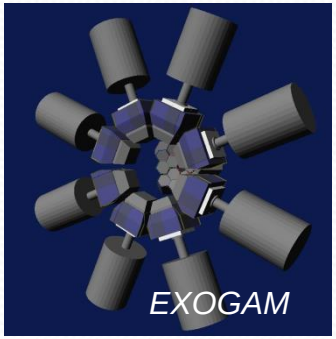


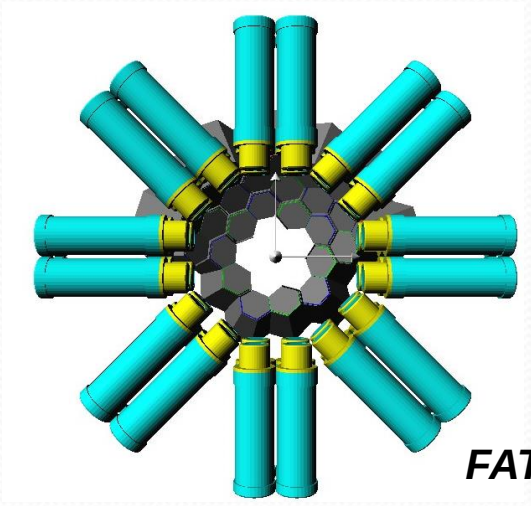




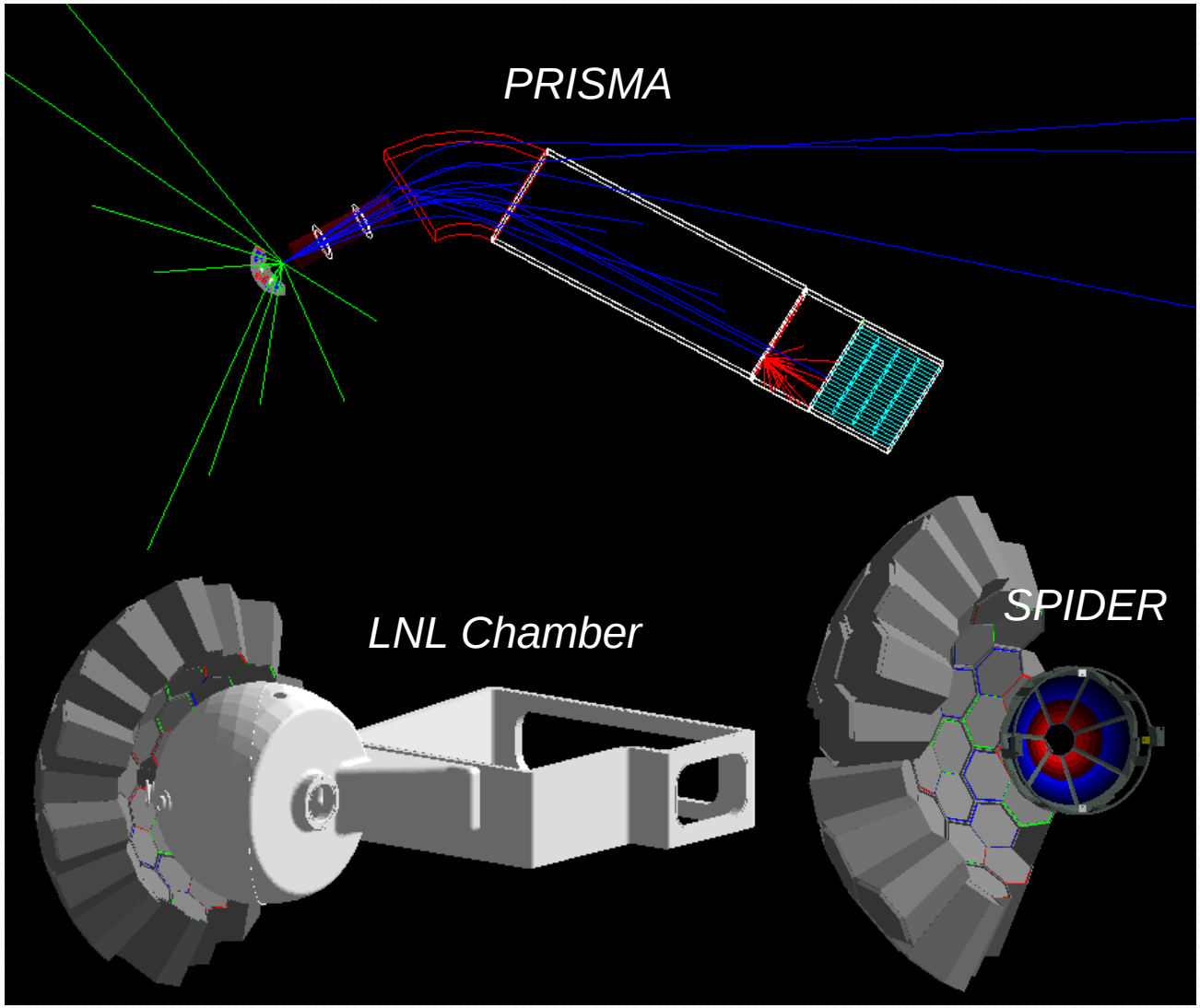








Amongst the latest addition:



Works individual
But yet to be combined together

Requirements for PRISMA ancillary:

- *Need to update the Dipole and quadrupole field map in trunk/field/*
  - *Can be imported from PRISMA simulation code*

- *Produce realistic external event file*
  - *Grazing code*
  - *Including ionic charge distribution*

- *Rely on modified native G4Transport class for ionic charge*

- *Define gas material at right pressure/temperature*

Thus, not so straight forward.

See alternative solution in next session.

