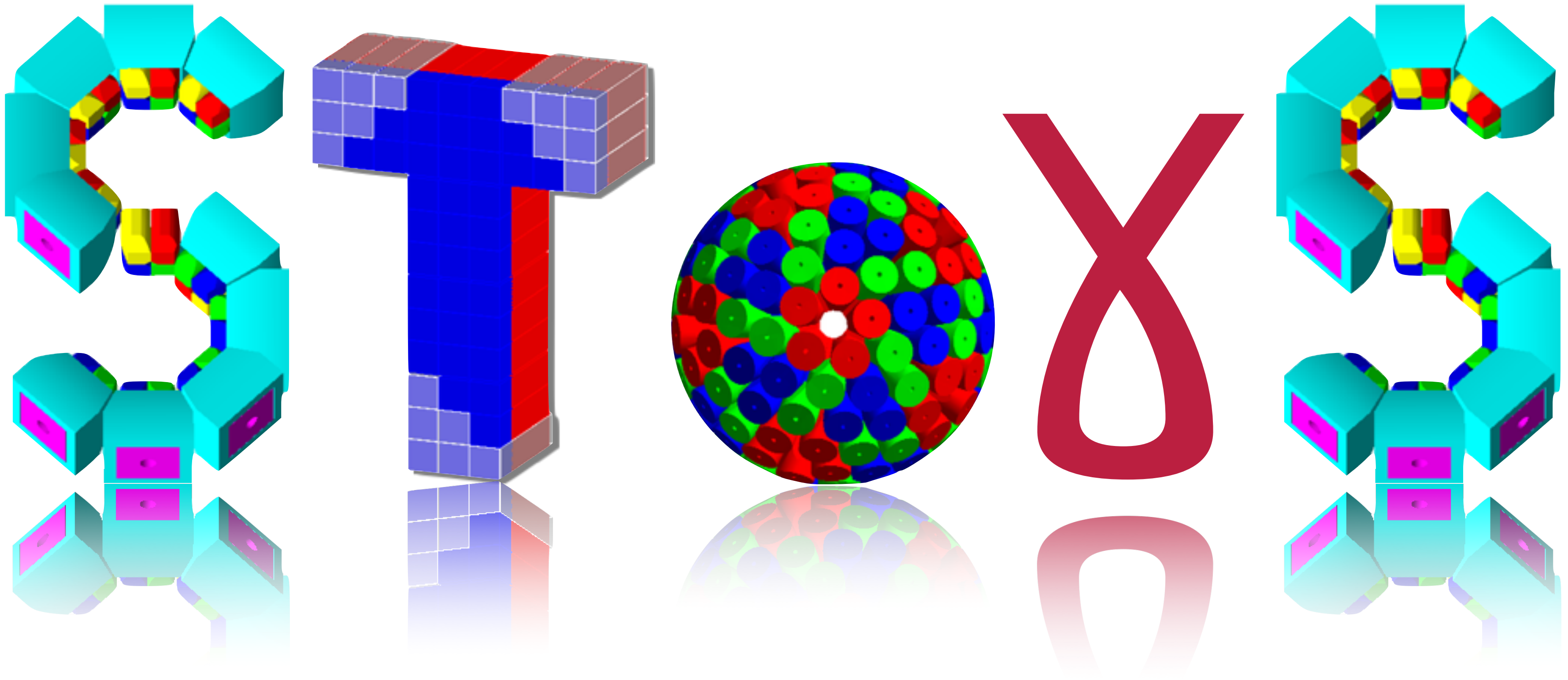
THE STOGS TOOLKIT



© I.Companis, C. Ducoin, O. Stézowski

UCBL Lyon 1

LIO Institut des Origines de Lyon
UNIVERSITE DE LYON

UNIVERSITÉ DE LYON

ipnl

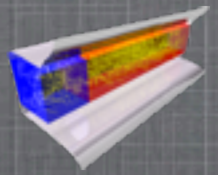
Workshop Simulation-ICC 2015 - GANIL 23-25/11/2015

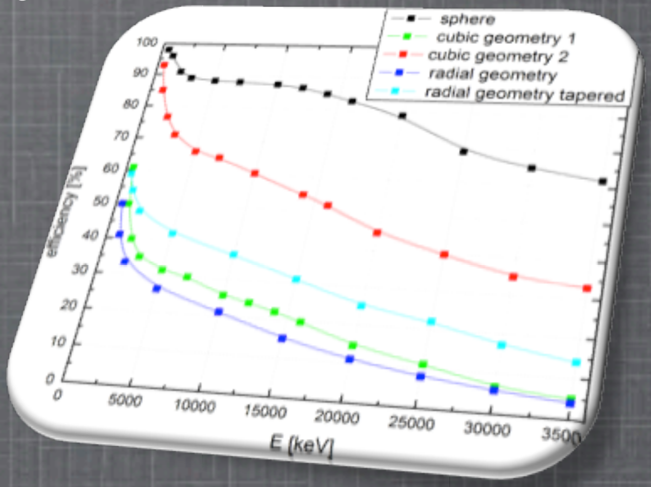
STоGS: What is it and for what purpose ??

Simulation Toolkit fOr Gamma-ray Spectroscopy
OR Simulation Toolkit fOr Ganil Spiral

It inherits from the PARIS package

- G4 developed to design PARIS i.e. two layers of scintillators
  - gamma-ray absorptions
  - scintillations: light creation / collection





Some Goals:

- Should keep some good PARIS features [R&D]
- More oriented to simulations of experiments
- Toward full simulation chains ‘à la Particle Physics’
  - i.e. work on realistic generators
  - i.e. work on complex setup
  - i.e. work on complex analysis
  - i.e. modification of physics in GEANT4

How to build is a GEANT4 application ???

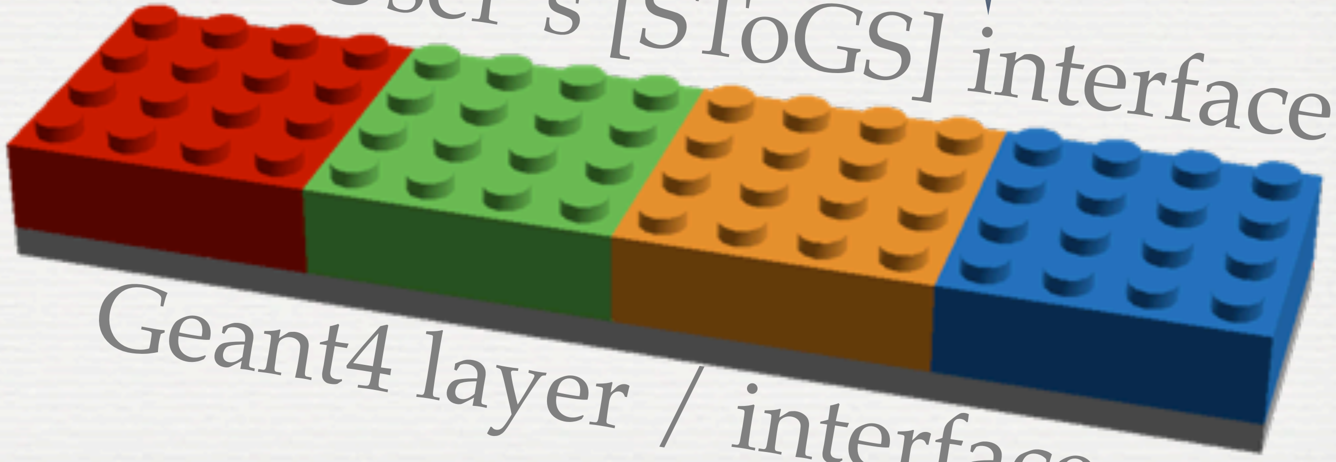
3 mandatory bricks and one almost mandatory

the detector construction - the description of the physics - the primary generator

- the user’s actions

Ingredients of a G4 application

It might be a good idea to NOT break too much the G4 model



The User’s [SToGS] interface

Geant4 layer / interface

How to build is a GEANT4 application ???

3 mandatory bricks and one almost mandatory

the detector construction - the description of the physics - the primary generator - the user's actions

STоGS provides many of such bricks

some are pure G4 - could be copied as such in other packages [files starting with SToGS_G4]

some relies on an additional SToGS layers [files starting with SToGS_ only]

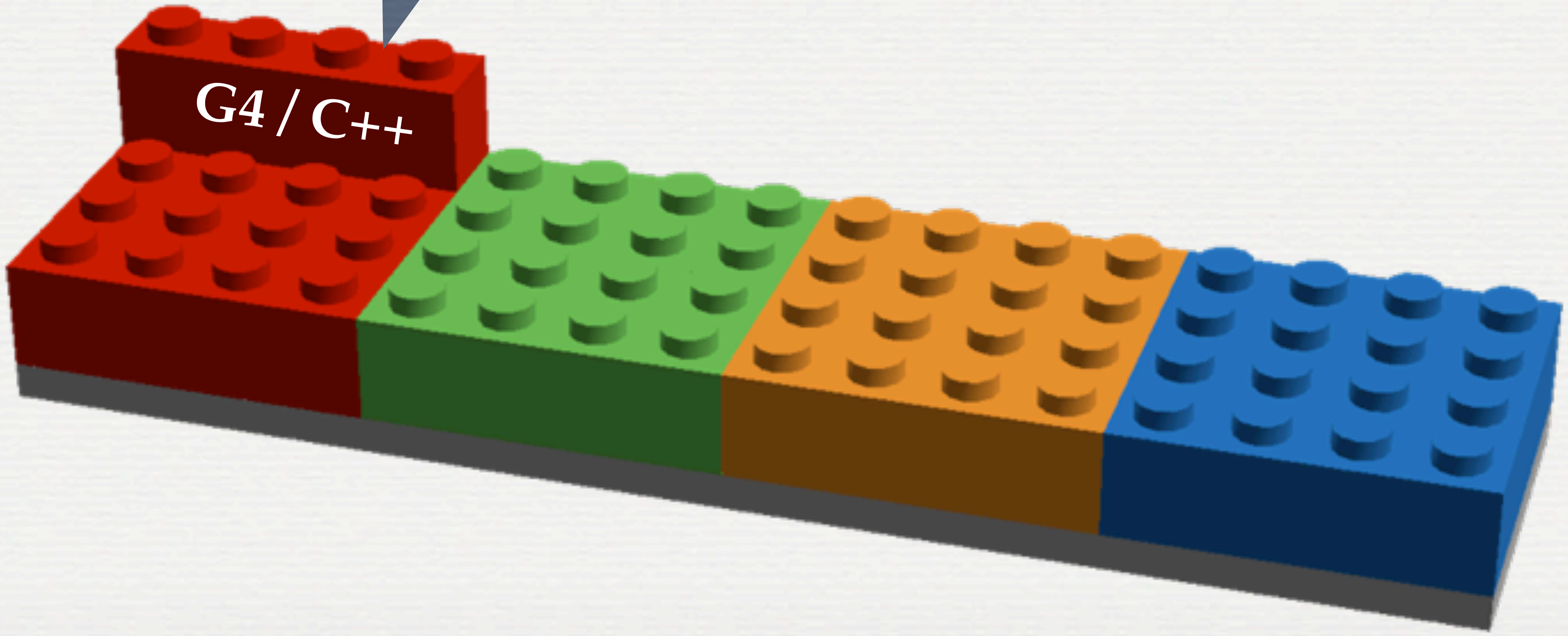
Main SToGS bricks described in the following slides ....

How to deal with **geometries** in SToGS ?

Standard Geant4

✔

the geometry is described
by using
C++ code
G4VUserDetectorConstruction

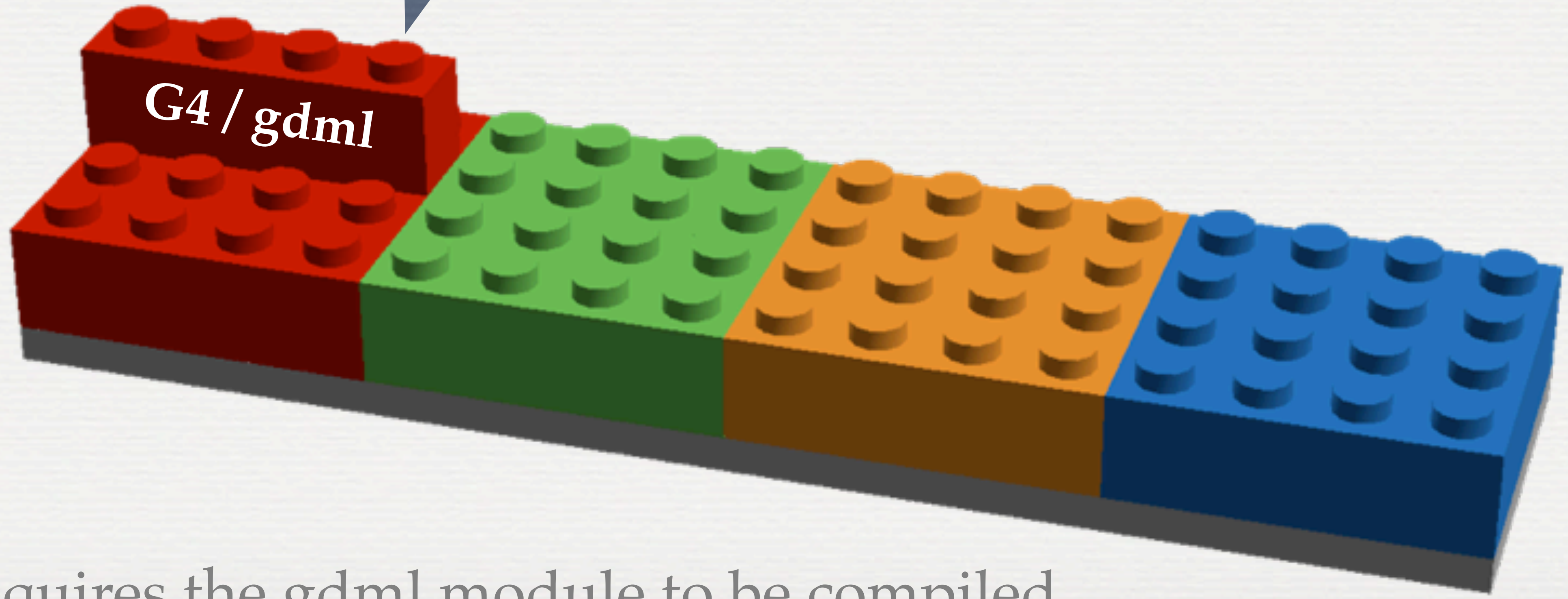


How to deal with **geometries** in SToGS ?

Standard Geant4

✔

the geometry is described
by using
an ascii file
with a xml/gdml syntax

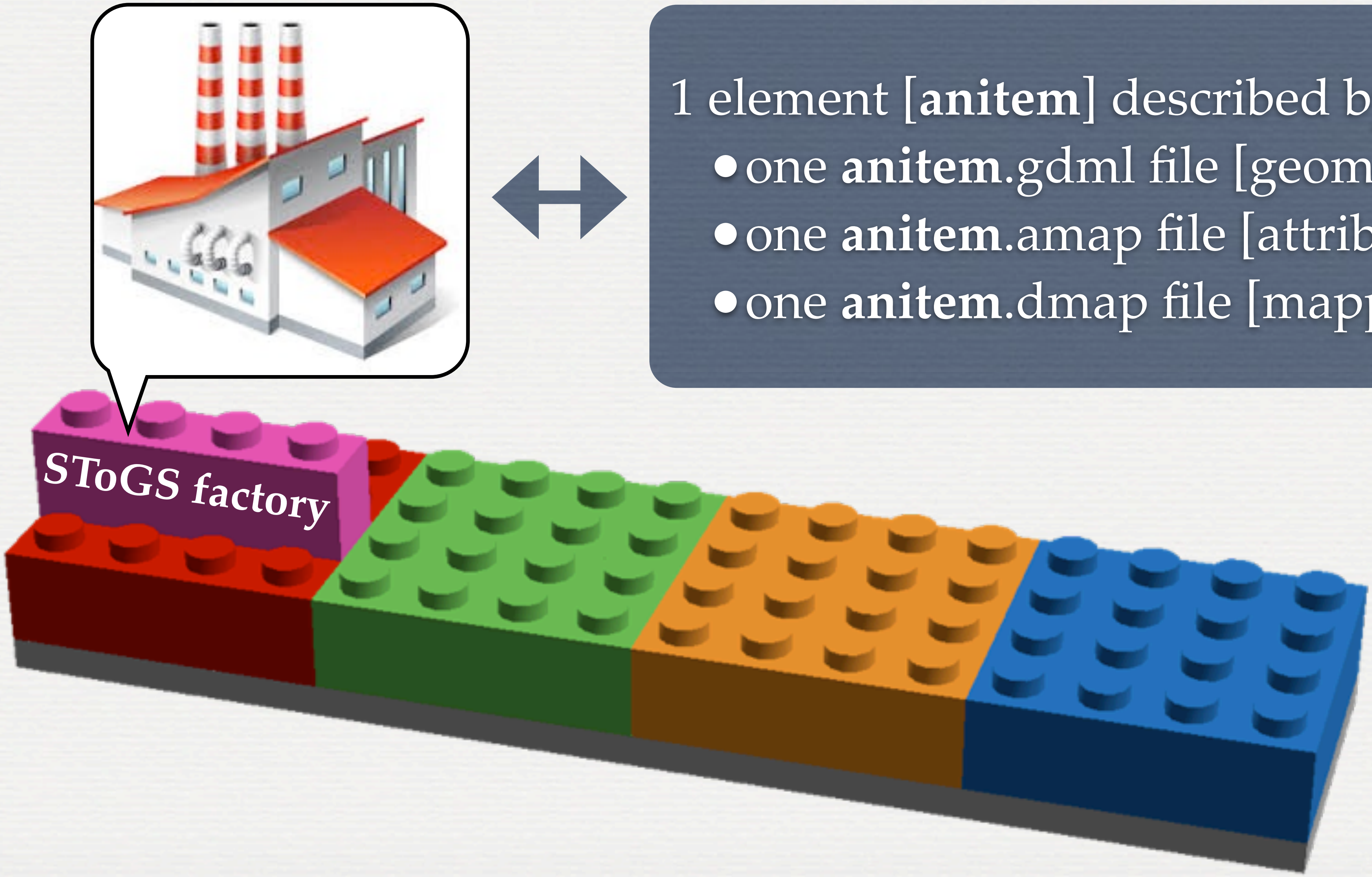


NOTE: it requires the gdml module to be compiled

How to deal with **geometries** in SToGS ?

STOGS mechanism

based on the idea of factories delivering standard detectors



1 element [**anitem**] described by:

- one **anitem**.gdml file [geometry]
- one **anitem**.amap file [attributs]
- one **anitem**.dmap file [mapping]

How to deal with **geometries** in SToGS ?

STOGS mechanism

based on the idea of factories delivering standard detectors

1 element [**anitem**] described by:

SToGS already provides several standard detectors

Organized in groups. For instance, one factory for Scintillators

The goal is to extend the system [ICC common goal]

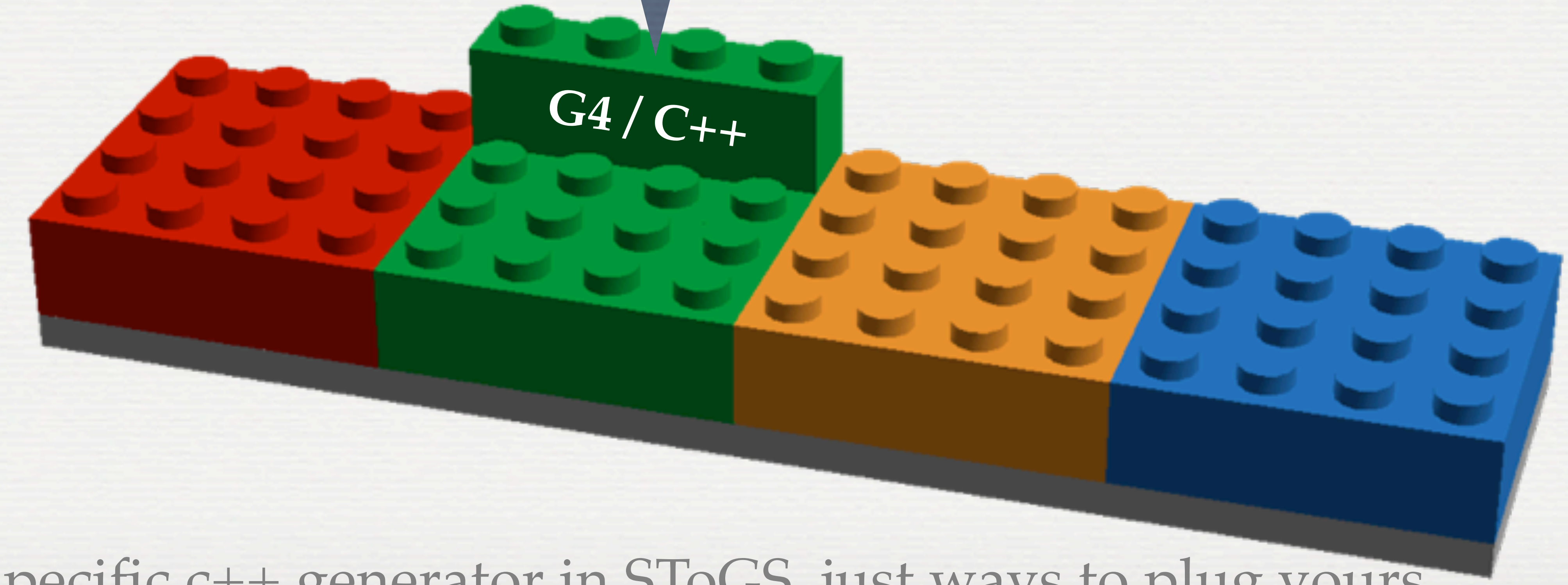
Main SToGS detectors described in the following slides ....

How to deal with **Primaries** in SToGS ?