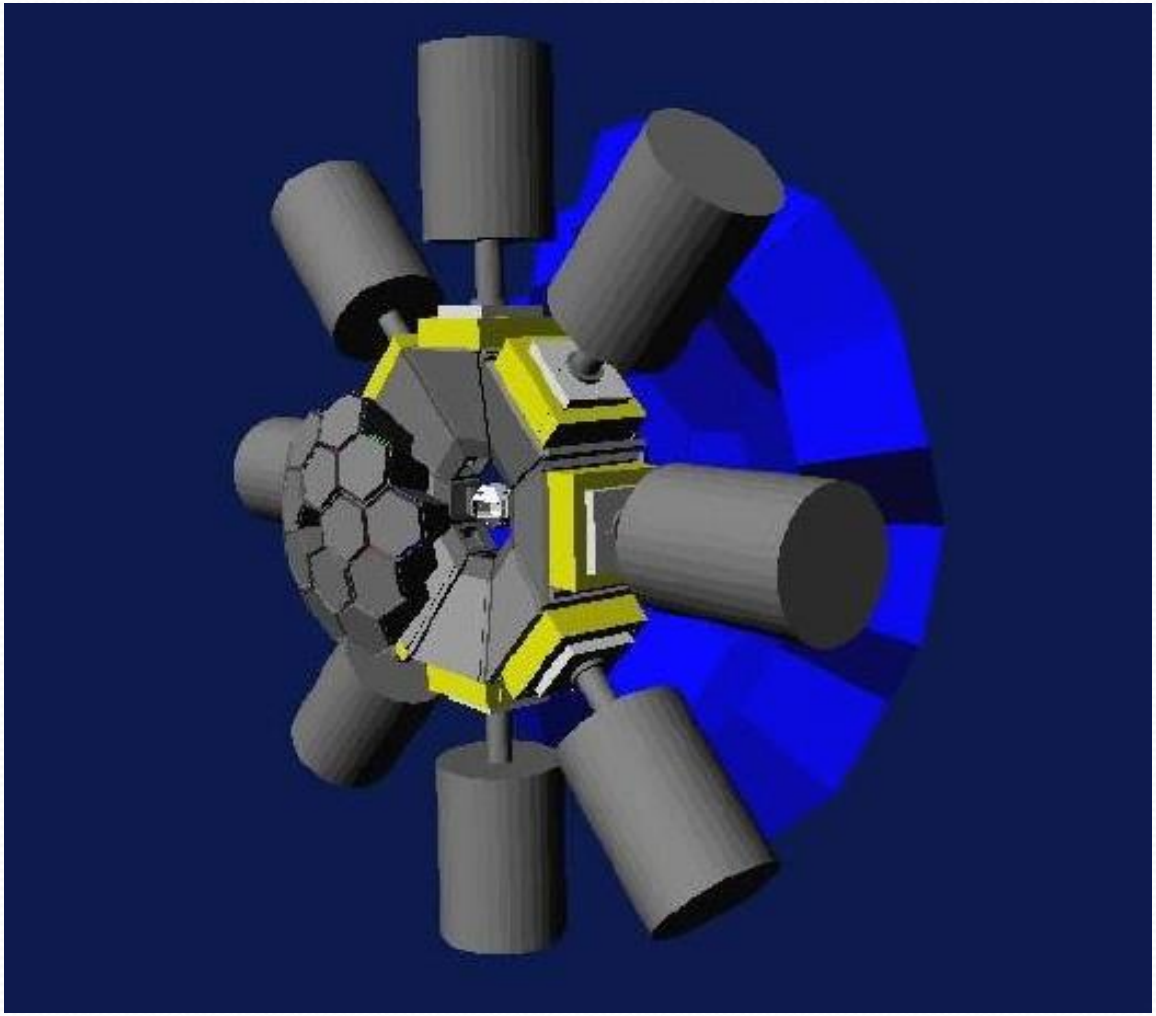
Using ancillaries

Run the command: ./build/Agata –a nAnc Index

>./build/Agata -a 1 9

Idle> /Agata/detector/enableAncillary

Multi-detector system



By running the command:

agata –a 3 7 8 9

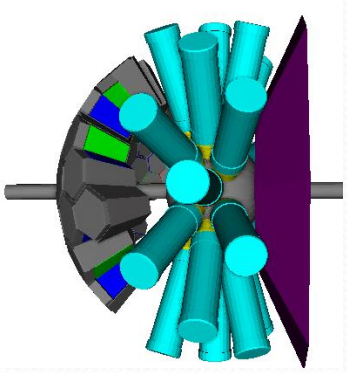
Nb of ancillaries

N-Wall id

DIAMANT id

EXOGAM id

Output file example (with FATIMA)



(fatima_id *1000 + cluster_id*10 + Module_id)

AGATA hit

```
-1   200.000 -0.83292  0.34366  0.43374 0
18070   13.435 -100.648   41.527   52.412 00 0.469
18070  146.896 -100.658   41.541   52.410 00 0.469
18070   39.669 -100.658   41.541   52.409 00 0.469
  -1   200.000  0.04015  0.93466 -0.35326 1
  -1   200.000  0.99356  0.08122  0.07904 2
18051    0.865  125.568   10.265    9.990 00    0.504
18051  128.128  125.565   10.259   10.001 00    0.504
18051   32.929  125.565   10.260   10.002 00    0.504
18051    1.554  125.570   10.263    9.990 00    0.504
18051   13.435  125.463   10.090    9.877 00    0.504
18051   20.024  125.463   10.090    9.877 00    0.504
18051    3.065  125.570   10.263    9.990 00    0.504
 -1   200.000 -0.15023 -0.01530 -0.98853 3
  6   11.067  -35.545  -3.619 -233.896 04  1.042
  6  124.508  -35.524  -3.623 -233.898 04  1.042
  6   64.425  -35.527  -3.625 -233.899 04  1.042
```

Incident γ ray, Lab energy, dx,dy,dz Evt number

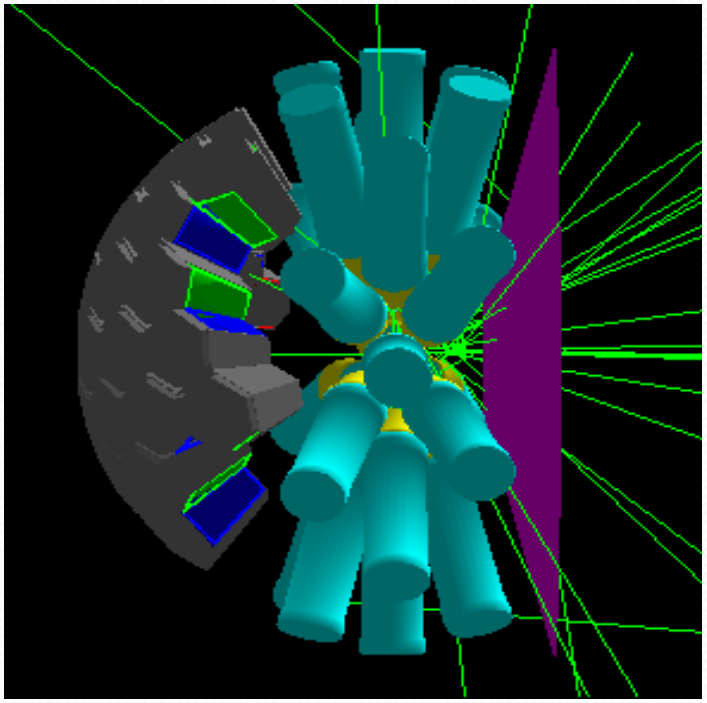
Energy Loss

Lab Position in mm

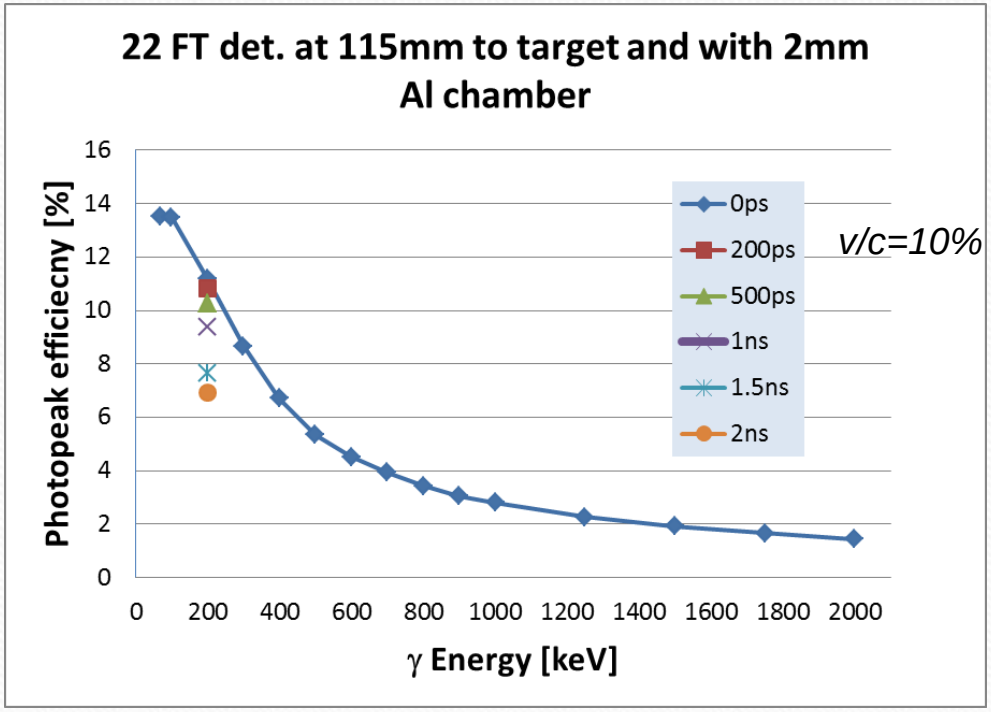
Detection Time in ns

*nSlice*10+nSector*

Ex: Lifetime effect on FATIMA photopeak efficiency