Standard Geant4

✔

Primary particles are generated by C++ code G4VUserPrimaryGeneratorAction



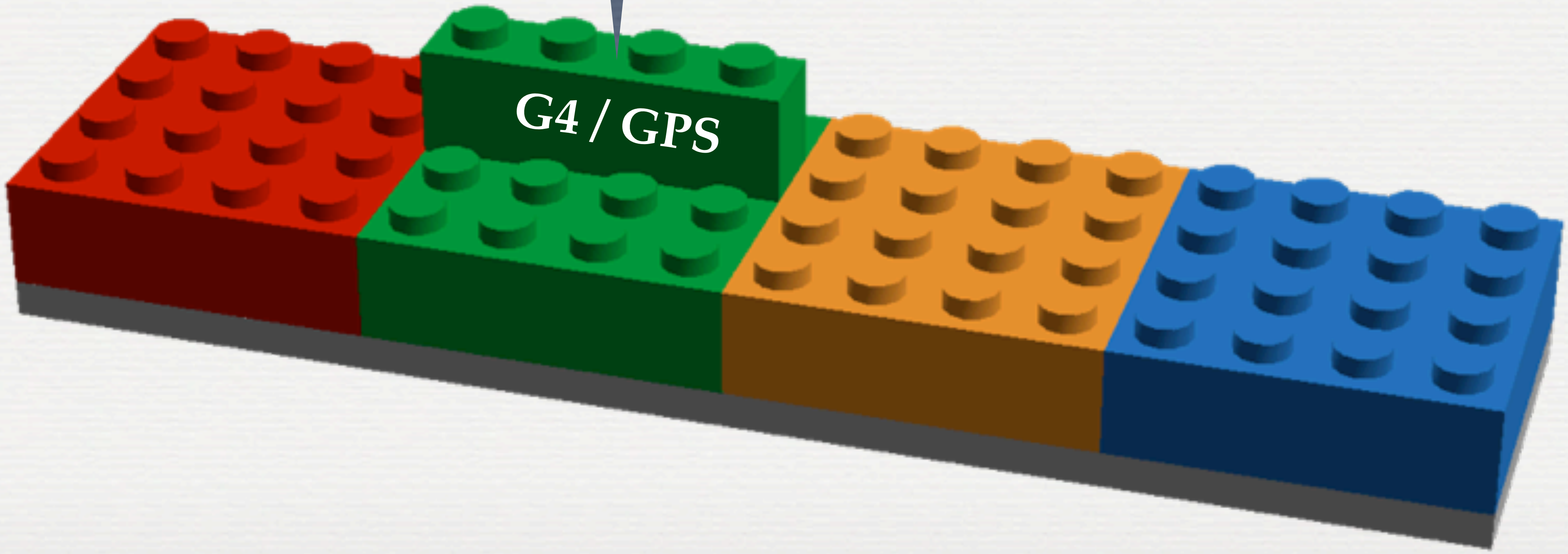
Note: no specific c++ generator in SToGS, just ways to plug yours

How to deal with **Primaries** in SToGS ?

Standard Geant4

✔

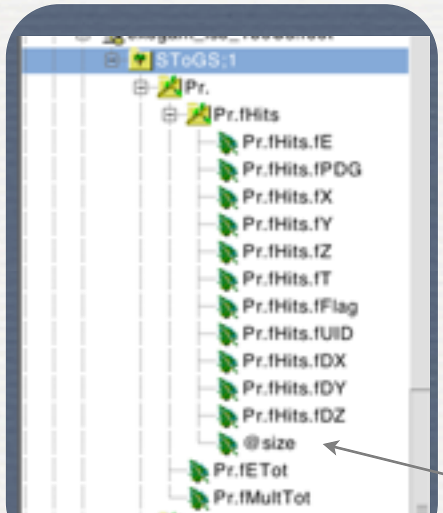
Primary particles are generated by ascii code [G4 macros]



How to deal with **Primaries** in SToGS ?

STOGS reads primaries in external files:

Primaries stored in ROOT Tree in a specific format



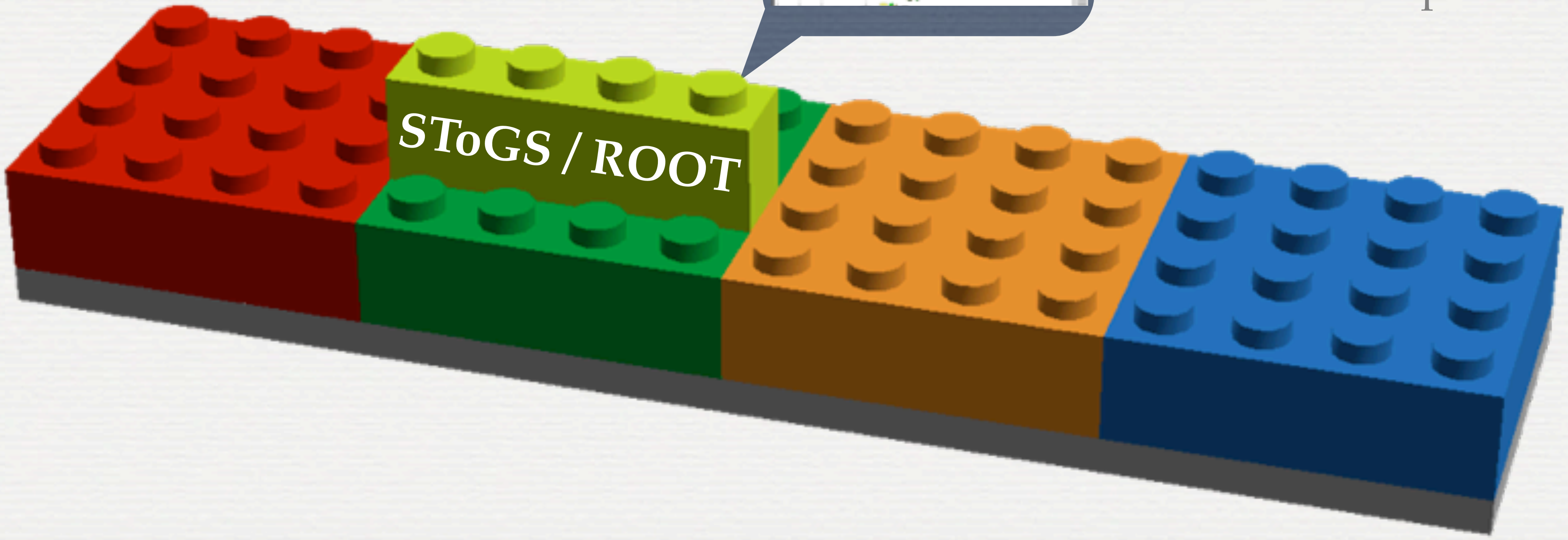
```
[illegible]
  SToGS;1
    Pr.
      Pr.fHits
        Pr.fHits.fE
        Pr.fHits.fPDG
        Pr.fHits.fX
        Pr.fHits.fY
        Pr.fHits.fZ
        Pr.fHits.fT
        Pr.fHits.fFlag
        Pr.fHits.fUID
        Pr.fHits.fDX
        Pr.fHits.fDY
        Pr.fHits.fDZ
        @size
      Pr.fETot
      Pr.fMultTot
```

Energies

PDG codes

ARRAYS } positions 4D
Quality flags
Unique IDs
Directions

# of emitted particles



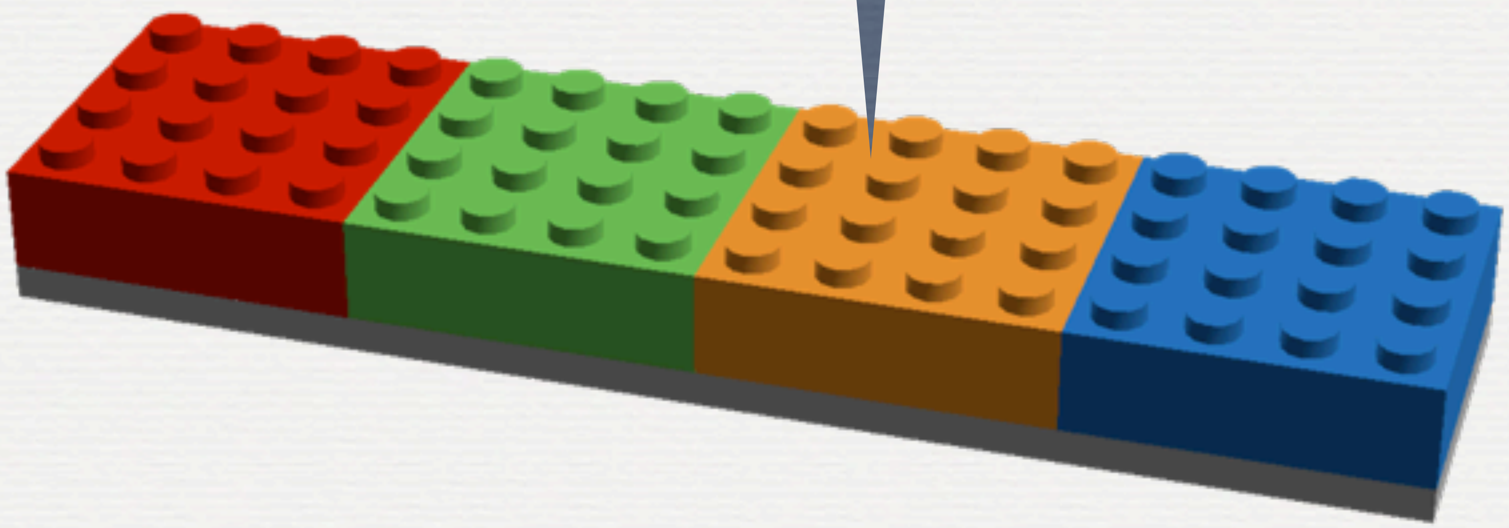
Note: there are no reading of external files in ascii format ...

How to deal with **Physics lists** in SToGS ?

Standard Geant4

✔

So far SToGS provides way to select (ascii configuration file) standard G4 physics lists

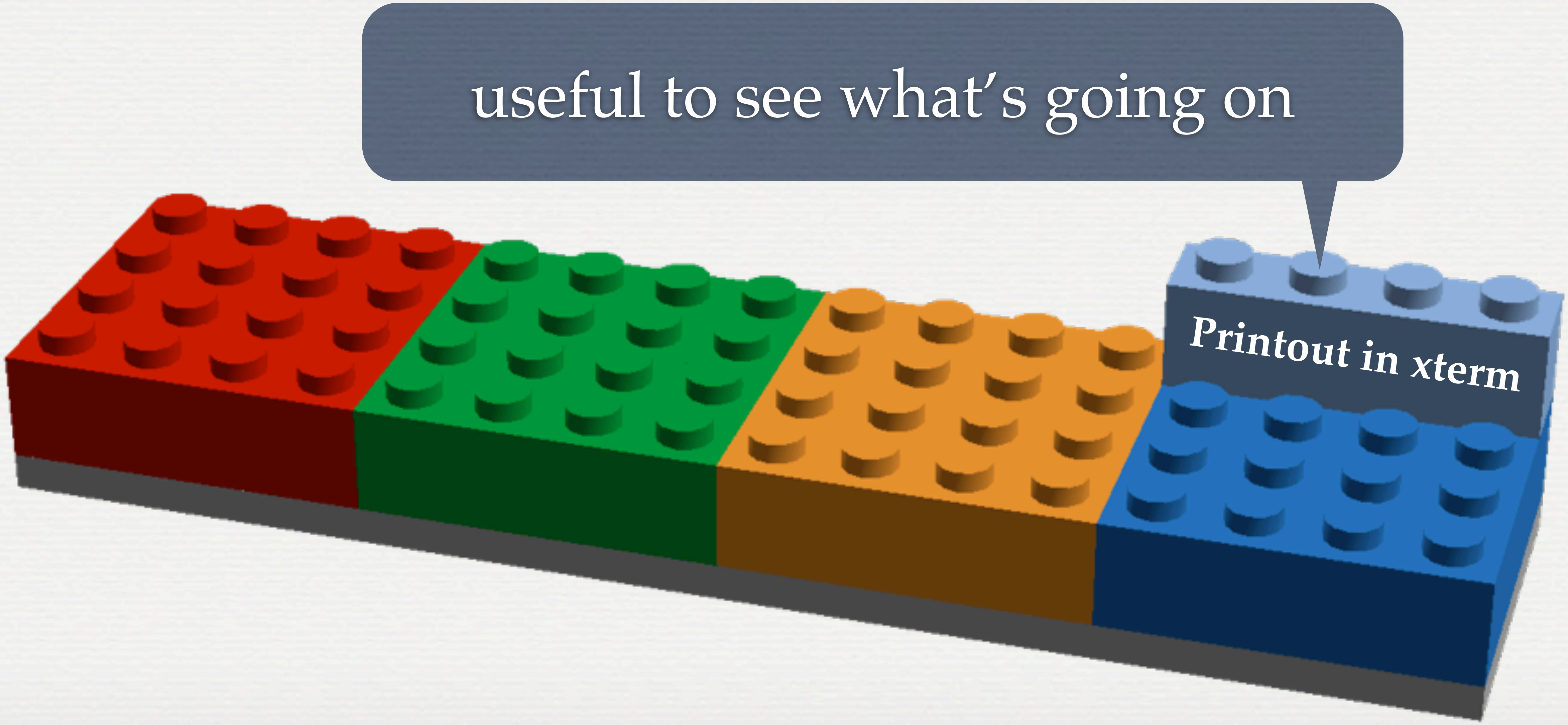


Note: it is foreseen to investigate the physics in GEANT4

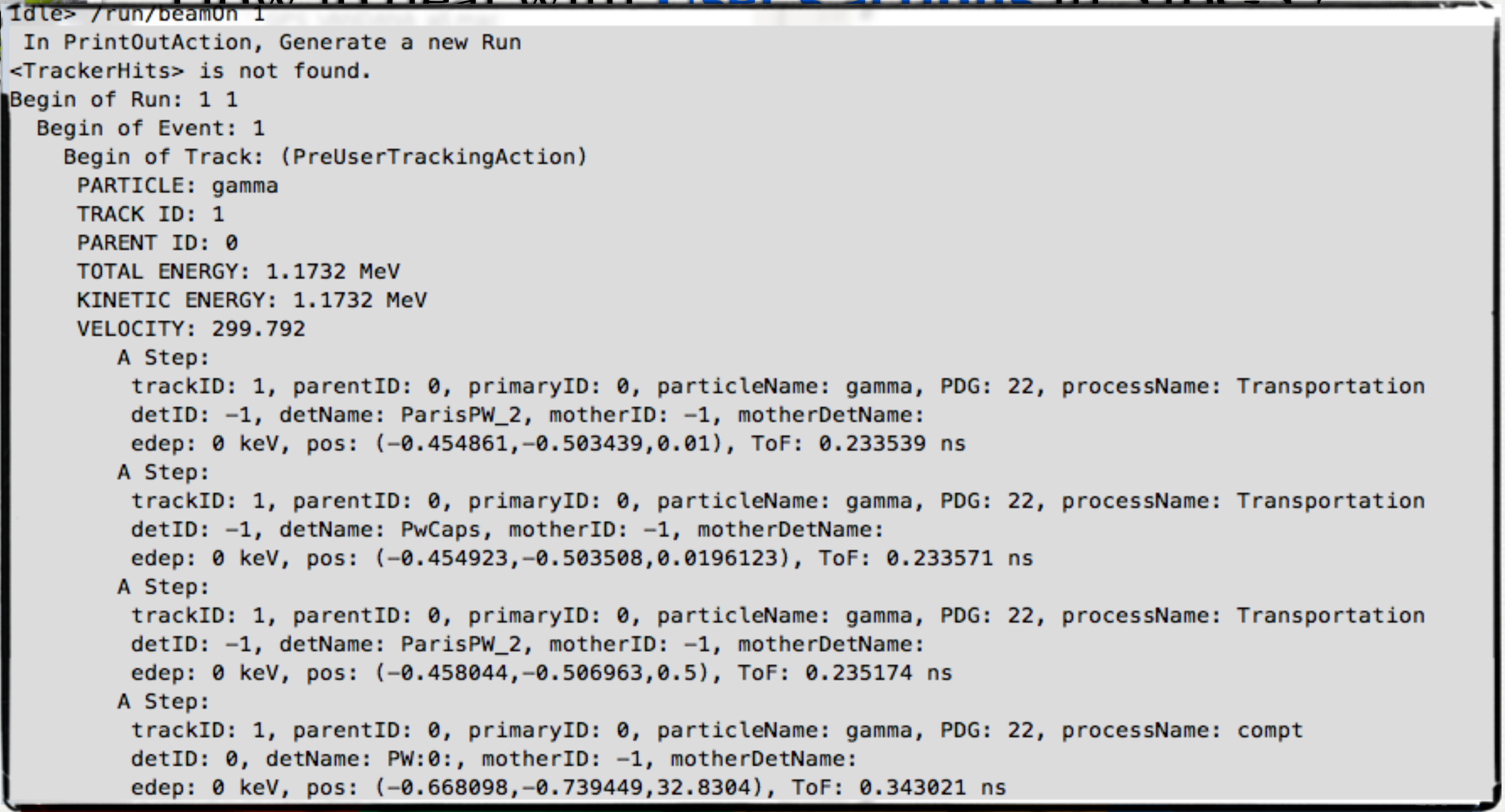
How to deal with **User's actions** in SToGS ?

At the beginning of G4

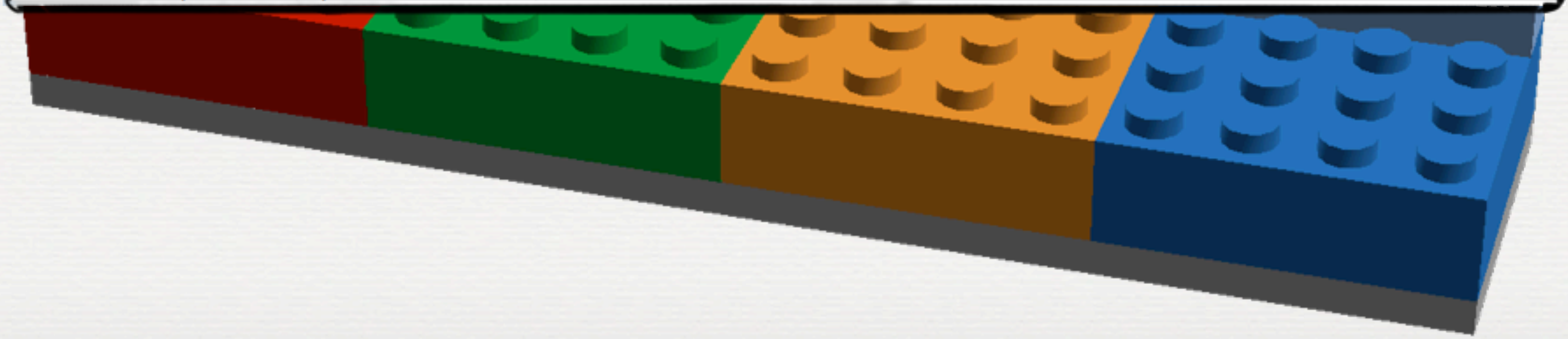
- almost no standard way to get out of G4 the physics
- Some possibilities now … not used in SToGS



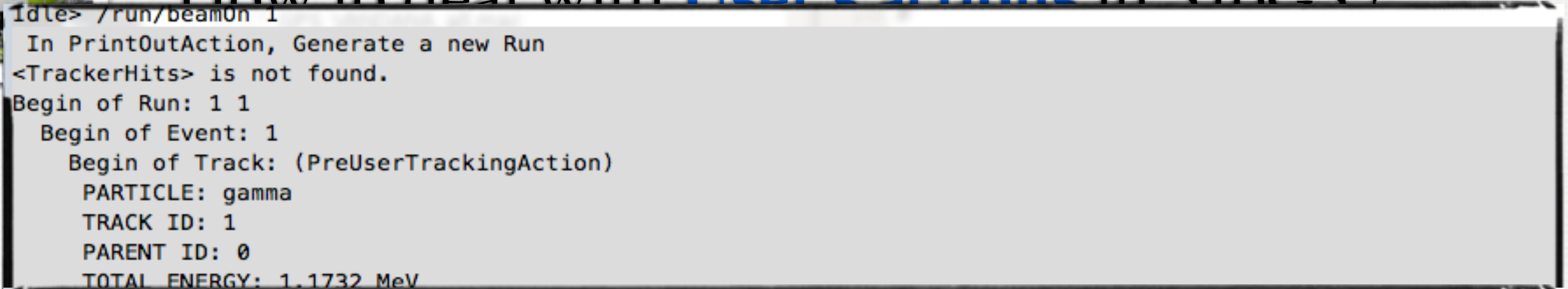
How to deal with User's actions in SToGS ?



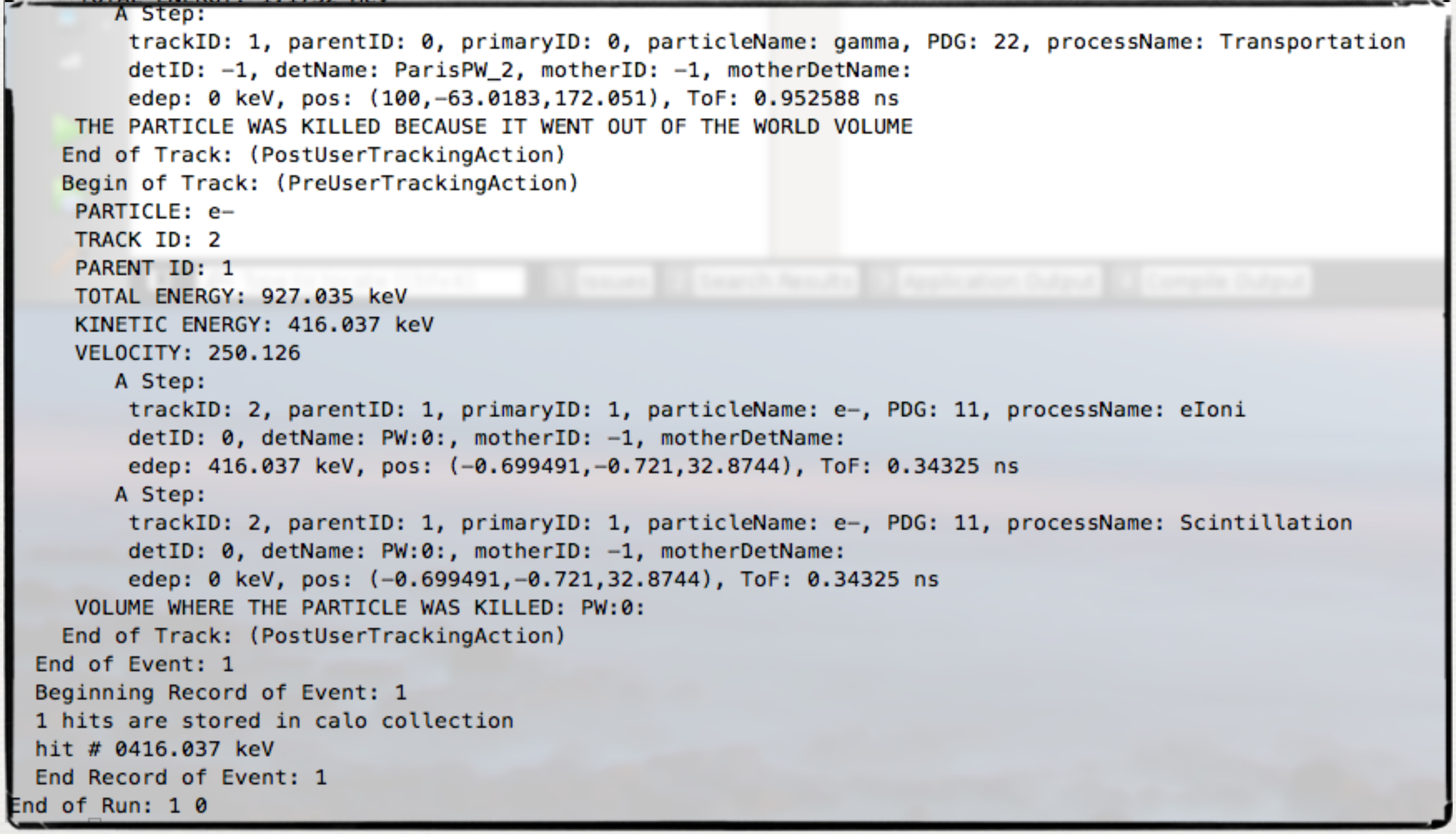
```
Idle> /run/beamOn 1
 In PrintOutAction, Generate a new Run
<TrackerHits> is not found.
Begin of Run: 1 1
  Begin of Event: 1
    Begin of Track: (PreUserTrackingAction)
     PARTICLE: gamma
     TRACK ID: 1
     PARENT ID: 0
     TOTAL ENERGY: 1.1732 MeV
     KINETIC ENERGY: 1.1732 MeV
     VELOCITY: 299.792
        A Step:
         trackID: 1, parentID: 0, primaryID: 0, particleName: gamma, PDG: 22, processName: Transportation
         detID: -1, detName: ParisPW_2, motherID: -1, motherDetName:
         edep: 0 keV, pos: (-0.454861,-0.503439,0.01), ToF: 0.233539 ns
        A Step:
         trackID: 1, parentID: 0, primaryID: 0, particleName: gamma, PDG: 22, processName: Transportation
         detID: -1, detName: PwCaps, motherID: -1, motherDetName:
         edep: 0 keV, pos: (-0.454923,-0.503508,0.0196123), ToF: 0.233571 ns
        A Step:
         trackID: 1, parentID: 0, primaryID: 0, particleName: gamma, PDG: 22, processName: Transportation
         detID: -1, detName: ParisPW_2, motherID: -1, motherDetName:
         edep: 0 keV, pos: (-0.458044,-0.506963,0.5), ToF: 0.235174 ns
        A Step:
         trackID: 1, parentID: 0, primaryID: 0, particleName: gamma, PDG: 22, processName: compt
         detID: 0, detName: PW:0:, motherID: -1, motherDetName:
         edep: 0 keV, pos: (-0.668098,-0.739449,32.8304), ToF: 0.343021 ns
```



How to deal with User's actions in SToGS ?



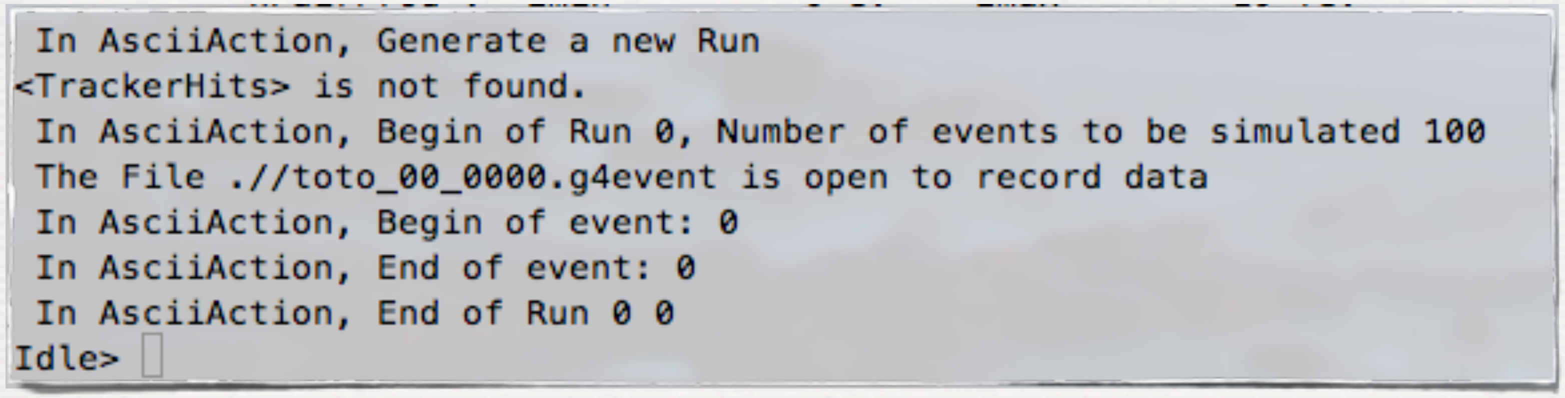
```
Idle> /run/beamOn 1
 In PrintOutAction, Generate a new Run
<TrackerHits> is not found.
Begin of Run: 1 1
  Begin of Event: 1
    Begin of Track: (PreUserTrackingAction)
     PARTICLE: gamma
     TRACK ID: 1
     PARENT ID: 0
     TOTAL ENERGY: 1.1732 MeV
```



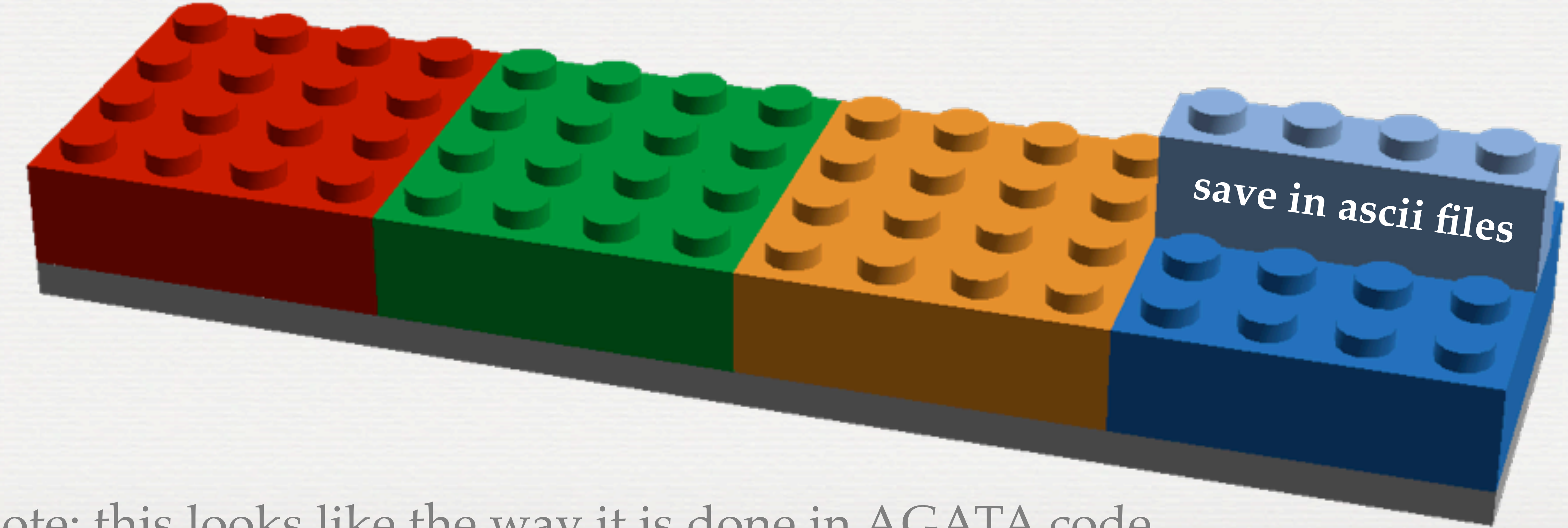
```
        A Step:
         trackID: 1, parentID: 0, primaryID: 0, particleName: gamma, PDG: 22, processName: Transportation
         detID: -1, detName: ParisPW_2, motherID: -1, motherDetName:
         edep: 0 keV, pos: (100,-63.0183,172.051), ToF: 0.952588 ns
     THE PARTICLE WAS KILLED BECAUSE IT WENT OUT OF THE WORLD VOLUME
    End of Track: (PostUserTrackingAction)
    Begin of Track: (PreUserTrackingAction)
     PARTICLE: e-
     TRACK ID: 2
     PARENT ID: 1
     TOTAL ENERGY: 927.035 keV
     KINETIC ENERGY: 416.037 keV
     VELOCITY: 250.126
        A Step:
         trackID: 2, parentID: 1, primaryID: 1, particleName: e-, PDG: 11, processName: eIoni
         detID: 0, detName: PW:0:, motherID: -1, motherDetName:
         edep: 416.037 keV, pos: (-0.699491,-0.721,32.8744), ToF: 0.34325 ns
        A Step:
         trackID: 2, parentID: 1, primaryID: 1, particleName: e-, PDG: 11, processName: Scintillation
         detID: 0, detName: PW:0:, motherID: -1, motherDetName:
         edep: 0 keV, pos: (-0.699491,-0.721,32.8744), ToF: 0.34325 ns
     VOLUME WHERE THE PARTICLE WAS KILLED: PW:0:
    End of Track: (PostUserTrackingAction)
  End of Event: 1
  Beginning Record of Event: 1
  1 hits are stored in calo collection
  hit # 0416.037 keV
  End Record of Event: 1
End of Run: 1 0
```

How to deal with **User's actions** in SToGS ?

More useful for R&D like phases ...
not adapted for simulations with large # of events



```
 In AsciiAction, Generate a new Run
<TrackerHits> is not found.
 In AsciiAction, Begin of Run 0, Number of events to be simulated 100
 The File .//toto_00_0000.g4event is open to record data
 In AsciiAction, Begin of event: 0
 In AsciiAction, End of event: 0
 In AsciiAction, End of Run 0 0
Idle> 
```

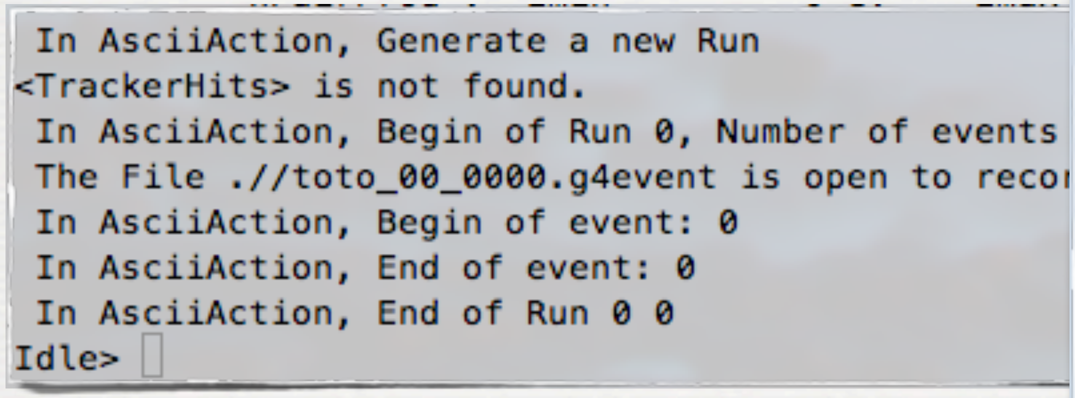


Note: this looks like the way it is done in AGATA code

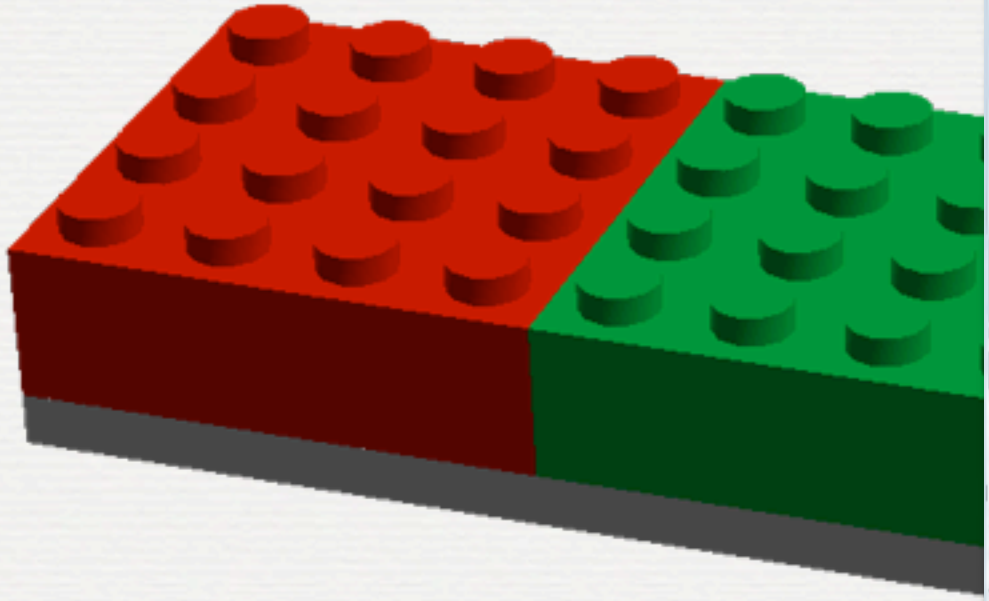
How to deal with

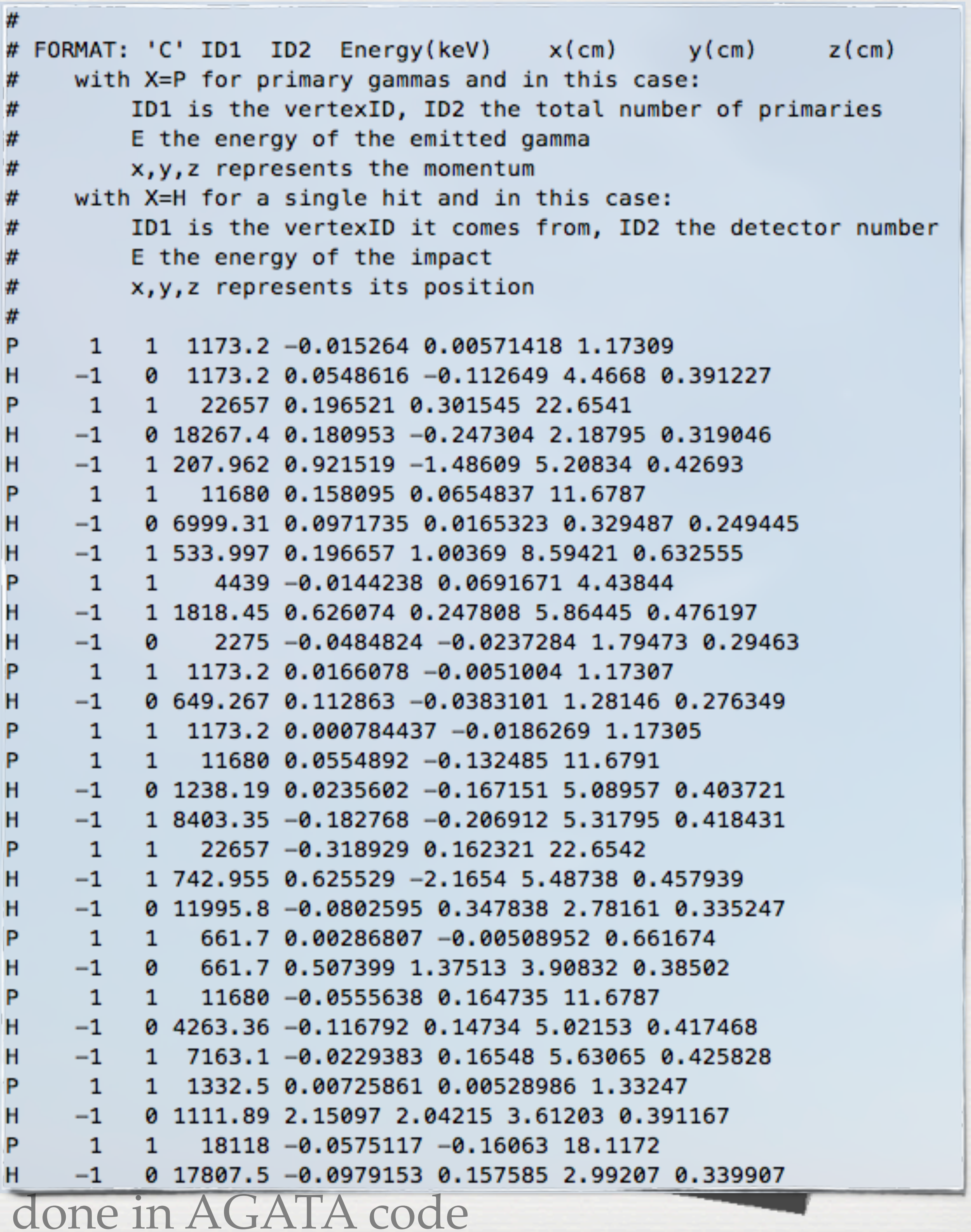
More useful f

not adapted for simu



```
 In AsciiAction, Generate a new Run
<TrackerHits> is not found.
 In AsciiAction, Begin of Run 0, Number of events
 The File .//toto_00_0000.g4event is open to recor
 In AsciiAction, Begin of event: 0
 In AsciiAction, End of event: 0
 In AsciiAction, End of Run 0 0
Idle> 
```





```
#
# FORMAT: 'C' ID1  ID2  Energy(keV)    x(cm)     y(cm)     z(cm)
#    with X=P for primary gammas and in this case:
#        ID1 is the vertexID, ID2 the total number of primaries
#        E the energy of the emitted gamma
#        x,y,z represents the momentum
#    with X=H for a single hit and in this case:
#        ID1 is the vertexID it comes from, ID2 the detector number
#        E the energy of the impact
#        x,y,z represents its position
#
P    1   1  1173.2 -0.015264 0.00571418 1.17309
H   -1   0  1173.2 0.0548616 -0.112649 4.4668 0.391227
P    1   1   22657 0.196521 0.301545 22.6541
H   -1   0 18267.4 0.180953 -0.247304 2.18795 0.319046
H   -1   1 207.962 0.921519 -1.48609 5.20834 0.42693
P    1   1   11680 0.158095 0.0654837 11.6787
H   -1   0 6999.31 0.0971735 0.0165323 0.329487 0.249445
H   -1   1 533.997 0.196657 1.00369 8.59421 0.632555
P    1   1    4439 -0.0144238 0.0691671 4.43844
H   -1   1 1818.45 0.626074 0.247808 5.86445 0.476197
H   -1   0    2275 -0.0484824 -0.0237284 1.79473 0.29463
P    1   1  1173.2 0.0166078 -0.0051004 1.17307
H   -1   0 649.267 0.112863 -0.0383101 1.28146 0.276349
P    1   1  1173.2 0.000784437 -0.0186269 1.17305
P    1   1   11680 0.0554892 -0.132485 11.6791
H   -1   0 1238.19 0.0235602 -0.167151 5.08957 0.403721
H   -1   1 8403.35 -0.182768 -0.206912 5.31795 0.418431
P    1   1   22657 -0.318929 0.162321 22.6542
H   -1   1 742.955 0.625529 -2.1654 5.48738 0.457939
H   -1   0 11995.8 -0.0802595 0.347838 2.78161 0.335247
P    1   1   661.7 0.00286807 -0.00508952 0.661674
H   -1   0   661.7 0.507399 1.37513 3.90832 0.38502
P    1   1   11680 -0.0555638 0.164735 11.6787
H   -1   0 4263.36 -0.116792 0.14734 5.02153 0.417468
H   -1   1  7163.1 -0.0229383 0.16548 5.63065 0.425828
P    1   1  1332.5 0.00725861 0.00528986 1.33247
H   -1   0 1111.89 2.15097 2.04215 3.61203 0.391167
P    1   1   18118 -0.0575117 -0.16063 18.1172
H   -1   0 17807.5 -0.0979153 0.157585 2.99207 0.339907
```

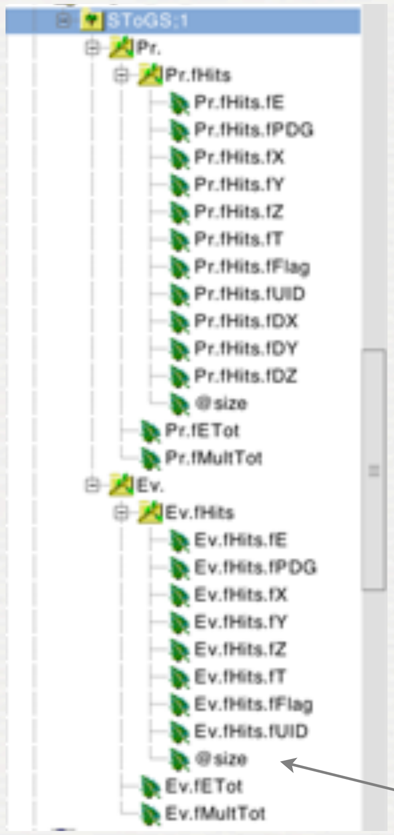
Note: this looks like the way it is done in AGATA code

How to deal with User's actions in SToGS ?