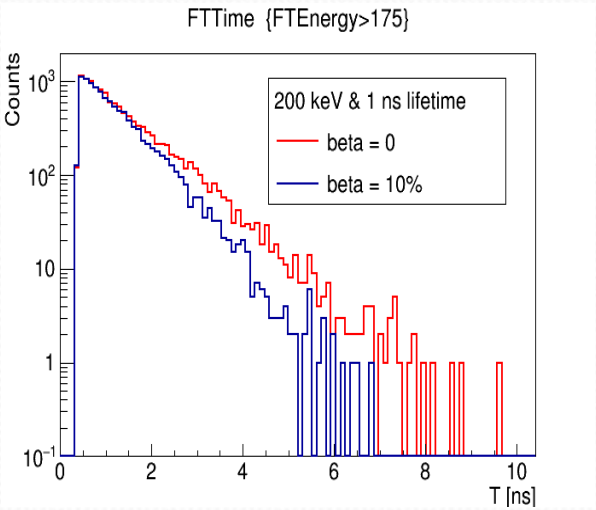
Note: Chamber present but not shown

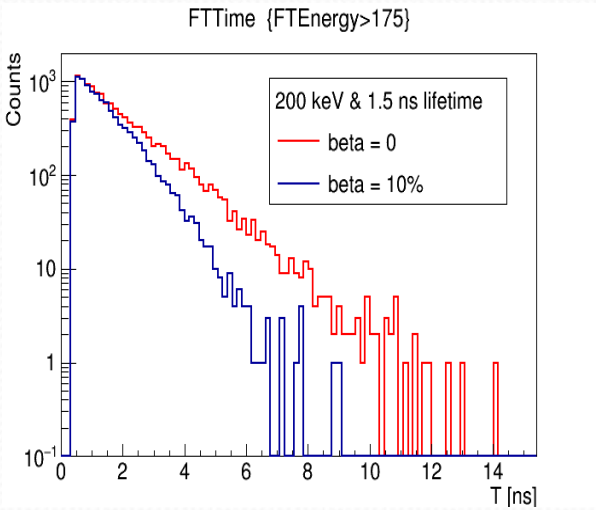


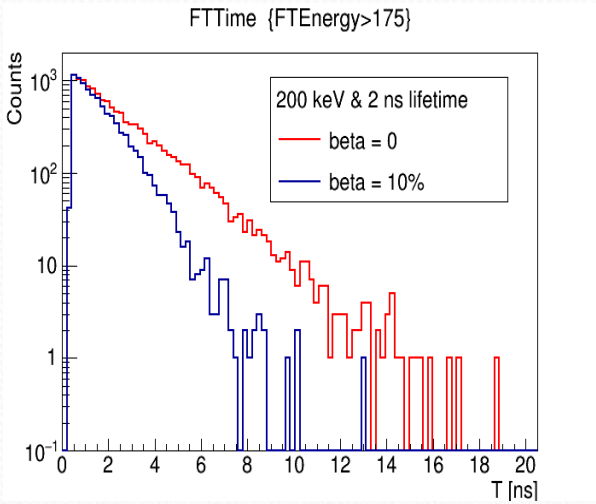
(Note: FATIMA only)

Using: /Agata/generator/gamma/tau x (ns)