Total energy

Total # of impacts

[H,K inheritance of PARIS]



```
SToGS;1
  Pr.
    Pr.fHits
      Pr.fHits.fE
      Pr.fHits.fPDG
      Pr.fHits.fX
      Pr.fHits.fY
      Pr.fHits.fZ
      Pr.fHits.fT
      Pr.fHits.fFlag
      Pr.fHits.fUID
      Pr.fHits.fDX
      Pr.fHits.fDY
      Pr.fHits.fDZ
      @size
    Pr.fETot
    Pr.fMultTot
  Ev.
    Ev.fHits
      Ev.fHits.fE
      Ev.fHits.fPDG
      Ev.fHits.fX
      Ev.fHits.fY
      Ev.fHits.fZ
      Ev.fHits.fT
      Ev.fHits.fFlag
      Ev.fHits.fUID
      @size
    Ev.fETot
    Ev.fMultTot
```

ARRAYS

Energy of each hit

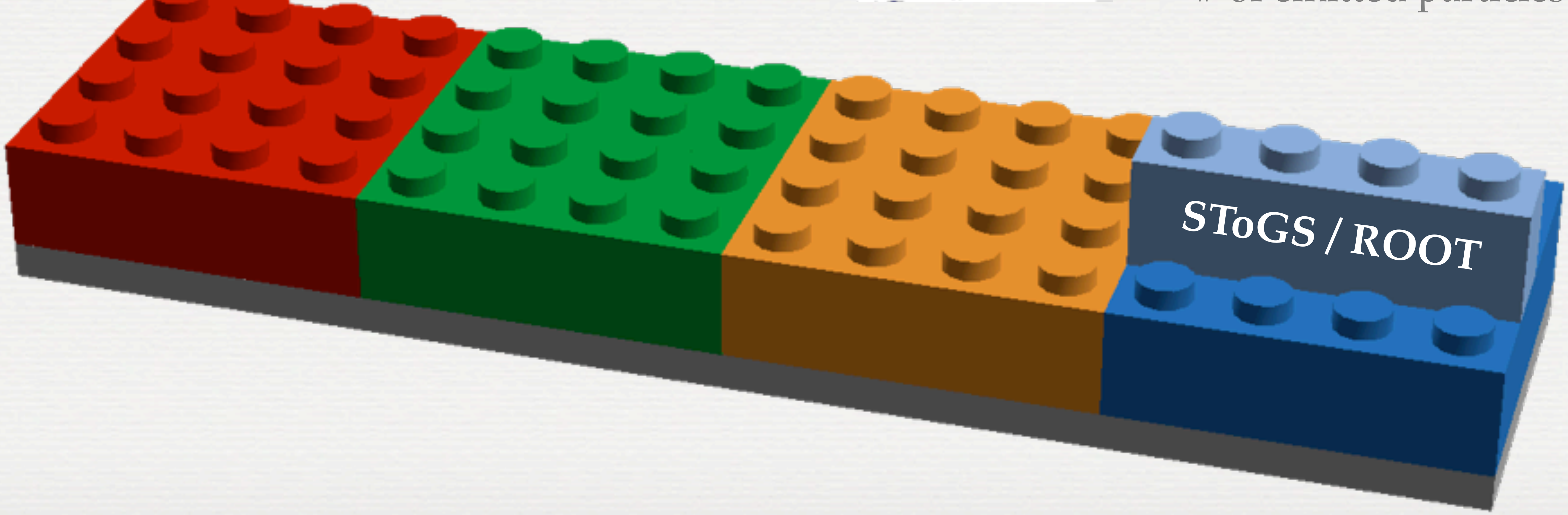
PDG codes

positions 4D

Quality flags

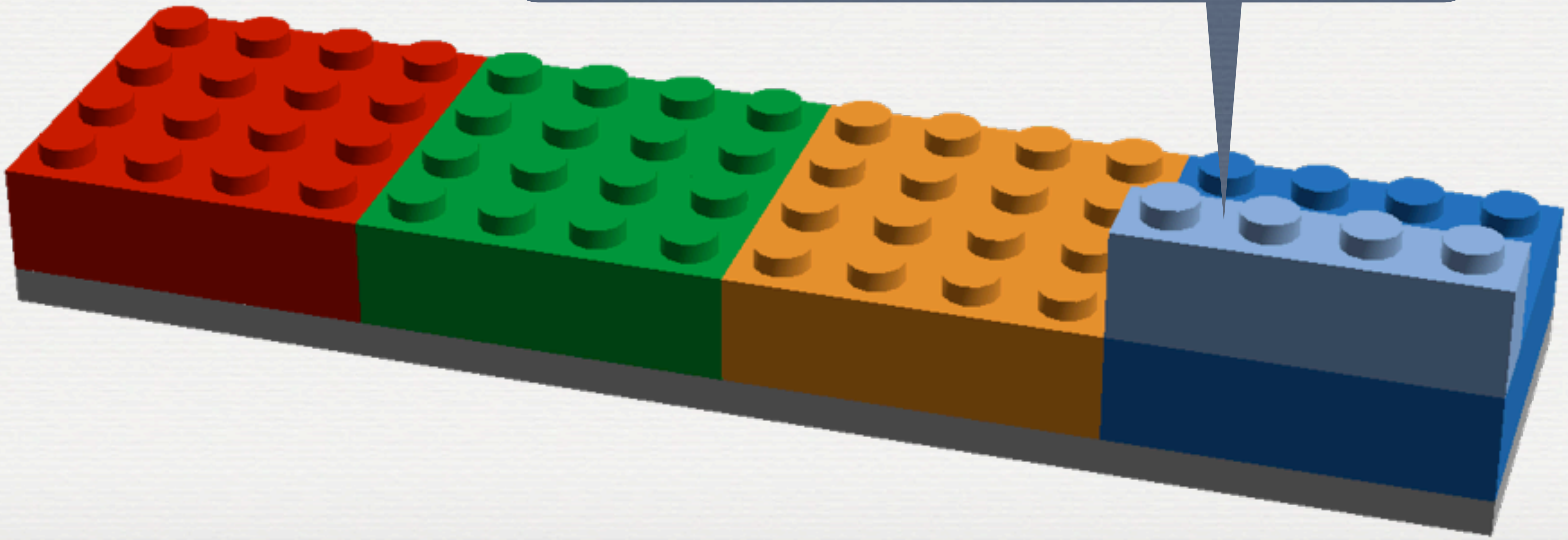
Unique detector IDs

# of emitted particles



How to deal with **User's actions** in SToGS ?

The system could be extended
so that a user can plug
his way to extract things ...



How to get / install / run SToGS ?

- To get the package git repository
  https://github.com/stezow/stogs
- It requires GEANT4 compiled with gdml support
- It uses cmake to configure

- It built three different applications
  - SToGS_DetectorBuilder
    to construct / check a setup
  - SToGS_Source
    to run an experiment with external primaries*
  - *SToGS_Beam*
    not yet implemented … beam like experiments

* primaries and tracking decoupled

How to get / install / run SToGS ?

- Some commands

```
git clone https://github.com/stezow/stogs.git SToGS

cmake -DGeant4_DIR=/where/geant4/has/been/built ../
```

- To get some documentation
  - User's guide on ATRUIM [https://atrium.in2p3.fr/]
  - The github includes

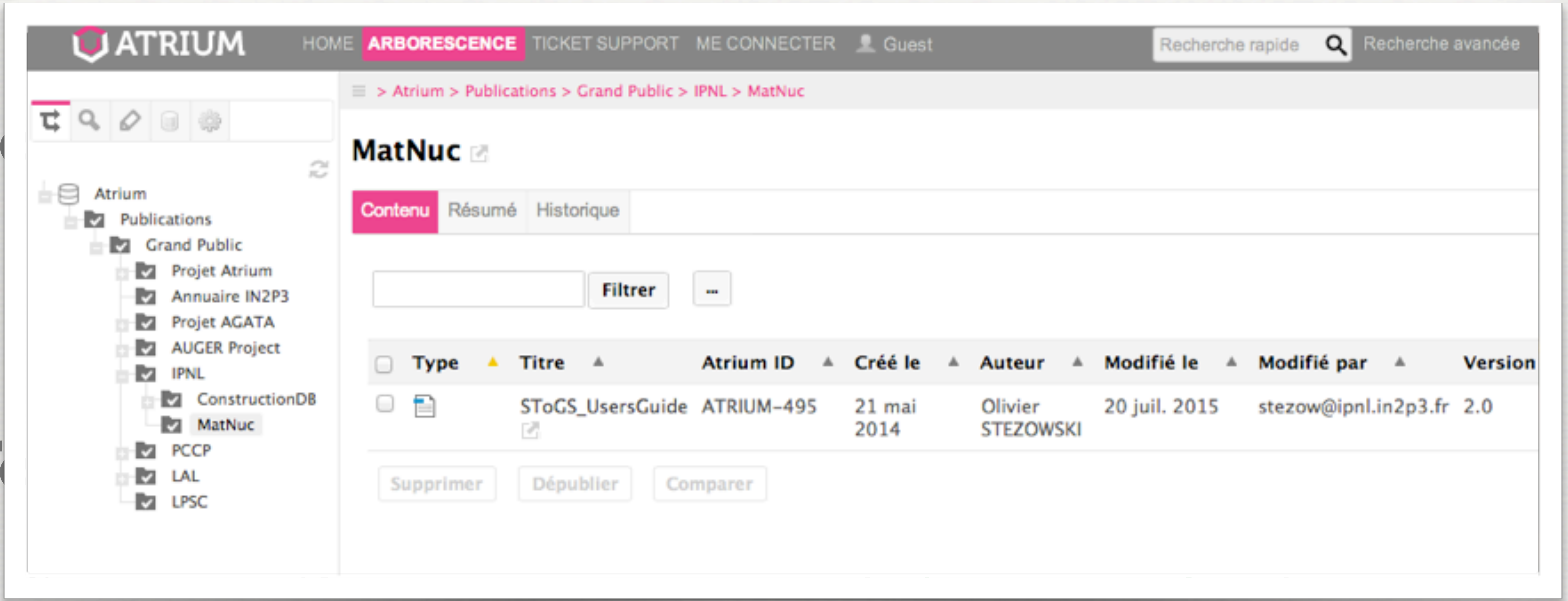
    bugtracker / wiki / news / activities
  - Doxygen [http://www.ipnl.in2p3.fr/perso/stezowski/SToGS/shtmldoc/]

    Documentation of the code itself

How to get / install / run SToGS ?

- S

- T



> Atrium > Publications > Grand Public > IPNL > MatNuc

- Atrium
  - Publications
    - Grand Public
      - Projet Atrium
      - Annuaire IN2P3
      - Projet AGATA
      - AUGER Project
      - IPNL
        - ConstructionDB
        - MatNuc
      - PCCP
      - LAL
      - LPSC

MatNuc

□	Type	Titre	Atrium ID	Créé le	Auteur	Modifié le	Modifié par	Version
□		SToGS_UsersGuide	ATRIUM-495	21 mai 2014	Olivier STEZOWSKI	20 juil. 2015	stezow@ipnl.in2p3.fr	2.0

- The github includes bugtracker / wiki / news / activities
- Doxygen [http://www.ipnl.in2p3.fr/perso/stezowski/SToGS/shtmldoc/] Documentation of the code itself

How to get / install / run SToGS ?

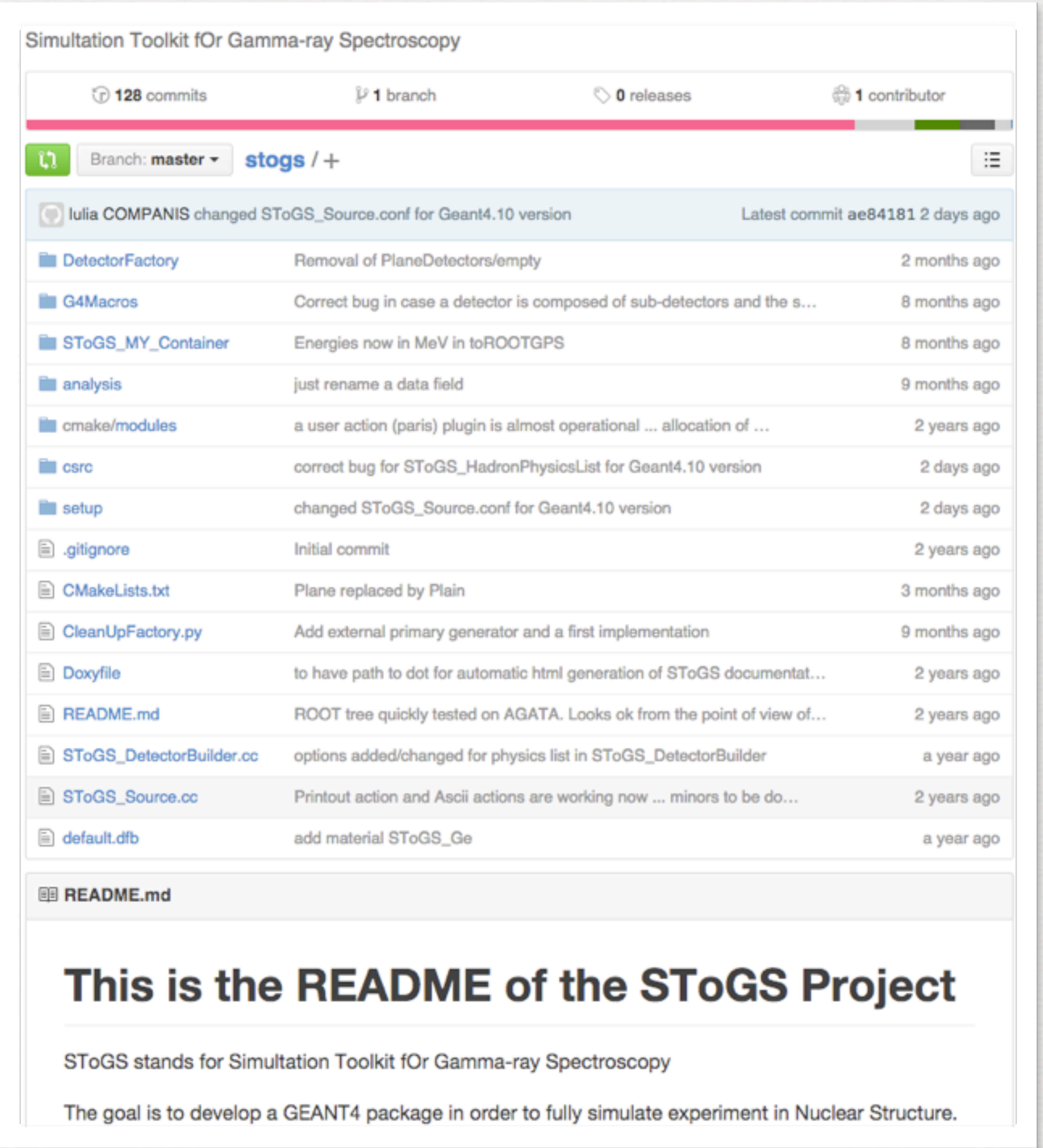
- Some commands

```
git clone https://github.com/stezow/stogs.git SToGS
cmake -DGeant4_DIR=/where/geant4/has/been/built ../
```

- To get some documentation
  - User's guide on ATRUIM [https://**atrium**.**in2p3**.fr/]
  - The github includes
    bugtracker / wiki / news / activities
  - Doxygen [**http://www.ipnl.in2p3.fr/perso/stezowski/SToGS/shtmldoc/**]
    Documentation of the code itself

How to

- Some command
    git clone https://
    cmake -DGeant
- To get some doc
    - User's guid
    - The github bugtracker
    - Doxygen [h Document



Simulation Toolkit fOr Gamma-ray Spectroscopy

128 commits | 1 branch | 0 releases | 1 contributor

Branch: master | stogs / +

Iulia COMPANIS changed SToGS_Source.conf for Geant4.10 version | Latest commit ae84181 2 days ago

DetectorFactory	Removal of PlaneDetectors/empty	2 months ago
G4Macros	Correct bug in case a detector is composed of sub-detectors and the s…	8 months ago
SToGS_MY_Container	Energies now in MeV in toROOTGPS	8 months ago
analysis	just rename a data field	9 months ago
cmake/modules	a user action (paris) plugin is almost operational ... allocation of ...	2 years ago
csrc	correct bug for SToGS_HadronPhysicsList for Geant4.10 version	2 days ago
setup	changed SToGS_Source.conf for Geant4.10 version	2 days ago
.gitignore	Initial commit	2 years ago
CMakeLists.txt	Plane replaced by Plain	3 months ago
CleanUpFactory.py	Add external primary generator and a first implementation	9 months ago
Doxyfile	to have path to dot for automatic html generation of SToGS documentat…	2 years ago
README.md	ROOT tree quickly tested on AGATA. Looks ok from the point of view of…	2 years ago
SToGS_DetectorBuilder.cc	options added/changed for physics list in SToGS_DetectorBuilder	a year ago
SToGS_Source.cc	Printout action and Ascii actions are working now ... minors to be do…	2 years ago
default.dfb	add material SToGS_Ge	a year ago

README.md

This is the README of the SToGS Project

SToGS stands for Simulation Toolkit fOr Gamma-ray Spectroscopy

The goal is to develop a GEANT4 package in order to fully simulate experiment in Nuclear Structure.

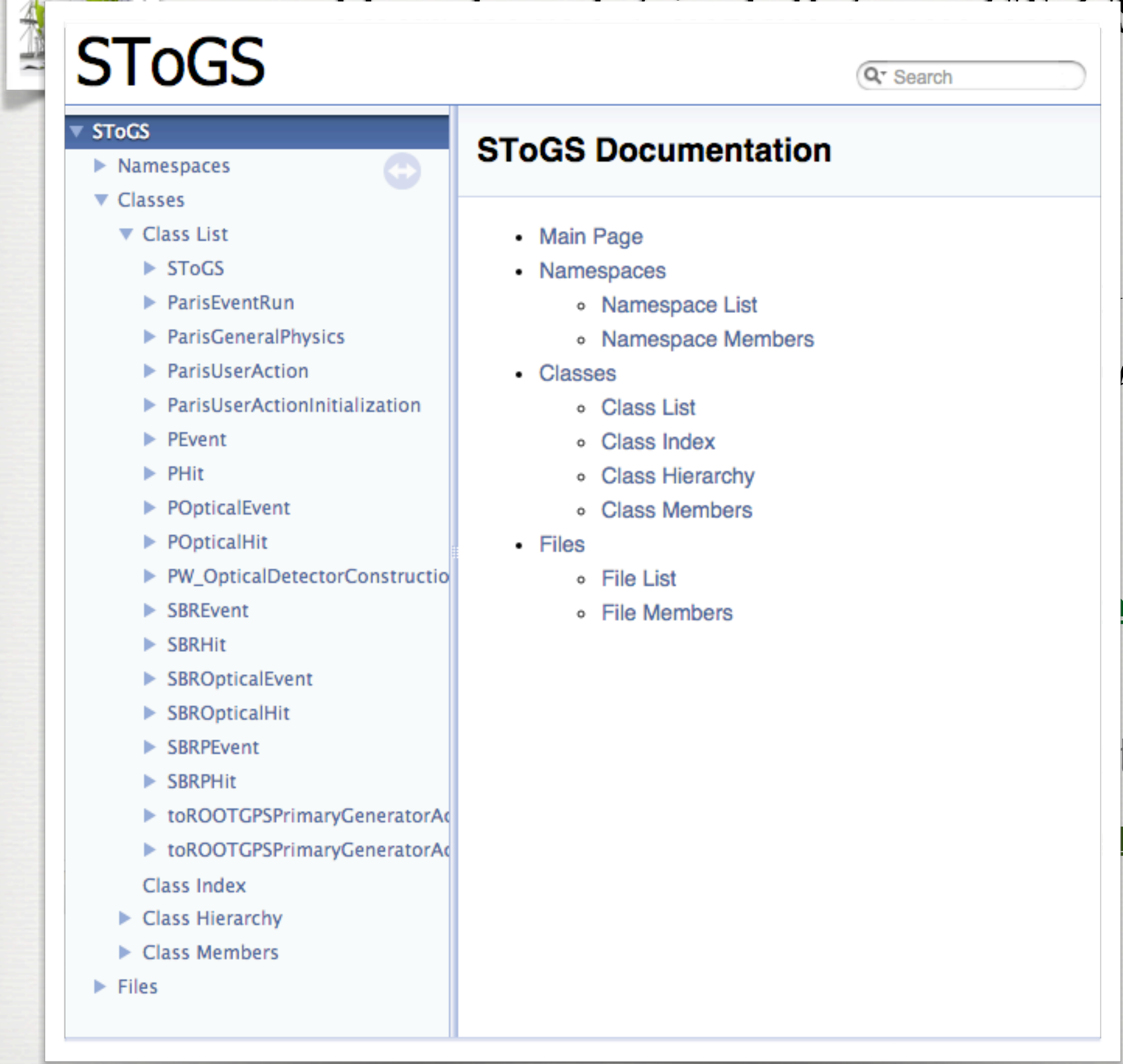
How to get / install / run SToGS ?

- Some commands

```
git clone https://github.com/stezow/stogs.git SToGS
cmake -DGeant4_DIR=/where/geant4/has/been/built ../
```

- To get some documentation
  - User's guide on ATRUIM [https://atrium.in2p3.fr/]
  - The github includes
    bugtracker / wiki / news / activities
  - Doxygen [http://www.ipnl.in2p3.fr/perso/stezowski/SToGS/shtmldoc/]
    Documentation of the code itself

S ?



SToGS

🔍 Search

- SToGS
  - Namespaces
  - Classes
    - Class List
      - SToGS
      - ParisEventRun
      - ParisGeneralPhysics
      - ParisUserAction
      - ParisUserActionInitialization
      - PEvent
      - PHit
      - POpticalEvent
      - POpticalHit
      - PW_OpticalDetectorConstructio
      - SBREvent
      - SBRHit
      - SBROpticalEvent
      - SBROpticalHit
      - SBRPEvent
      - SBRPHit
      - toROOTGPSPrimaryGeneratorAc
      - toROOTGPSPrimaryGeneratorAc
    - Class Index
    - Class Hierarchy
    - Class Members
  - Files

SToGS Documentation

- Main Page
- Namespaces
  - Namespace List
  - Namespace Members
- Classes
  - Class List
  - Class Index
  - Class Hierarchy
  - Class Members
- Files
  - File List
  - File Members

S

uilt ../

.in2p3.fr/]

ties

ki/SToGS/shtmldoc/]

How to get / install / run SToGS ?

- Some commands

```
git clone https://github.com/stezow/stogs.git SToGS

cmake -DGeant4_DIR=/where/geant4/has/been/built ../
```

- To get some documentation
  - User’s guide on ATRUIM [https://**atrium**.**in2p3**.fr/]
  - The github includes

    bugtracker / wiki / news / activities
  - Doxygen [**http://www.ipnl.in2p3.fr/perso/stezowski/SToGS/shtmldoc/**]

    Documentation of the code itself

How to deal with **geometries** in SToGS factory ?