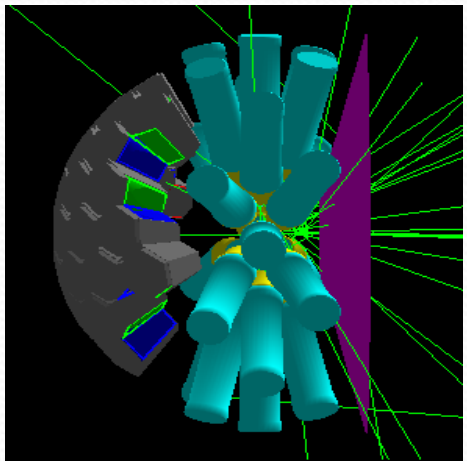
Ex: Simulated decay curves in FATIMA





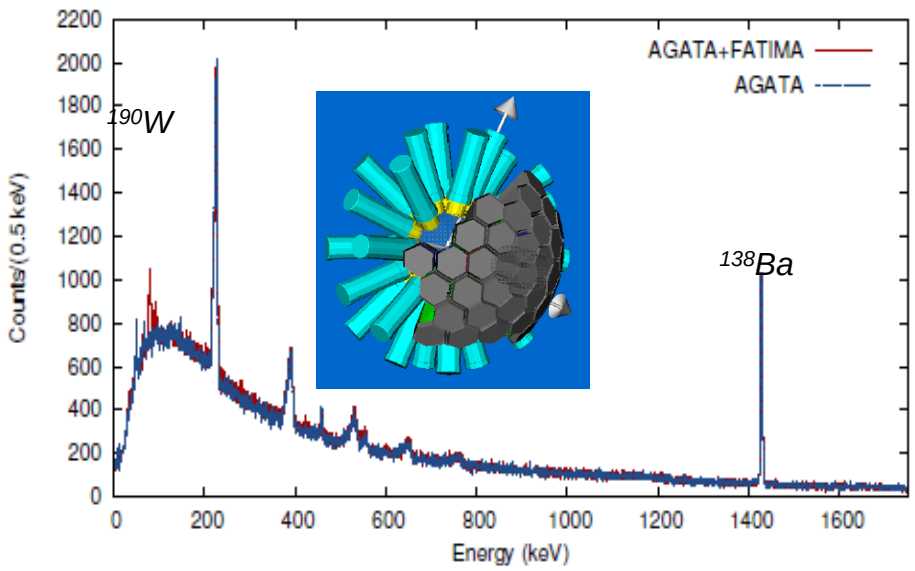


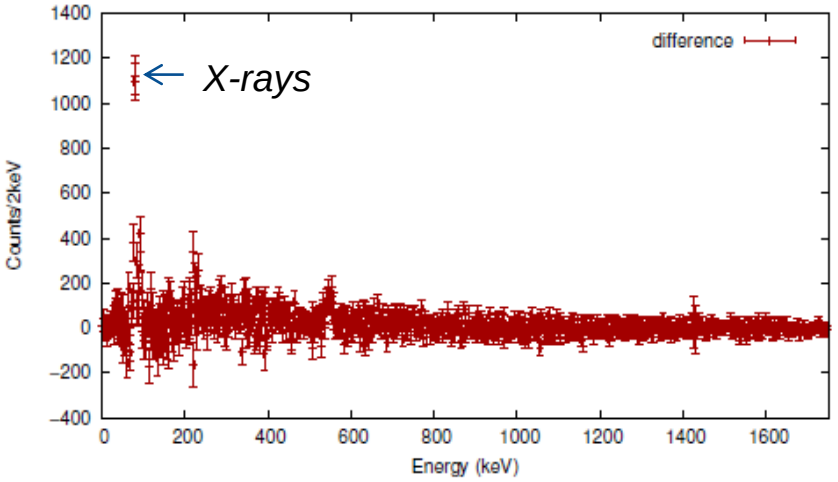
Absorption effect on the decay curve when source not at rest



Ex: Effect of Fatima on AGATA tracking ?

Courtesy of Philipp John & Alain Goasduff



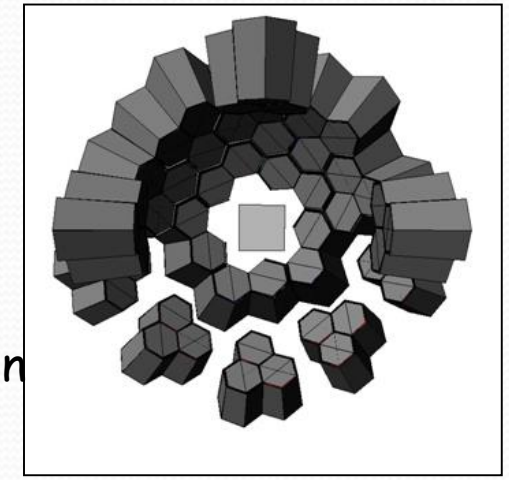


+2p channel in $^{136}Xe+^{192}Os$ at 900MeV

Doppler corrected using ^{138}Ba recoil,

^{138}Ba γ rays: 1435.8 and 462.8 keV,
^{190}W γ rays: 207, 357, 485, 591, 695 keV,
+ 20 γ rays for low energy background

Other “ancillaries”

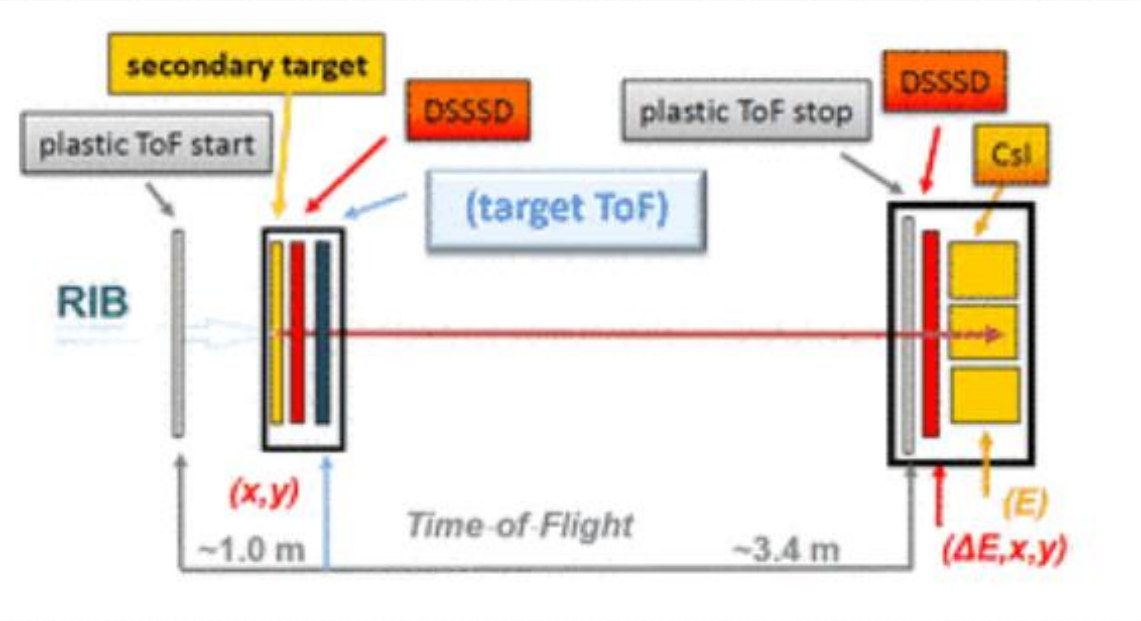


- EUROBALL/MINIBALL
  - Not really defined like the other ancillaries
    - Close to AGATA design
    - Just different crystal/cluster size & segmentation
  - Thus → modified AGATA crystal/cluster