- The factories contains some standard detectors

  They are mainly built first time SToGS_DetectorBuilder is called
- System largely based on gdml

  It could be exchanged with other packages G4/ROOT

  It brings geometries at the user's level through text files

Ex:

aChateau2Crystal.gdml

aChateau2Crystal.amap

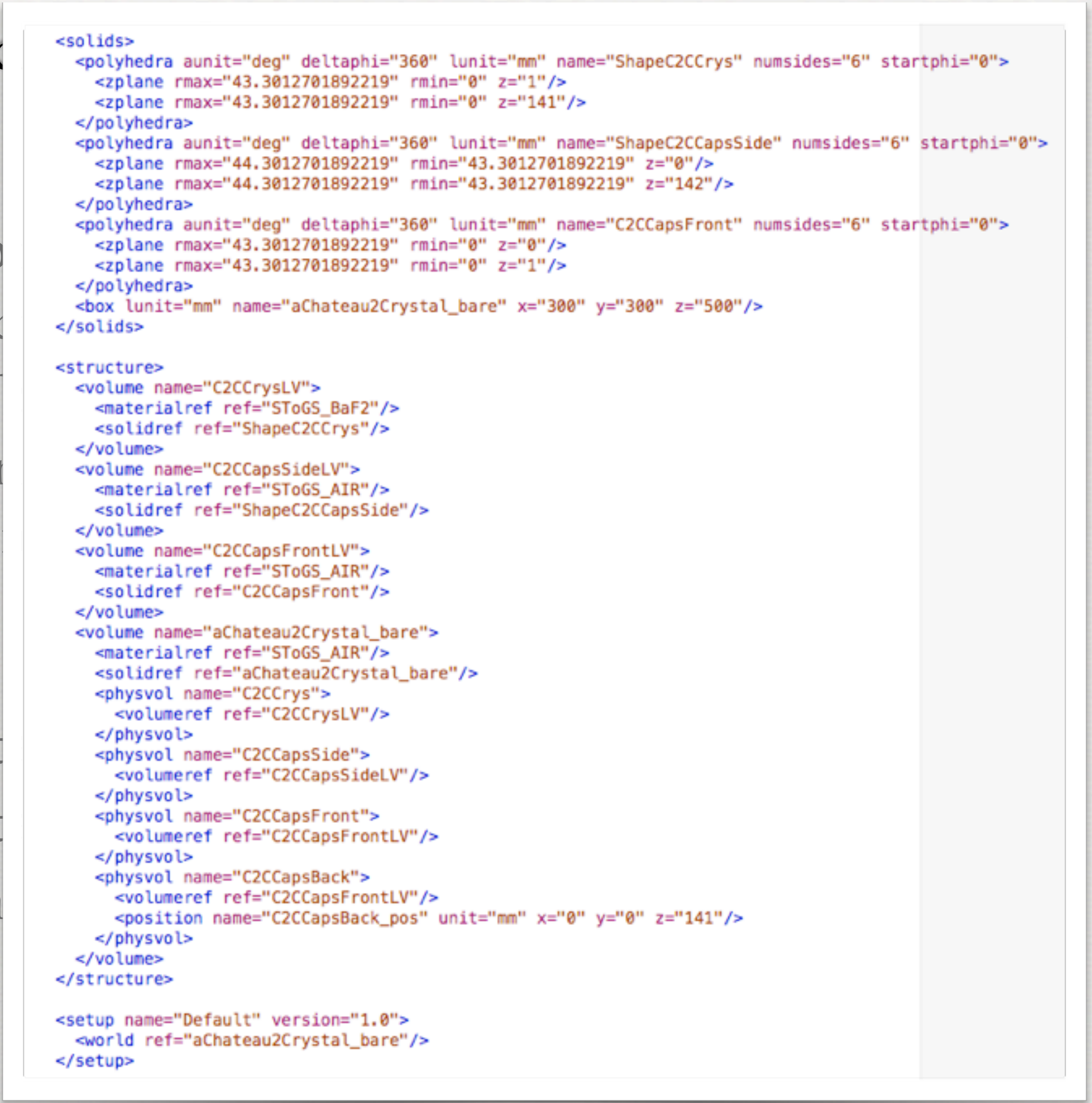
Chateau2Crystal.dmap

How to c

- The factories co
  They are mainly b
- System largely l
  It could be exchar
  It brings geometr

Ex:

aChateau2Cryst
aChateau2Cryst
Chateau2Crysta



```
<solids>
  <polyhedra aunit="deg" deltaphi="360" lunit="mm" name="ShapeC2CCrys" numsides="6" startphi="0">
    <zplane rmax="43.3012701892219" rmin="0" z="1"/>
    <zplane rmax="43.3012701892219" rmin="0" z="141"/>
  </polyhedra>
  <polyhedra aunit="deg" deltaphi="360" lunit="mm" name="ShapeC2CCapsSide" numsides="6" startphi="0">
    <zplane rmax="44.3012701892219" rmin="43.3012701892219" z="0"/>
    <zplane rmax="44.3012701892219" rmin="43.3012701892219" z="142"/>
  </polyhedra>
  <polyhedra aunit="deg" deltaphi="360" lunit="mm" name="C2CCapsFront" numsides="6" startphi="0">
    <zplane rmax="43.3012701892219" rmin="0" z="0"/>
    <zplane rmax="43.3012701892219" rmin="0" z="1"/>
  </polyhedra>
  <box lunit="mm" name="aChateau2Crystal_bare" x="300" y="300" z="500"/>
</solids>

<structure>
  <volume name="C2CCrysLV">
    <materialref ref="SToGS_BaF2"/>
    <solidref ref="ShapeC2CCrys"/>
  </volume>
  <volume name="C2CCapsSideLV">
    <materialref ref="SToGS_AIR"/>
    <solidref ref="ShapeC2CCapsSide"/>
  </volume>
  <volume name="C2CCapsFrontLV">
    <materialref ref="SToGS_AIR"/>
    <solidref ref="C2CCapsFront"/>
  </volume>
  <volume name="aChateau2Crystal_bare">
    <materialref ref="SToGS_AIR"/>
    <solidref ref="aChateau2Crystal_bare"/>
    <physvol name="C2CCrys">
      <volumeref ref="C2CCrysLV"/>
    </physvol>
    <physvol name="C2CCapsSide">
      <volumeref ref="C2CCapsSideLV"/>
    </physvol>
    <physvol name="C2CCapsFront">
      <volumeref ref="C2CCapsFrontLV"/>
    </physvol>
    <physvol name="C2CCapsBack">
      <volumeref ref="C2CCapsFrontLV"/>
      <position name="C2CCapsBack_pos" unit="mm" x="0" y="0" z="141"/>
    </physvol>
  </volume>
</structure>

<setup name="Default" version="1.0">
  <world ref="aChateau2Crystal_bare"/>
</setup>
```

How to deal with geometries in SToGS factory ?

- The factories contains some standard detectors

  They are mainly built first time SToGS_DetectorBuilder is called
- System largely based on gdml

  It could be exchanged with other packages G4/ROOT

  It brings geometries at the user's level through text files

Ex:

aChateau2Crystal.gdml

aChateau2Crystal.amap

Chateau2Crystal.dmap

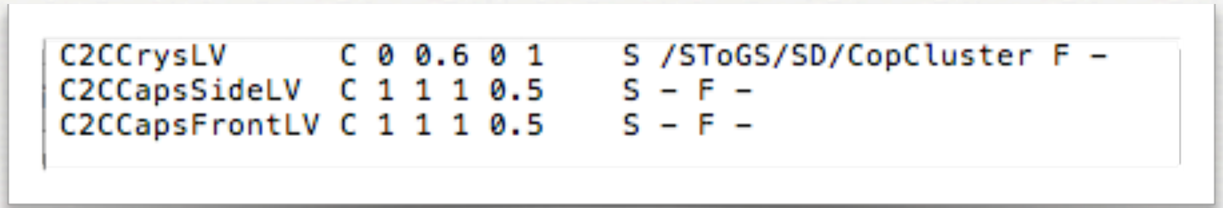
How to deal with geometries in SToGS factory ?

- The factories contains some standard detectors

  They are mainly built first time SToGS_DetectorBuilder is called
- System largely based on gdml

  It could be exchanged with other packages G4/ROOT

  It brings geometries at the user's level through text files



```
C2CCrysLV       C 0 0.6 0 1     S /SToGS/SD/CopCluster F -
C2CCapsSideLV   C 1 1 1 0.5     S - F -
C2CCapsFrontLV  C 1 1 1 0.5     S - F -
```

Ex:

aChateau2Crystal.gdml

aChateau2Crystal.amap

Chateau2Crystal.dmap

How to deal with **geometries** in SToGS factory ?

- The factories contains some standard detectors

  They are mainly built first time SToGS_DetectorBuilder is called
- System largely based on gdml

  It could be exchanged with other packages G4/ROOT

  It brings geometries at the user's level through text files

Ex:

aChateau2Crystal.gdml

aChateau2Crystal.amap

Chateau2Crystal.dmap

How to deal with **geometries** in SToGS factory ?

- The factories contains some standard detectors

  They are mainly built first time SToGS_DetectorBuilder is called

- System largely based on gdml

  It could be exchanged with other packages G4/ROOT

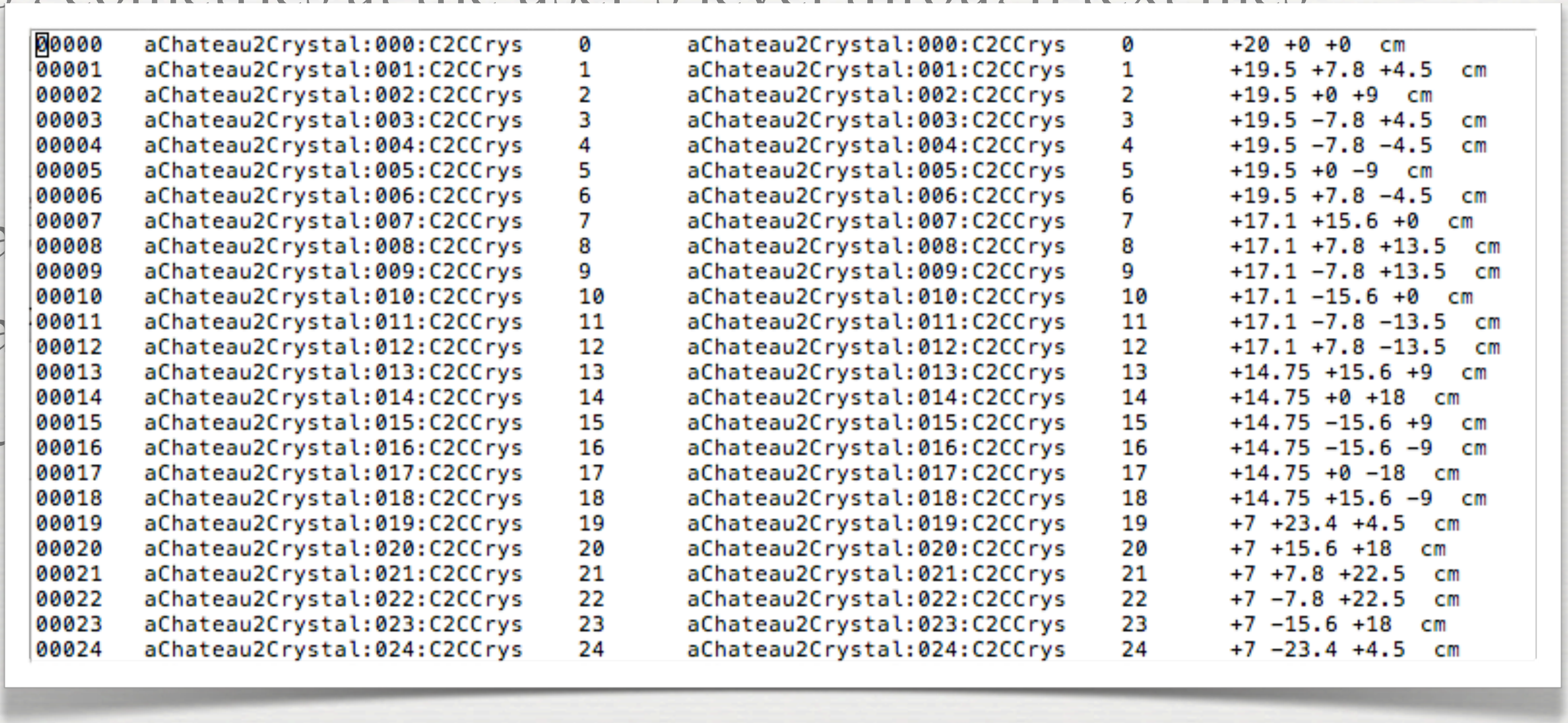
  It brings geometries at the user's level through text files

Ex:

aChatea

aChatea

Chateau



```
00000   aChateau2Crystal:000:C2CCrys    0       aChateau2Crystal:000:C2CCrys    0       +20 +0 +0  cm
00001   aChateau2Crystal:001:C2CCrys    1       aChateau2Crystal:001:C2CCrys    1       +19.5 +7.8 +4.5  cm
00002   aChateau2Crystal:002:C2CCrys    2       aChateau2Crystal:002:C2CCrys    2       +19.5 +0 +9  cm
00003   aChateau2Crystal:003:C2CCrys    3       aChateau2Crystal:003:C2CCrys    3       +19.5 -7.8 +4.5  cm
00004   aChateau2Crystal:004:C2CCrys    4       aChateau2Crystal:004:C2CCrys    4       +19.5 -7.8 -4.5  cm
00005   aChateau2Crystal:005:C2CCrys    5       aChateau2Crystal:005:C2CCrys    5       +19.5 +0 -9  cm
00006   aChateau2Crystal:006:C2CCrys    6       aChateau2Crystal:006:C2CCrys    6       +19.5 +7.8 -4.5  cm
00007   aChateau2Crystal:007:C2CCrys    7       aChateau2Crystal:007:C2CCrys    7       +17.1 +15.6 +0  cm
00008   aChateau2Crystal:008:C2CCrys    8       aChateau2Crystal:008:C2CCrys    8       +17.1 +7.8 +13.5  cm
00009   aChateau2Crystal:009:C2CCrys    9       aChateau2Crystal:009:C2CCrys    9       +17.1 -7.8 +13.5  cm
00010   aChateau2Crystal:010:C2CCrys    10      aChateau2Crystal:010:C2CCrys    10      +17.1 -15.6 +0  cm
00011   aChateau2Crystal:011:C2CCrys    11      aChateau2Crystal:011:C2CCrys    11      +17.1 -7.8 -13.5  cm
00012   aChateau2Crystal:012:C2CCrys    12      aChateau2Crystal:012:C2CCrys    12      +17.1 +7.8 -13.5  cm
00013   aChateau2Crystal:013:C2CCrys    13      aChateau2Crystal:013:C2CCrys    13      +14.75 +15.6 +9  cm
00014   aChateau2Crystal:014:C2CCrys    14      aChateau2Crystal:014:C2CCrys    14      +14.75 +0 +18  cm
00015   aChateau2Crystal:015:C2CCrys    15      aChateau2Crystal:015:C2CCrys    15      +14.75 -15.6 +9  cm
00016   aChateau2Crystal:016:C2CCrys    16      aChateau2Crystal:016:C2CCrys    16      +14.75 -15.6 -9  cm
00017   aChateau2Crystal:017:C2CCrys    17      aChateau2Crystal:017:C2CCrys    17      +14.75 +0 -18  cm
00018   aChateau2Crystal:018:C2CCrys    18      aChateau2Crystal:018:C2CCrys    18      +14.75 +15.6 -9  cm
00019   aChateau2Crystal:019:C2CCrys    19      aChateau2Crystal:019:C2CCrys    19      +7 +23.4 +4.5  cm
00020   aChateau2Crystal:020:C2CCrys    20      aChateau2Crystal:020:C2CCrys    20      +7 +15.6 +18  cm
00021   aChateau2Crystal:021:C2CCrys    21      aChateau2Crystal:021:C2CCrys    21      +7 +7.8 +22.5  cm
00022   aChateau2Crystal:022:C2CCrys    22      aChateau2Crystal:022:C2CCrys    22      +7 -7.8 +22.5  cm
00023   aChateau2Crystal:023:C2CCrys    23      aChateau2Crystal:023:C2CCrys    23      +7 -15.6 +18  cm
00024   aChateau2Crystal:024:C2CCrys    24      aChateau2Crystal:024:C2CCrys    24      +7 -23.4 +4.5  cm
```

How to deal with **geometries** in SToGS factory ?

- The factories contains some standard detectors

  They are mainly built first time SToGS_DetectorBuilder is called
- System largely based on gdml

  It could be exchanged with other packages G4/ROOT

  It brings geometries at the user's level through text files

Ex:

aChateau2Crystal.gdml

aChateau2Crystal.amap

Chateau2Crystal.dmap

How to deal with **geometries** in SToGS factory ?

Here is a snapshot of the factories organization ...

```
tree -d DetectorFactory/
          DetectorFactory/
          |-- Arrays
          |   `-- Builders
          |-- Generics
          |   `-- Demos
          |-- MyFactory
          |   `-- Builders
          |-- MyStore
          |   `-- Builders
          |-- Imports
          |-- PlaneDetectors
          |-- Rooms
          |   `-- Builders
          |-- Scintillators
          |   `-- Builders
          `-- SemiConductors
              `-- Ge
                  `-- Builders
```

arrays already in the factory

detectors depending of a conf file

place to save setups

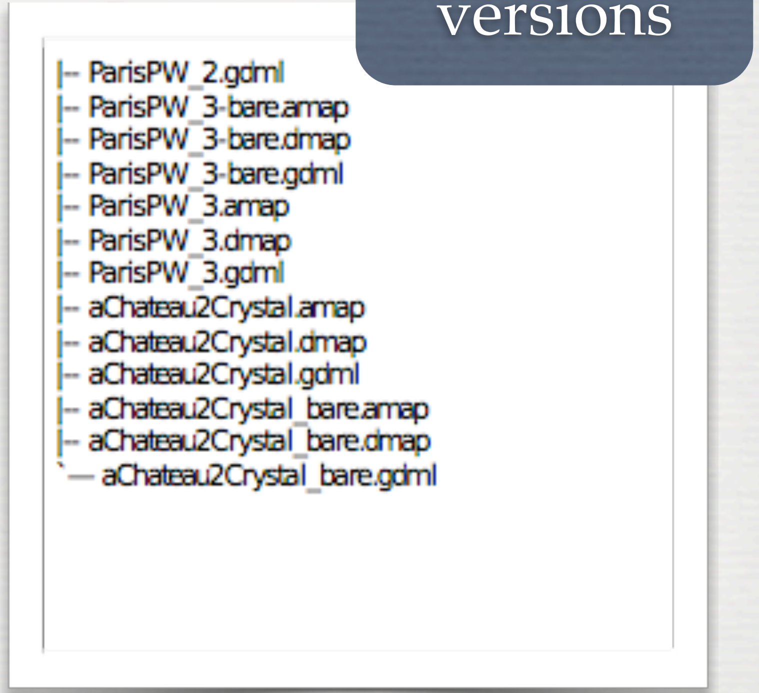
from other packages

basics shapes in gdml

place for rooms

organization by type

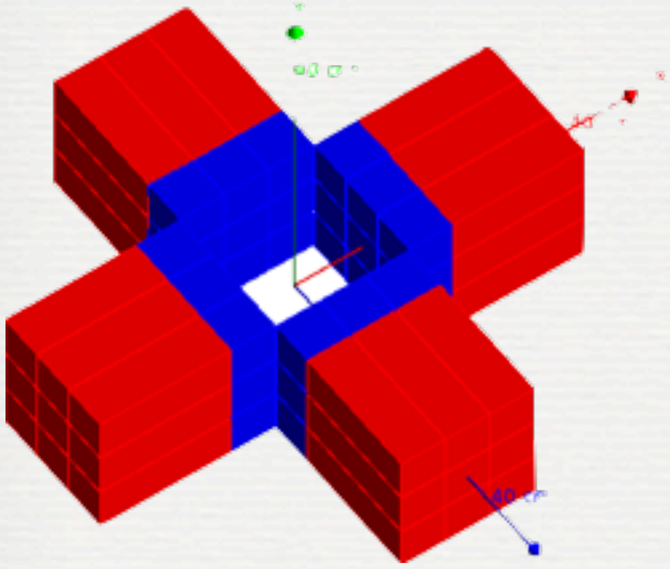
different versions

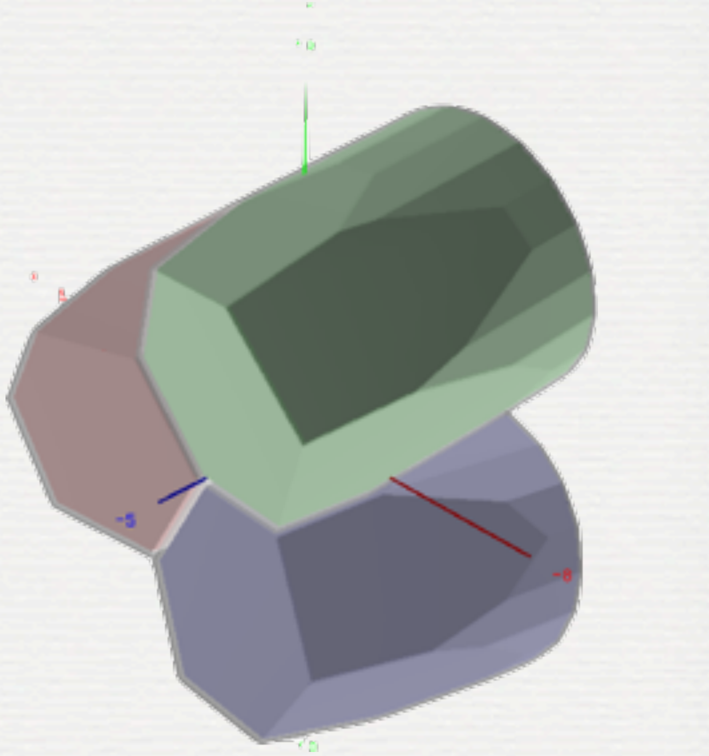


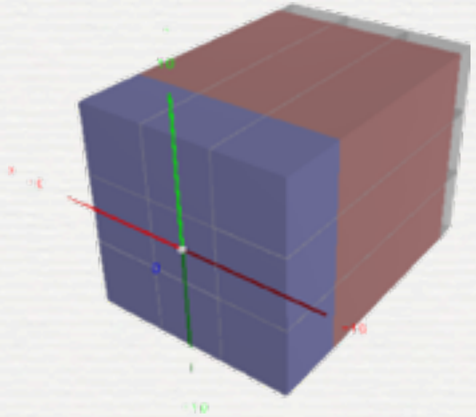
```
|-- ParisPW_2.gdml
|-- ParisPW_3-bare.amap
|-- ParisPW_3-bare.dmap
|-- ParisPW_3-bare.gdml
|-- ParisPW_3.amap
|-- ParisPW_3.dmap
|-- ParisPW_3.gdml
|-- aChateau2Crystal.amap
|-- aChateau2Crystal.dmap
|-- aChateau2Crystal.gdml
|-- aChateau2Crystal_bare.amap
|-- aChateau2Crystal_bare.dmap
`-- aChateau2Crystal_bare.gdml
```

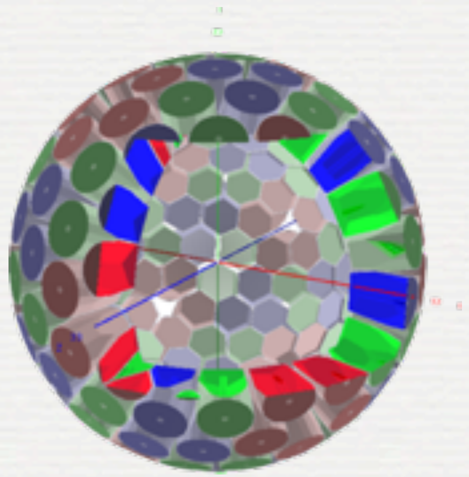
How to deal with geometries in SToGS factory ?

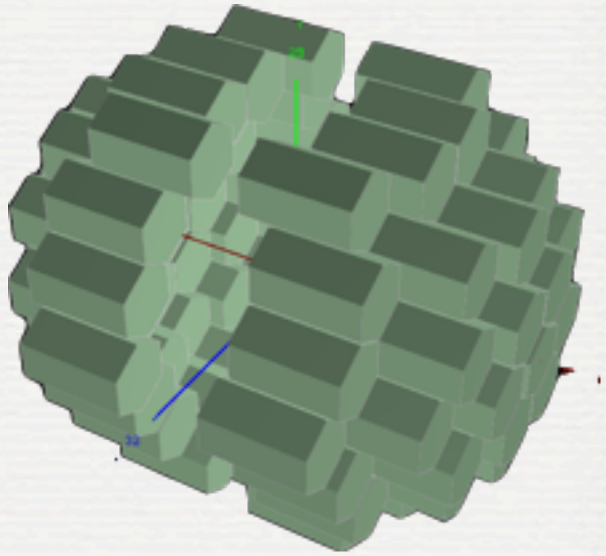
... and some of them ...

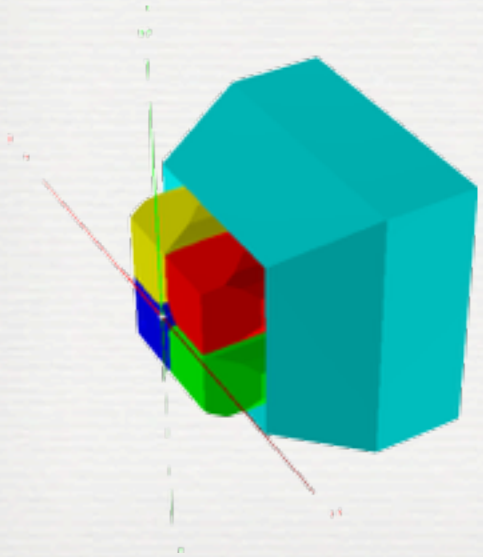


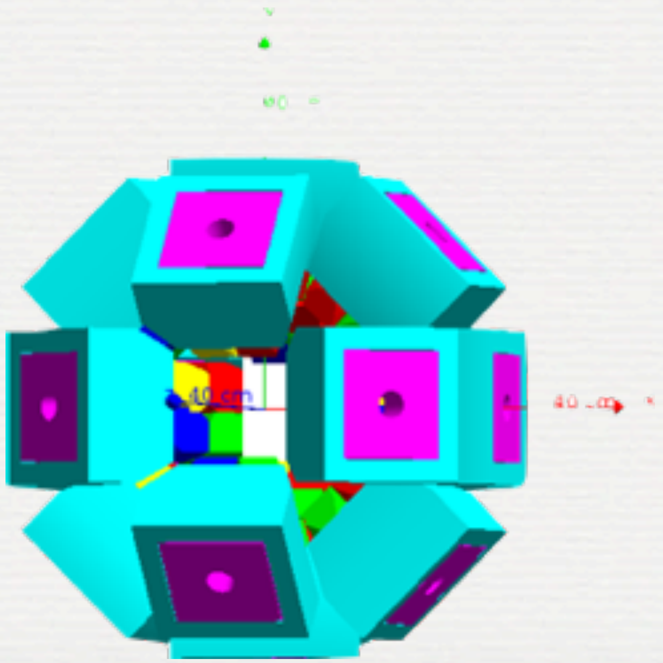






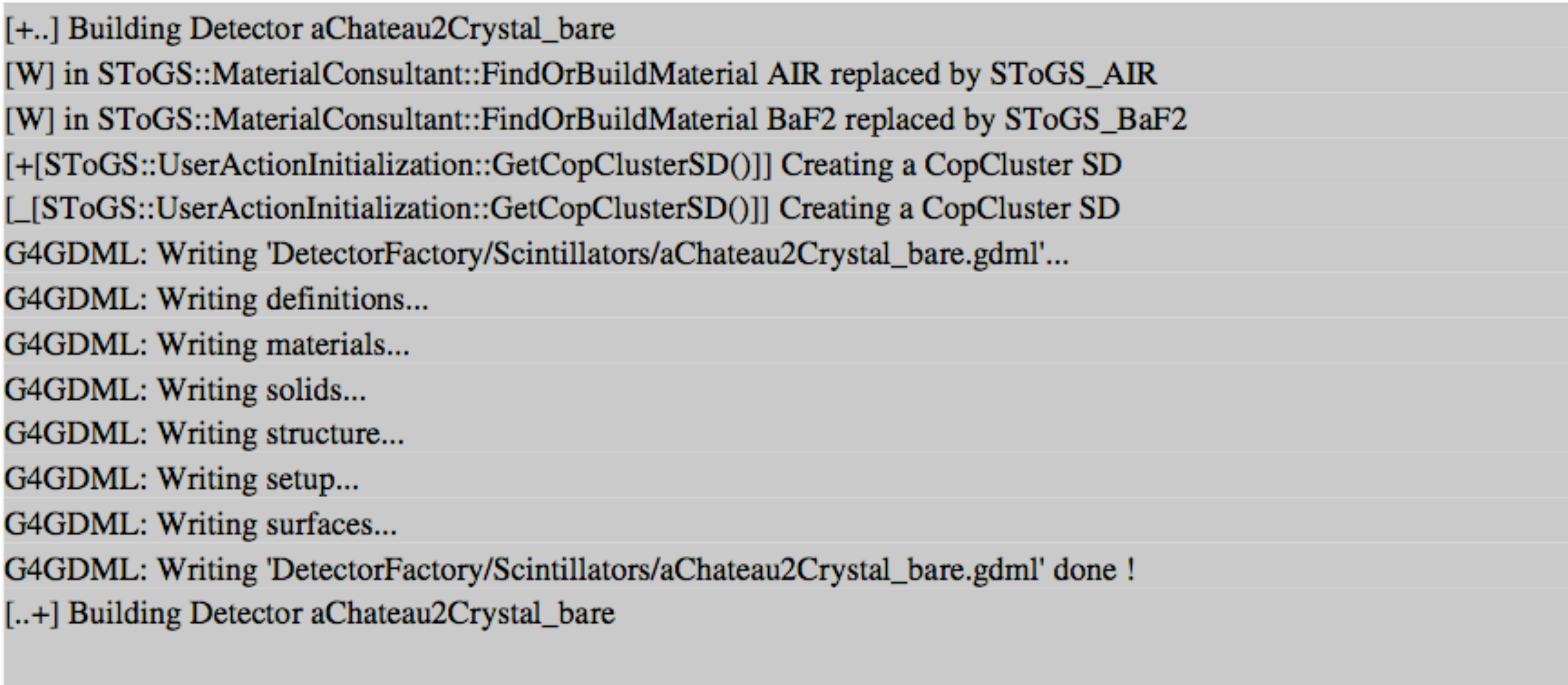






How to deal with **geometries** in SToGS factory ?

Example of one detector built first time SToGS_DetectorBuilder is called



```
[+..] Building Detector aChateau2Crystal_bare
[W] in SToGS::MaterialConsultant::FindOrBuildMaterial AIR replaced by SToGS_AIR
[W] in SToGS::MaterialConsultant::FindOrBuildMaterial BaF2 replaced by SToGS_BaF2
[+[SToGS::UserActionInitialization::GetCopClusterSD()]] Creating a CopCluster SD
[_[SToGS::UserActionInitialization::GetCopClusterSD()]] Creating a CopCluster SD
G4GDML: Writing 'DetectorFactory/Scintillators/aChateau2Crystal_bare.gdml'...
G4GDML: Writing definitions...
G4GDML: Writing materials...
G4GDML: Writing solids...
G4GDML: Writing structure...
G4GDML: Writing setup...
G4GDML: Writing surfaces...
G4GDML: Writing 'DetectorFactory/Scintillators/aChateau2Crystal_bare.gdml' done !
[..+] Building Detector aChateau2Crystal_bare
```

Note:

one script *CleanUpFactory.py* provided to safely clean up the factory

How to deal with geometries in SToGS factory ?

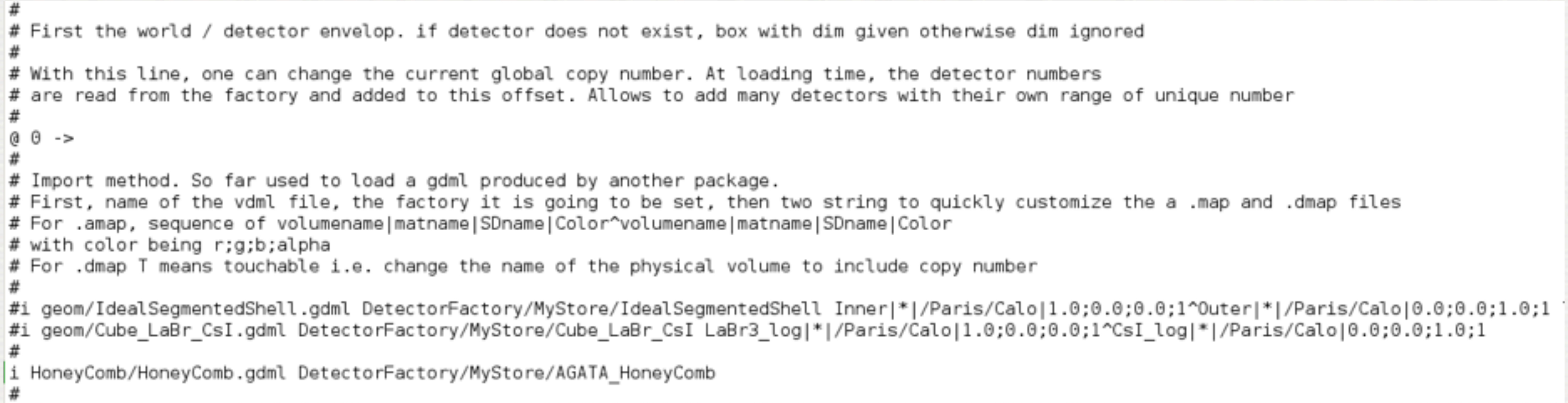
- SToGS_DetectorBuilder to help building a setup through a configuration XXXX.dfb *
- SToGS_DetectorBuilder to set mapping / sensitivity
- SToGS_DetectorBuilder to save the setup in myStore

Commands:

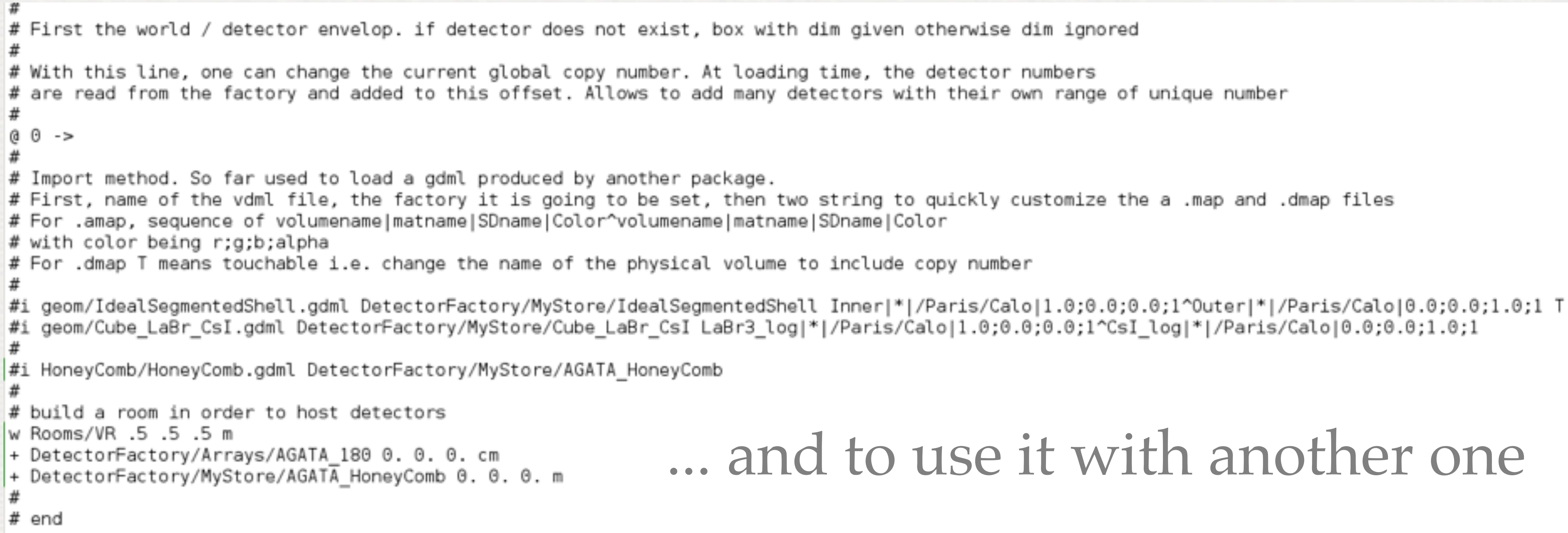
```
Commands:
  in terminal
          SToGS_DetectorBuilder -dfb my.dfb
  in G4 session
          DetectoryFactory/Save MySetup
```

* The default file is default.dfb

Ex of .dfb files. To import a pure gdml file ...



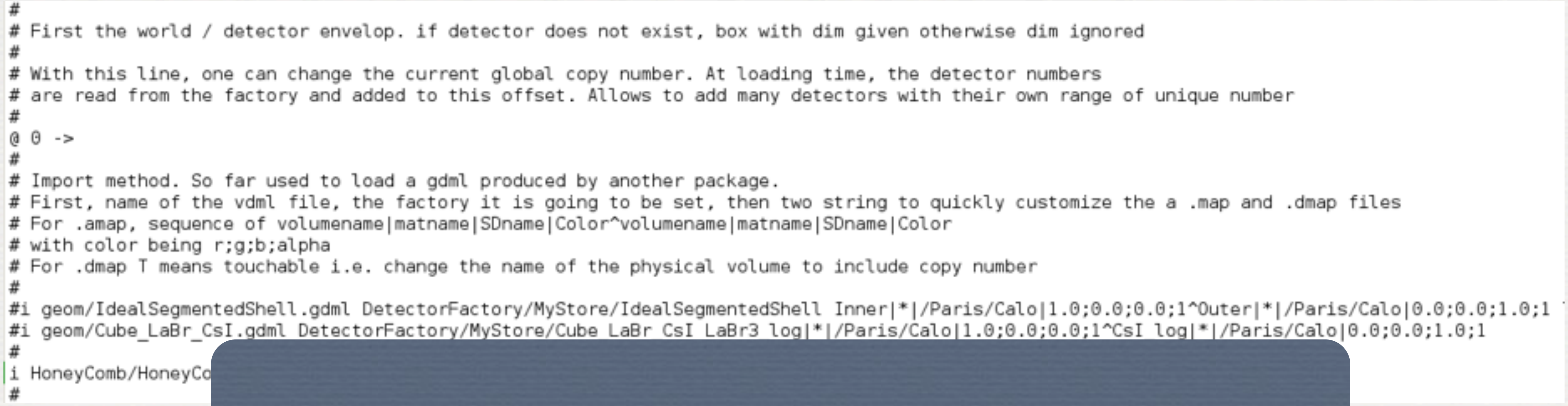
```
#
# First the world / detector envelop. if detector does not exist, box with dim given otherwise dim ignored
#
# With this line, one can change the current global copy number. At loading time, the detector numbers
# are read from the factory and added to this offset. Allows to add many detectors with their own range of unique number
#
@ 0 ->
#
# Import method. So far used to load a gdml produced by another package.
# First, name of the vdml file, the factory it is going to be set, then two string to quickly customize the a .map and .dmap files
# For .amap, sequence of volumename|matname|SDname|Color^volumename|matname|SDname|Color
# with color being r;g;b;alpha
# For .dmap T means touchable i.e. change the name of the physical volume to include copy number
#
#i geom/IdealSegmentedShell.gdml DetectorFactory/MyStore/IdealSegmentedShell Inner|*|/Paris/Calo|1.0;0.0;0.0;1^Outer|*|/Paris/Calo|0.0;0.0;1.0;1
#i geom/Cube_LaBr_CsI.gdml DetectorFactory/MyStore/Cube_LaBr_CsI LaBr3_log|*|/Paris/Calo|1.0;0.0;0.0;1^CsI_log|*|/Paris/Calo|0.0;0.0;1.0;1
#
i HoneyComb/HoneyComb.gdml DetectorFactory/MyStore/AGATA_HoneyComb
#
```



```
#
# First the world / detector envelop. if detector does not exist, box with dim given otherwise dim ignored
#
# With this line, one can change the current global copy number. At loading time, the detector numbers
# are read from the factory and added to this offset. Allows to add many detectors with their own range of unique number
#
@ 0 ->
#
# Import method. So far used to load a gdml produced by another package.
# First, name of the vdml file, the factory it is going to be set, then two string to quickly customize the a .map and .dmap files
# For .amap, sequence of volumename|matname|SDname|Color^volumename|matname|SDname|Color
# with color being r;g;b;alpha
# For .dmap T means touchable i.e. change the name of the physical volume to include copy number
#
#i geom/IdealSegmentedShell.gdml DetectorFactory/MyStore/IdealSegmentedShell Inner|*|/Paris/Calo|1.0;0.0;0.0;1^Outer|*|/Paris/Calo|0.0;0.0;1.0;1 T
#i geom/Cube_LaBr_CsI.gdml DetectorFactory/MyStore/Cube_LaBr_CsI LaBr3_log|*|/Paris/Calo|1.0;0.0;0.0;1^CsI_log|*|/Paris/Calo|0.0;0.0;1.0;1
#
#i HoneyComb/HoneyComb.gdml DetectorFactory/MyStore/AGATA_HoneyComb
#
# build a room in order to host detectors
w Rooms/VR .5 .5 .5 m
+ DetectorFactory/Arrays/AGATA_180 0. 0. 0. cm
+ DetectorFactory/MyStore/AGATA_HoneyComb 0. 0. 0. m
#
# end
```

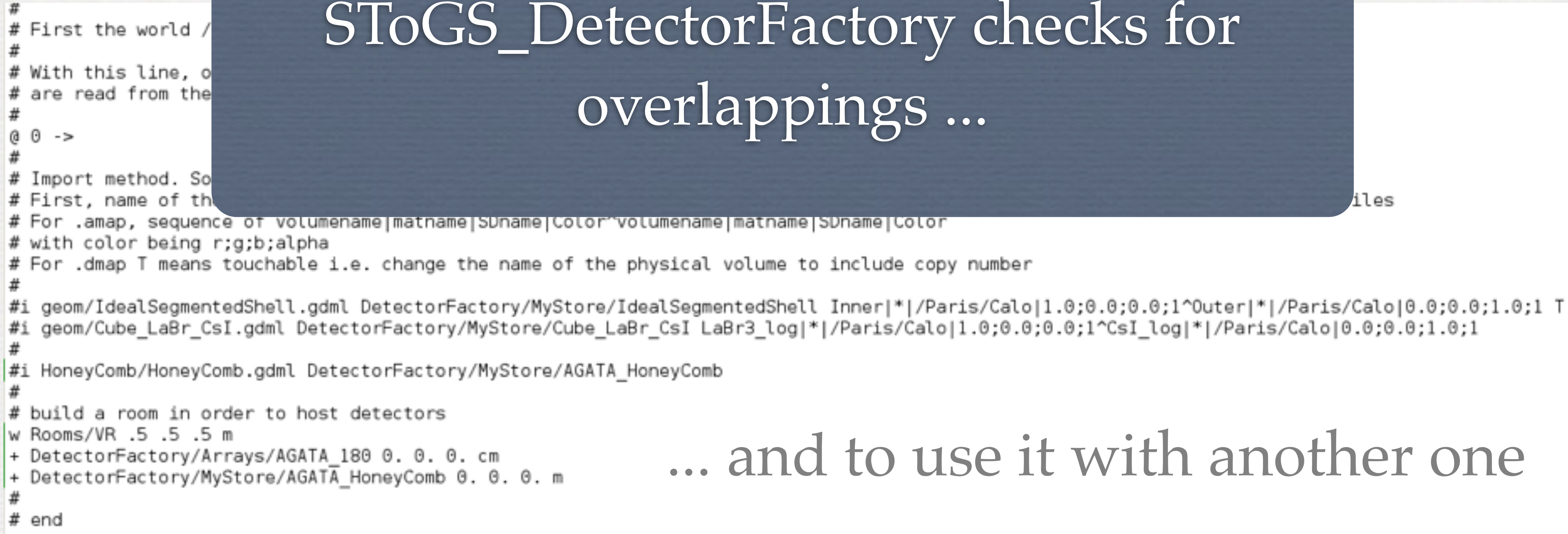
... and to use it with another one

Ex of .dfb files. To import a pure gdml file ...



```
#
# First the world / detector envelop. if detector does not exist, box with dim given otherwise dim ignored
#
# With this line, one can change the current global copy number. At loading time, the detector numbers
# are read from the factory and added to this offset. Allows to add many detectors with their own range of unique number
#
@ 0 ->
#
# Import method. So far used to load a gdml produced by another package.
# First, name of the vdml file, the factory it is going to be set, then two string to quickly customize the a .map and .dmap files
# For .amap, sequence of volumename|matname|SDname|Color^volumename|matname|SDname|Color
# with color being r;g;b;alpha
# For .dmap T means touchable i.e. change the name of the physical volume to include copy number
#
#i geom/IdealSegmentedShell.gdml DetectorFactory/MyStore/IdealSegmentedShell Inner|*|/Paris/Calo|1.0;0.0;0.0;1^Outer|*|/Paris/Calo|0.0;0.0;1.0;1
#i geom/Cube_LaBr_CsI.gdml DetectorFactory/MyStore/Cube LaBr CsI LaBr3 log|*|/Paris/Calo|1.0;0.0;0.0;1^CsI log|*|/Paris/Calo|0.0;0.0;1.0;1
#
i HoneyComb/HoneyCo
#
```