  - Use mgt tracking code to analyse the resulting data

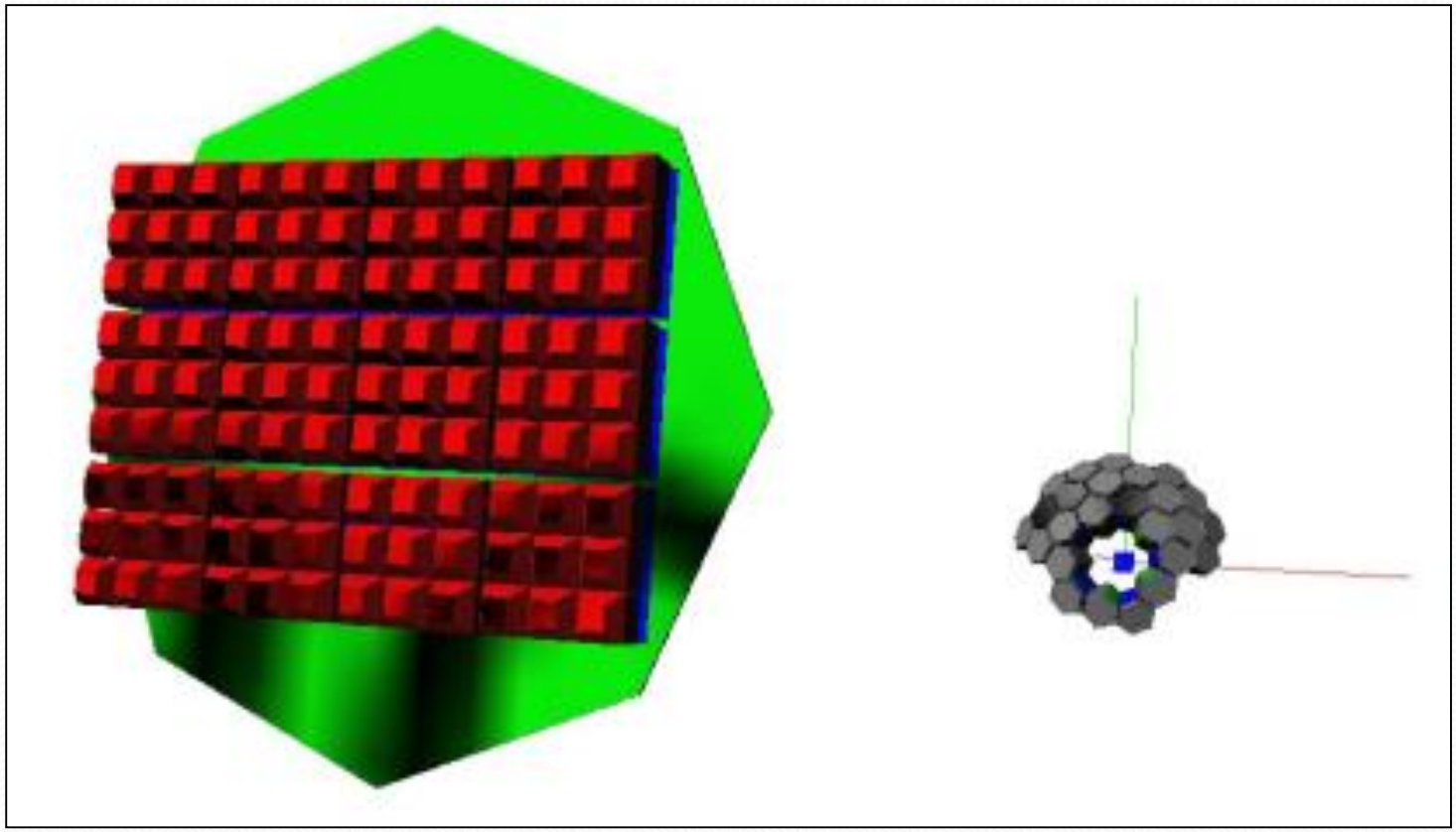
- PRISMA
  - Importing Field map from PRISMA
  - Digitization of FP detector to be completed
    - Coincidences with acceptance effects
    - Doppler correction