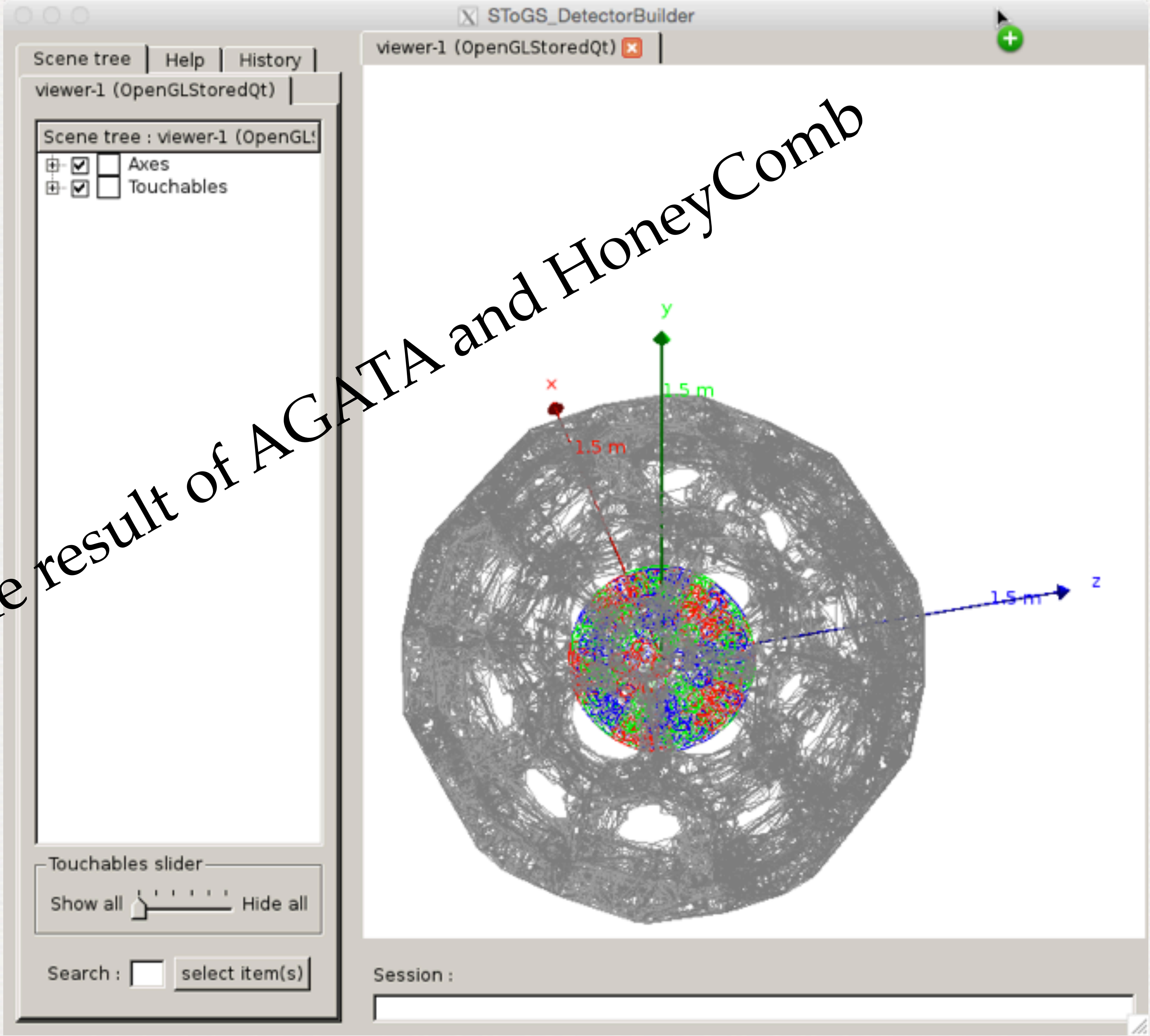
SToGS_DetectorFactory checks for overlappings ...



```
#
# First the world /
#
# With this line, o
# are read from the
#
@ 0 ->
#
# Import method. So
# First, name of th                                                                                                          iles
# For .amap, sequence of volumename|matname|SDname|Color^volumename|matname|SDname|Color
# with color being r;g;b;alpha
# For .dmap T means touchable i.e. change the name of the physical volume to include copy number
#
#i geom/IdealSegmentedShell.gdml DetectorFactory/MyStore/IdealSegmentedShell Inner|*|/Paris/Calo|1.0;0.0;0.0;1^Outer|*|/Paris/Calo|0.0;0.0;1.0;1 T
#i geom/Cube_LaBr_CsI.gdml DetectorFactory/MyStore/Cube_LaBr_CsI LaBr3_log|*|/Paris/Calo|1.0;0.0;0.0;1^CsI_log|*|/Paris/Calo|0.0;0.0;1.0;1
#
#i HoneyComb/HoneyComb.gdml DetectorFactory/MyStore/AGATA_HoneyComb
#
# build a room in order to host detectors
w Rooms/VR .5 .5 .5 m
+ DetectorFactory/Arrays/AGATA_180 0. 0. 0. cm
+ DetectorFactory/MyStore/AGATA_HoneyComb 0. 0. 0. m
#
# end
```

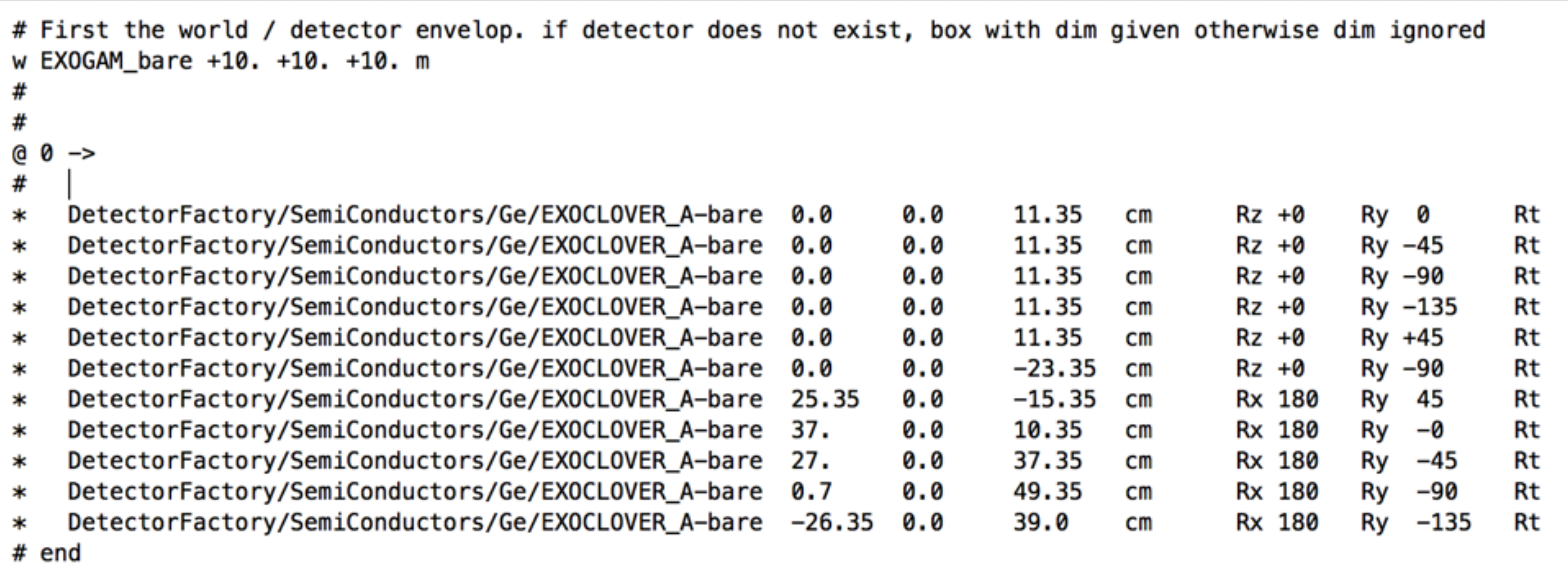
... and to use it with another one

How to deal with **geometries** in SToGS factory ?



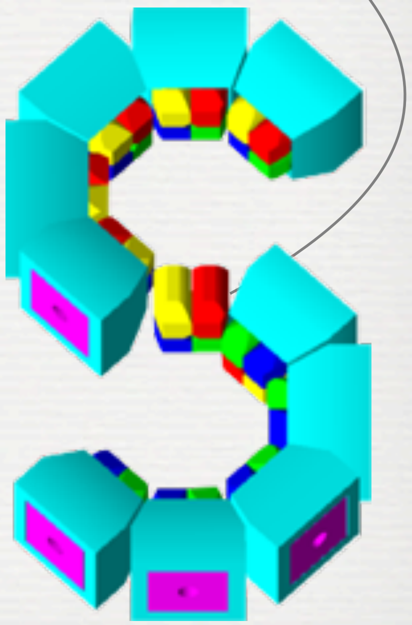
Here is the result of AGATA and HoneyComb

Another .dfb file



```
# First the world / detector envelop. if detector does not exist, box with dim given otherwise dim ignored
w EXOGAM_bare +10. +10. +10. m
#
#
@ 0 ->
#   |
*   DetectorFactory/SemiConductors/Ge/EXOCLOVER_A-bare  0.0     0.0     11.35   cm      Rz +0    Ry  0      Rt
*   DetectorFactory/SemiConductors/Ge/EXOCLOVER_A-bare  0.0     0.0     11.35   cm      Rz +0    Ry -45     Rt
*   DetectorFactory/SemiConductors/Ge/EXOCLOVER_A-bare  0.0     0.0     11.35   cm      Rz +0    Ry -90     Rt
*   DetectorFactory/SemiConductors/Ge/EXOCLOVER_A-bare  0.0     0.0     11.35   cm      Rz +0    Ry -135    Rt
*   DetectorFactory/SemiConductors/Ge/EXOCLOVER_A-bare  0.0     0.0     11.35   cm      Rz +0    Ry +45     Rt
*   DetectorFactory/SemiConductors/Ge/EXOCLOVER_A-bare  0.0     0.0     -23.35  cm      Rz +0    Ry -90     Rt
*   DetectorFactory/SemiConductors/Ge/EXOCLOVER_A-bare  25.35   0.0     -15.35  cm      Rx 180   Ry  45     Rt
*   DetectorFactory/SemiConductors/Ge/EXOCLOVER_A-bare  37.     0.0     10.35   cm      Rx 180   Ry  -0     Rt
*   DetectorFactory/SemiConductors/Ge/EXOCLOVER_A-bare  27.     0.0     37.35   cm      Rx 180   Ry  -45    Rt
*   DetectorFactory/SemiConductors/Ge/EXOCLOVER_A-bare  0.7     0.0     49.35   cm      Rx 180   Ry  -90    Rt
*   DetectorFactory/SemiConductors/Ge/EXOCLOVER_A-bare  -26.35  0.0     39.0    cm      Rx 180   Ry  -135   Rt
# end
```

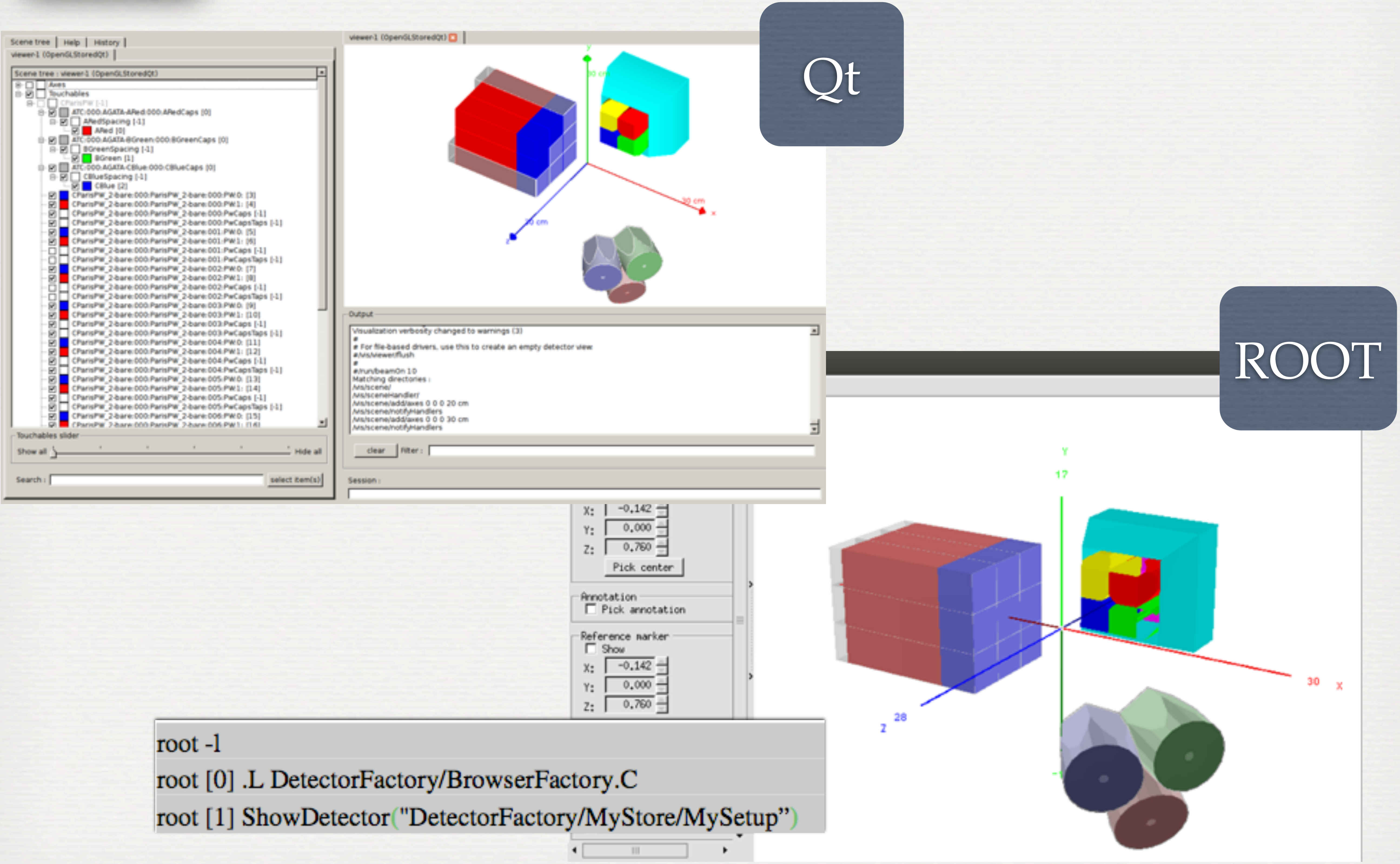
take a detector ... add it with translations / rotations ...



Ex to add a generic, the file MyTwoShells.geo contains the required informations

+ DetectorFactory/Generics/TwoShells$MyTwoShells.geo 0. 0. 0. m

How to deal with **geometries** in SToGS factory ?



Qt

Scene tree | Help | History

viewer-1 (OpenGLStoredQt)

Scene tree : viewer-1 (OpenGLStoredQt)

- ☐ ☐ Axes
- ☑ ☐ Touchables
  - ☐ ☐ CParisPW [-1]
    - ☑ ATC:000:AGATA-ARed:000:ARedCaps [0]
      - ☑ ☐ ARedSpacing [-1]
        - ☑ 🟥 ARed [0]
    - ☑ ATC:000:AGATA-BGreen:000:BGreenCaps [0]
      - ☑ ☐ BGreenSpacing [-1]
        - ☑ 🟩 BGreen [1]
    - ☑ ATC:000:AGATA-CBlue:000:CBlueCaps [0]
      - ☑ ☐ CBlueSpacing [-1]
        - ☑ 🟦 CBlue [2]
    - ☑ 🟦 CParisPW_2-bare:000:ParisPW_2-bare:000:PW0: [3]
    - ☑ 🟥 CParisPW_2-bare:000:ParisPW_2-bare:000:PW1: [4]
    - ☑ ☐ CParisPW_2-bare:000:ParisPW_2-bare:000:PwCaps [-1]
    - ☑ ☐ CParisPW_2-bare:000:ParisPW_2-bare:000:PwCapsTaps [-1]
    - ☑ 🟦 CParisPW_2-bare:000:ParisPW_2-bare:001:PW0: [5]
    - ☑ 🟥 CParisPW_2-bare:000:ParisPW_2-bare:001:PW1: [6]
    - ☐ ☐ CParisPW_2-bare:000:ParisPW_2-bare:001:PwCaps [-1]
    - ☐ ☐ CParisPW_2-bare:000:ParisPW_2-bare:001:PwCapsTaps [-1]
    - ☑ 🟦 CParisPW_2-bare:000:ParisPW_2-bare:002:PW0: [7]
    - ☑ 🟥 CParisPW_2-bare:000:ParisPW_2-bare:002:PW1: [8]
    - ☐ ☐ CParisPW_2-bare:000:ParisPW_2-bare:002:PwCaps [-1]
    - ☐ ☐ CParisPW_2-bare:000:ParisPW_2-bare:002:PwCapsTaps [-1]
    - ☑ 🟦 CParisPW_2-bare:000:ParisPW_2-bare:003:PW0: [9]
    - ☑ 🟥 CParisPW_2-bare:000:ParisPW_2-bare:003:PW1: [10]
    - ☑ ☐ CParisPW_2-bare:000:ParisPW_2-bare:003:PwCaps [-1]
    - ☑ ☐ CParisPW_2-bare:000:ParisPW_2-bare:003:PwCapsTaps [-1]
    - ☑ 🟦 CParisPW_2-bare:000:ParisPW_2-bare:004:PW0: [11]
    - ☑ 🟥 CParisPW_2-bare:000:ParisPW_2-bare:004:PW1: [12]
    - ☑ ☐ CParisPW_2-bare:000:ParisPW_2-bare:004:PwCaps [-1]
    - ☑ ☐ CParisPW_2-bare:000:ParisPW_2-bare:004:PwCapsTaps [-1]
    - ☑ 🟦 CParisPW_2-bare:000:ParisPW_2-bare:005:PW0: [13]
    - ☑ 🟥 CParisPW_2-bare:000:ParisPW_2-bare:005:PW1: [14]
    - ☑ ☐ CParisPW_2-bare:000:ParisPW_2-bare:005:PwCaps [-1]
    - ☑ ☐ CParisPW_2-bare:000:ParisPW_2-bare:005:PwCapsTaps [-1]
    - ☑ 🟦 CParisPW_2-bare:000:ParisPW_2-bare:006:PW0: [15]
    - ☑ 🟥 CParisPW_2-bare:000:ParisPW_2-bare:006:PW1: [16]

Touchables slider

Show all — Hide all

Search :

select item(s)

viewer-1 (OpenGLStoredQt)

Output

```
Visualization verbosity changed to warnings (3)
#
# For file-based drivers, use this to create an empty detector view:
#/vis/viewer/flush
#
#/run/beamOn 10
Matching directories :
/vis/scene/
/vis/sceneHandler/
/vis/scene/add/axes 0 0 0 20 cm
/vis/scene/notifyHandlers
/vis/scene/add/axes 0 0 0 30 cm
/vis/scene/notifyHandlers
```

clear   Filter :

Session :

ROOT

X: -0,142
Y: 0,000
Z: 0,760

Pick center

Annotation

☐ Pick annotation

Reference marker

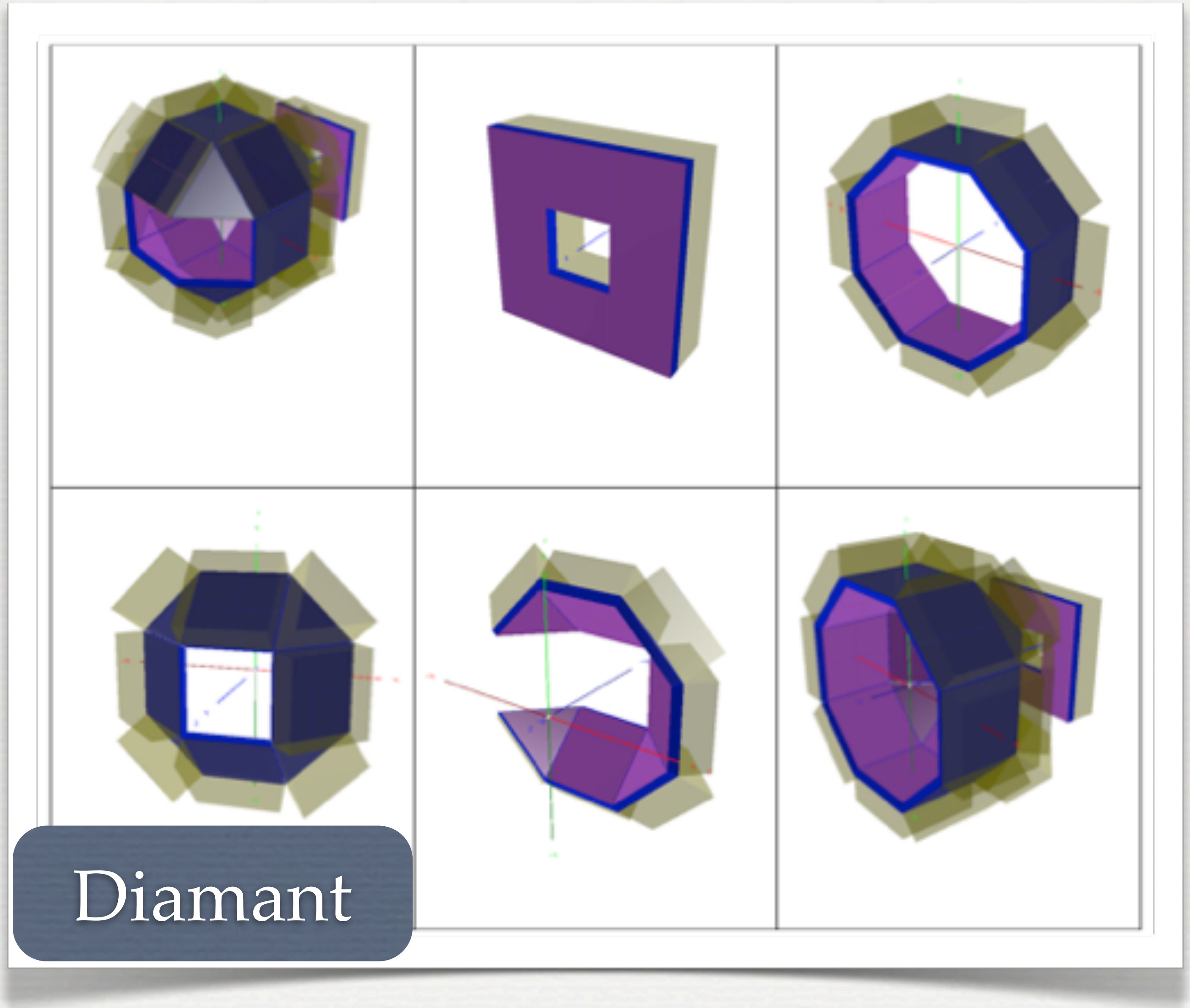
☐ Show

X: -0,142
Y: 0,000
Z: 0,760

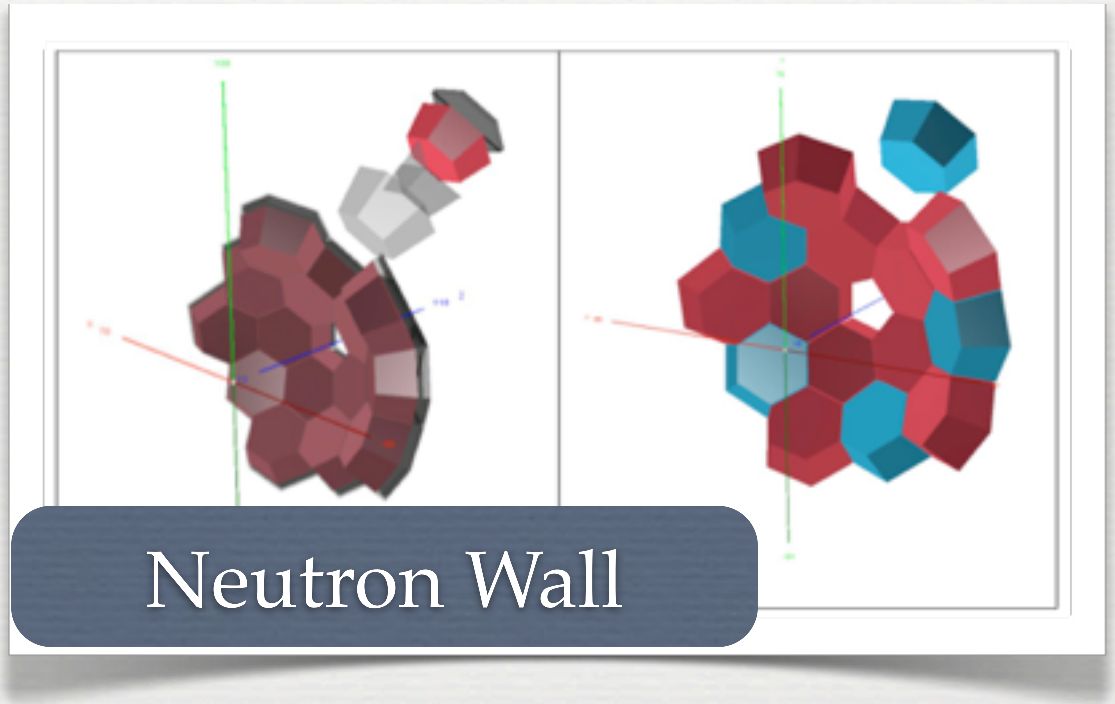
```
root -l
root [0] .L DetectorFactory/BrowserFactory.C
root [1] ShowDetector("DetectorFactory/MyStore/MySetup")
```

How to deal with **geometries** in SToGS factory ?