The ancillary LYCCA-0

- Courtesy of D. Bloor & P. Joshi (York University)
- LYCCA-0 for prespec experiment at GSI.



Lycca-0 in AGATA code



End of presentation #4

Summary

Overview of the AGATA simulation code

- “old” GEANT4 program and could be improved in many ways
- Many ancillaries are available to users
- Many tools are also still under developments
- New users are welcome !
  - Please report bugs or development you need.
- New developers are even more than welcome !

Tutorials:

Ex 1:

With the command: ./build/Agata –a nAnc Index

Set GDMLPATH to your gdml/ folder

>./build/Agata -a 1 21

/control/execute macros_LNL/agata_OGL_anc.mac

Ex 2:

With the command: ./build/Agata –a nAnc Index

>./build/Agata -a 1 36

/control/execute macros_LNL/agata_OGL_anc.mac

Ex3: Simulation with FATIMA

>./build/Agata -a 1 18

Idle> /Agata/file/enableLM

Idle> /Agata/file/info/outputMask 11100110

Idle> control/execute macros/agata8T4D_Anc_OGL.mac

Idle> /Agata/generator/gamma/energy 200.

Idle> /Agata/generator/gamma/tau 2

Idle> /Agata/run/beamOn 10000

A new GammaEvent.0000 should be created !

> Inspect GammaEvents.0000 and check the time is recorded.

Then, exit QT and go to analysis/fatima and run FatimaRead as described in README.

Open a root session and inspect the time spectrum in the root output file FatimaTree.root. Is the lifetime 2ns ?

Ex4: Same as Ex3 with beta=10%

>./build/Agata -a 1 18

Idle> /Agata/file/enableLM

Idle> /Agata/file/info/outputMask 11100110

Idle> control/execute macros/agata8T4D_Anc_OGL.mac

Idle> /Agata/generator/gamma/energy 200.

Idle> /Agata/generator/recoil/beta 10

Idle> /Agata/generator/gamma/tau 2

Idle> /Agata/run/beamOn 10000

A new GammaEvent.0000 should be created !

> Inspect GammaEvents.0000 and check the time is recorded.

Then, exit QT and go to analysis/fatima and run FatimaRead as described in README.

Open a root session and inspect the time spectrum in the root output file FatimaTree.root. Compare time spectra with the one produced in Ex3.

Ex4: with PRISMA

Edit macros_LNL/agata_OGL_anc_Ext.mac

Check the external event file is set for events/90Zr_ionicCharge.evt:

Agata/generator/emitter/eventFile events/90Zr_ionicCharge.evt

Set G4PROTONHPDATA variable to point to the Geant4 data folder G4TENDL1.3.2/Proton

Set G4AGATAVACUUMINWORLD to 1

> ./build/Agata –n -Ext -a 2 34 35

/control/execute macros_LNL/agata_OGL_anc_Ext.mac