Some detectors imported through gdml



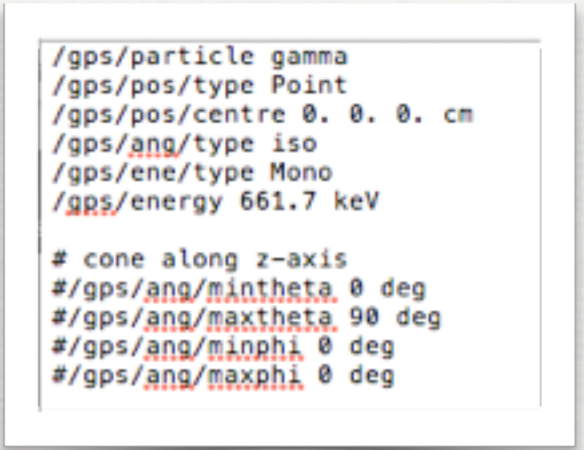
- gdml, not a magic tool
  - Pb in converting some shapes
    - not all G4 supported / different in root
  - Pb with names !
    - many C++ geometries give the same name for things
- But in gdml
  - a name is used almost as pointer ...
  - should be unique !



How to deal with **Primaries** in SToGS ?

GPS: standard G4, it uses G4 macro language

➥ it thus uses distributions to randomly produce events

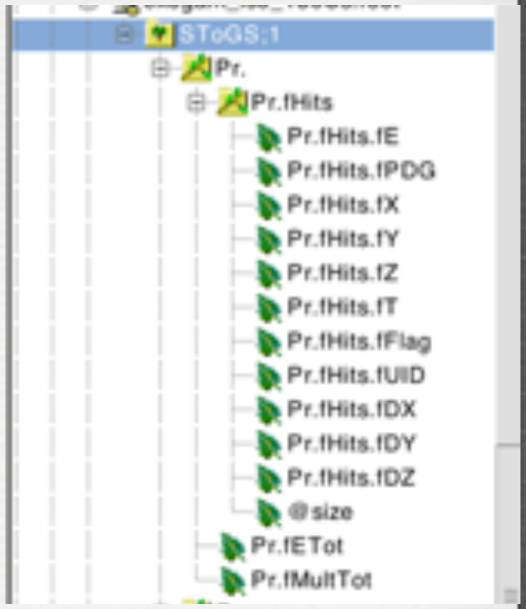


```
/gps/particle gamma
/gps/pos/type Point
/gps/pos/centre 0. 0. 0. cm
/gps/ang/type iso
/gps/ene/type Mono
/gps/energy 661.7 keV

# cone along z-axis
#/gps/ang/mintheta 0 deg
#/gps/ang/maxtheta 90 deg
#/gps/ang/minphi 0 deg
#/gps/ang/maxphi 0 deg
```

toROOTGPS* extracts **Primaries** from a ROOTTree

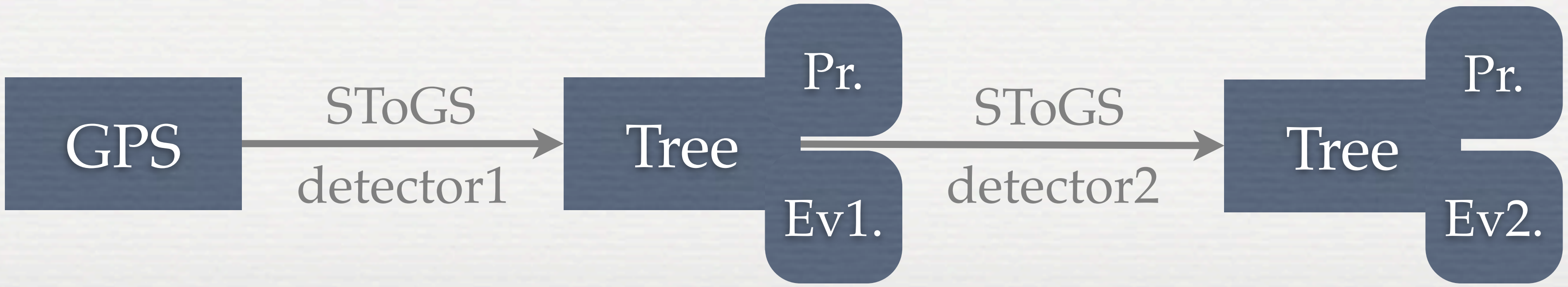
➥ multi-thread oriented ... each thread reads a list of ROOT files



```
SToGS;1
  Pr.
    Pr.fHits
      Pr.fHits.fE
      Pr.fHits.fPDG
      Pr.fHits.fX
      Pr.fHits.fY
      Pr.fHits.fZ
      Pr.fHits.fT
      Pr.fHits.fFlag
      Pr.fHits.fUID
      Pr.fHits.fDX
      Pr.fHits.fDY
      Pr.fHits.fDZ
      @size
    Pr.fETot
    Pr.fMultTot
```

Note: The primaries are saved in the SToGS output Tree

One can do then:



* SToGS tools

How to run the ***SToGS_Source*** application ?

While *SToGS_DetectorBuilder* is used to build the setup, *SToGS_Source* performed the simulation

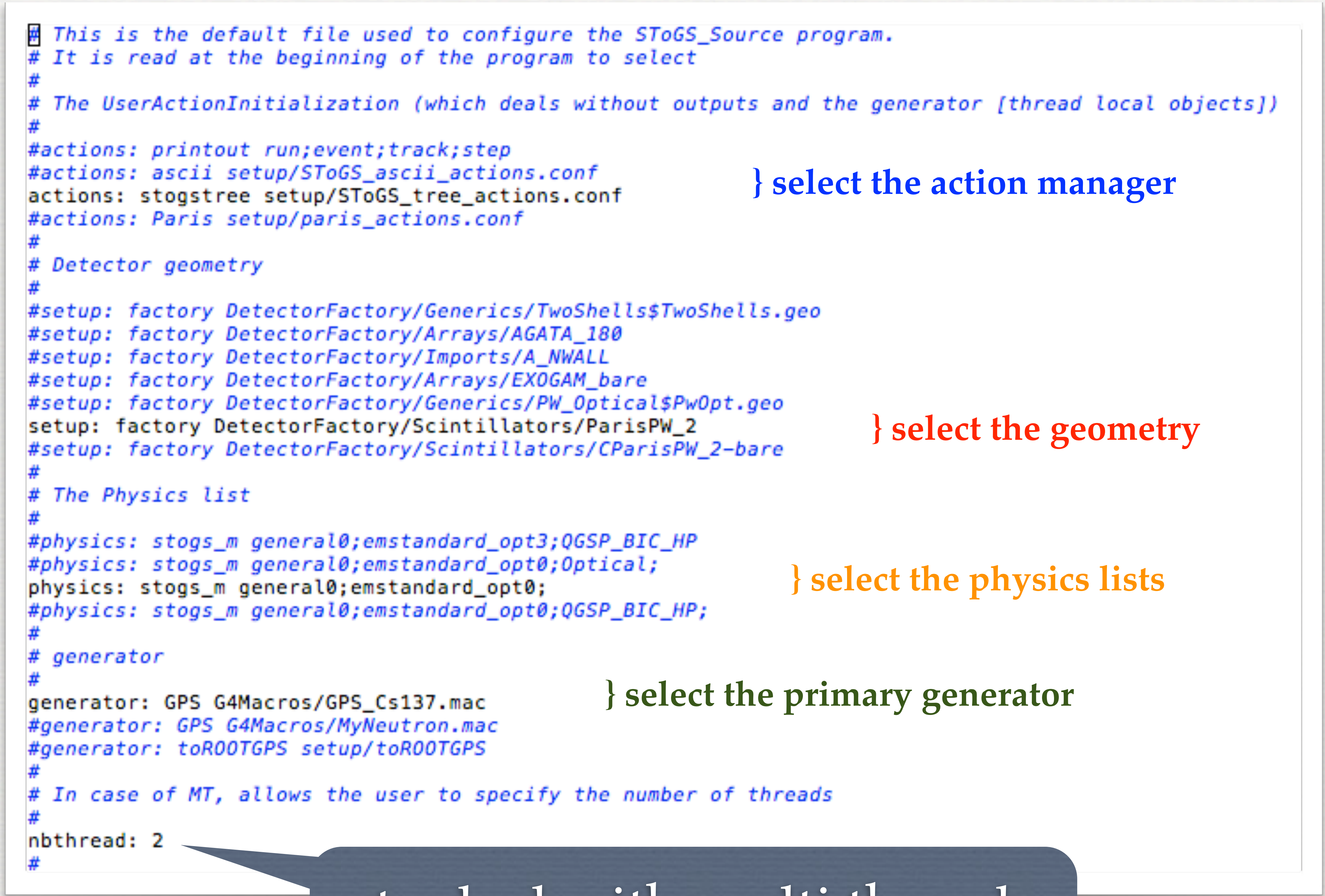
It requires an ascii file to select the 4 main bricks

```
Commands:
        SToGS_Source -c conffile -b g4batchmacro.mac
or
        SToGS_Source -c conffile
or
        SToGS_Source
```

} Interactive session

means conffile is setup/SToGS_Source.conf

How to run the *SToGS_Source* application ?



```
# This is the default file used to configure the SToGS_Source program.
# It is read at the beginning of the program to select
#
# The UserActionInitialization (which deals without outputs and the generator [thread local objects])
#
#actions: printout run;event;track;step
#actions: ascii setup/SToGS_ascii_actions.conf
actions: stogstree setup/SToGS_tree_actions.conf
#actions: Paris setup/paris_actions.conf
#
# Detector geometry
#
#setup: factory DetectorFactory/Generics/TwoShells$TwoShells.geo
#setup: factory DetectorFactory/Arrays/AGATA_180
#setup: factory DetectorFactory/Imports/A_NWALL
#setup: factory DetectorFactory/Arrays/EXOGAM_bare
#setup: factory DetectorFactory/Generics/PW_Optical$PwOpt.geo
setup: factory DetectorFactory/Scintillators/ParisPW_2
#setup: factory DetectorFactory/Scintillators/CParisPW_2-bare
#
# The Physics list
#
#physics: stogs_m general0;emstandard_opt3;QGSP_BIC_HP
#physics: stogs_m general0;emstandard_opt0;Optical;
physics: stogs_m general0;emstandard_opt0;
#physics: stogs_m general0;emstandard_opt0;QGSP_BIC_HP;
#
# generator
#
generator: GPS G4Macros/GPS_Cs137.mac
#generator: GPS G4Macros/MyNeutron.mac
#generator: toROOTGPS setup/toROOTGPS
#
# In case of MT, allows the user to specify the number of threads
#
nbthread: 2
#
```

} select the action manager

} select the geometry

} select the physics lists

} select the primary generator

to deal with multi thread

How to plug externals in ***SToGS*** ?

To deal with ROOT / Trees and SToGS events

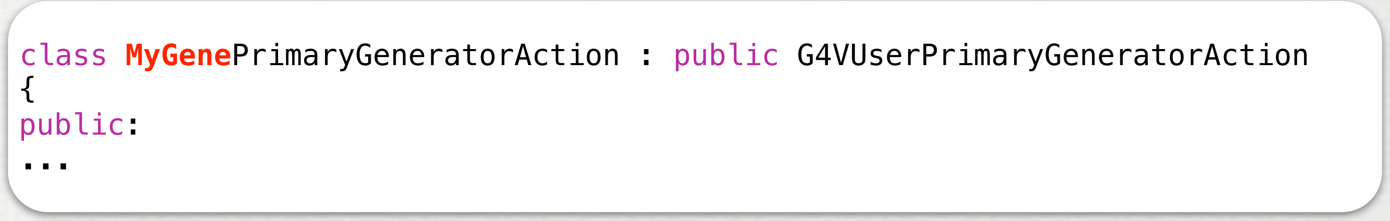
- ROOT should be operational
- go to analysis/SToGS and type make
- you should get libSToGS_BaseROOTEvents.so
- then during cmake, SToGS detects and built things

Standard modules can be plugged*

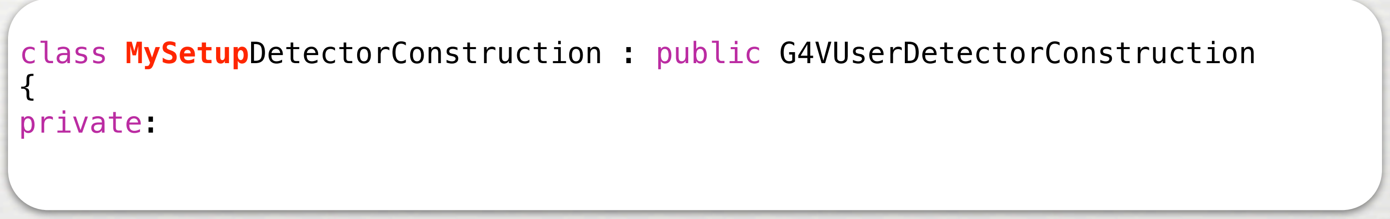
see SToGS_MY_Container/README

General commands

```
cmake -DGeant4_DIR=/PATH/TO/GEANT4DIR/
      -DMY_DIR=./SToGS_MY_Container/My_Customized_in_SToGS
      -DMY_PRI=MyGene
      -DMY_DET=MySetup
      ../
```



```
class MyGenePrimaryGeneratorAction : public G4VUserPrimaryGeneratorAction
{
public:
...
```



```
class MySetupDetectorConstruction : public G4VUserDetectorConstruction
{
private:
```

*geometries/primary generator

How to plug externals in *SToGS* ?

To

R

g

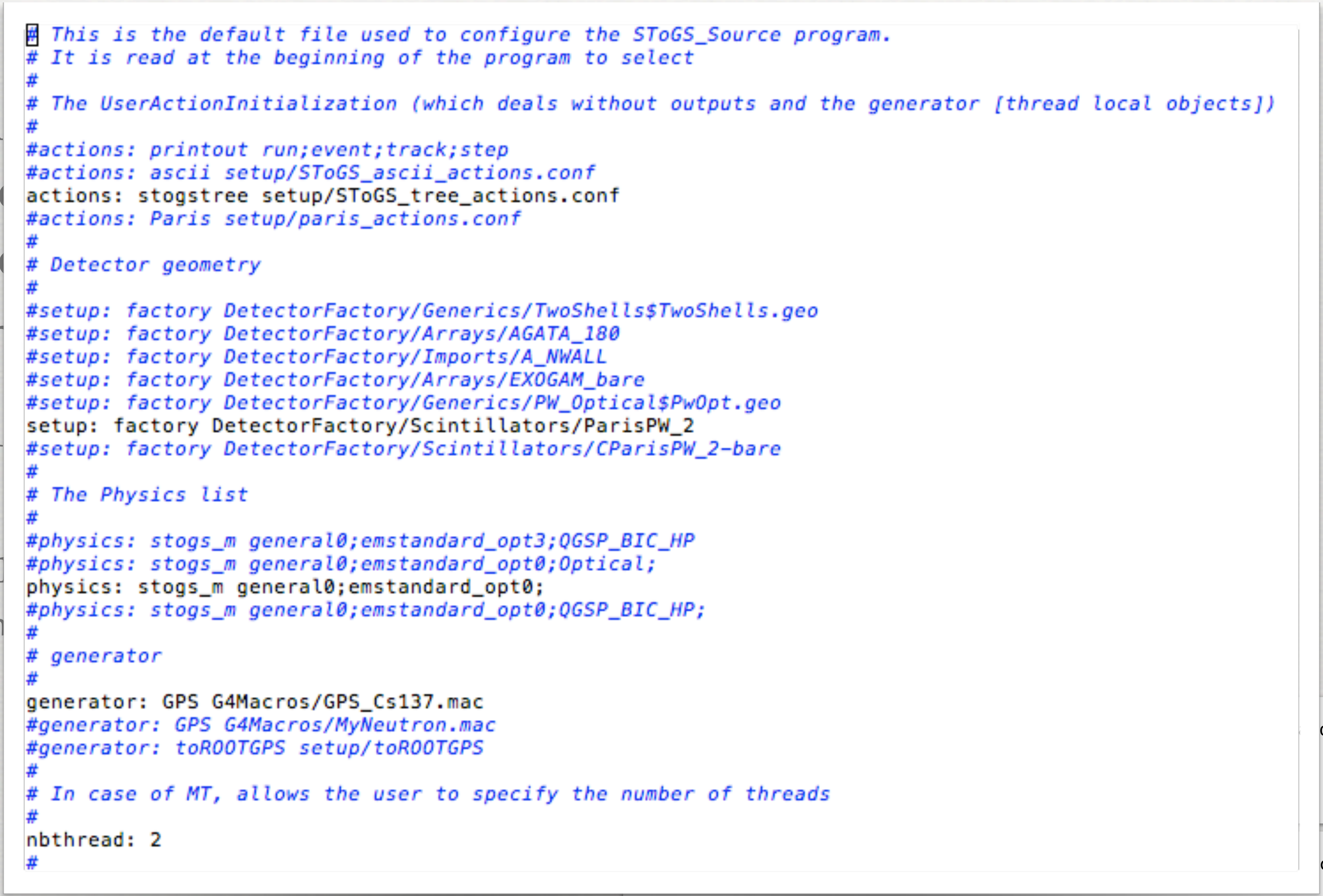
y

th

Sta

Ge

cm



```
# This is the default file used to configure the SToGS_Source program.
# It is read at the beginning of the program to select
#
# The UserActionInitialization (which deals without outputs and the generator [thread local objects])
#
#actions: printout run;event;track;step
#actions: ascii setup/SToGS_ascii_actions.conf
actions: stogstree setup/SToGS_tree_actions.conf
#actions: Paris setup/paris_actions.conf
#
# Detector geometry
#
#setup: factory DetectorFactory/Generics/TwoShells$TwoShells.geo
#setup: factory DetectorFactory/Arrays/AGATA_180
#setup: factory DetectorFactory/Imports/A_NWALL
#setup: factory DetectorFactory/Arrays/EXOGAM_bare
#setup: factory DetectorFactory/Generics/PW_Optical$PwOpt.geo
setup: factory DetectorFactory/Scintillators/ParisPW_2
#setup: factory DetectorFactory/Scintillators/CParisPW_2-bare
#
# The Physics list
#
#physics: stogs_m general0;emstandard_opt3;QGSP_BIC_HP
#physics: stogs_m general0;emstandard_opt0;Optical;
physics: stogs_m general0;emstandard_opt0;
#physics: stogs_m general0;emstandard_opt0;QGSP_BIC_HP;
#
# generator
#
generator: GPS G4Macros/GPS_Cs137.mac
#generator: GPS G4Macros/MyNeutron.mac
#generator: toROOTGPS setup/toROOTGPS
#
# In case of MT, allows the user to specify the number of threads
#
nbthread: 2
#
```

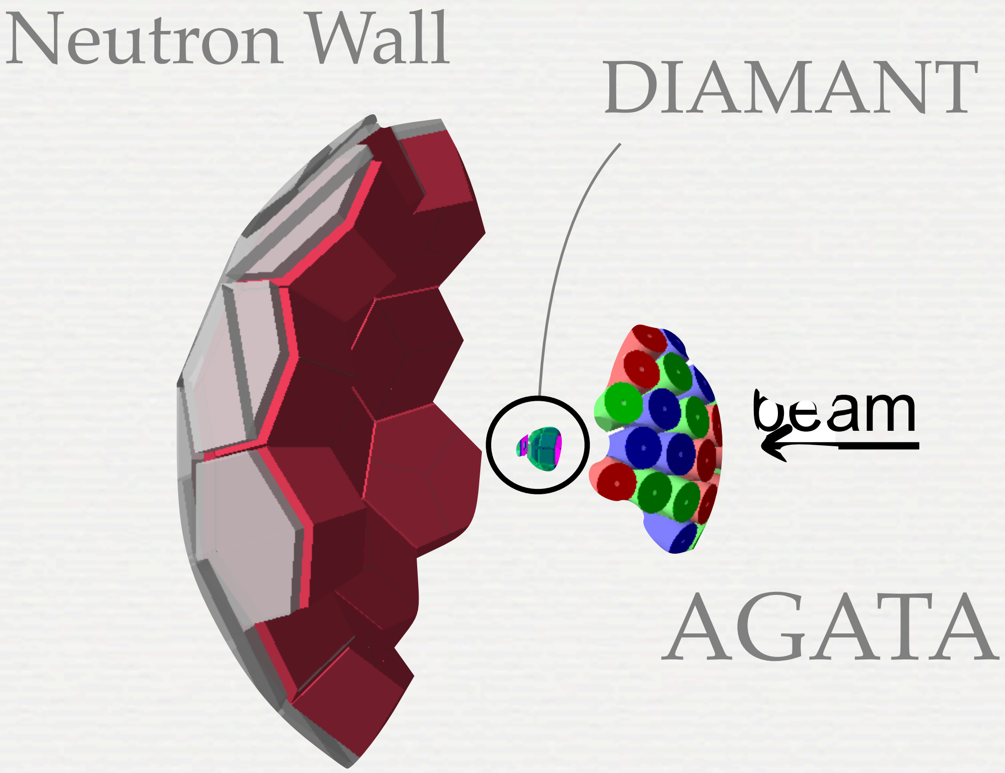
```
orAction
```

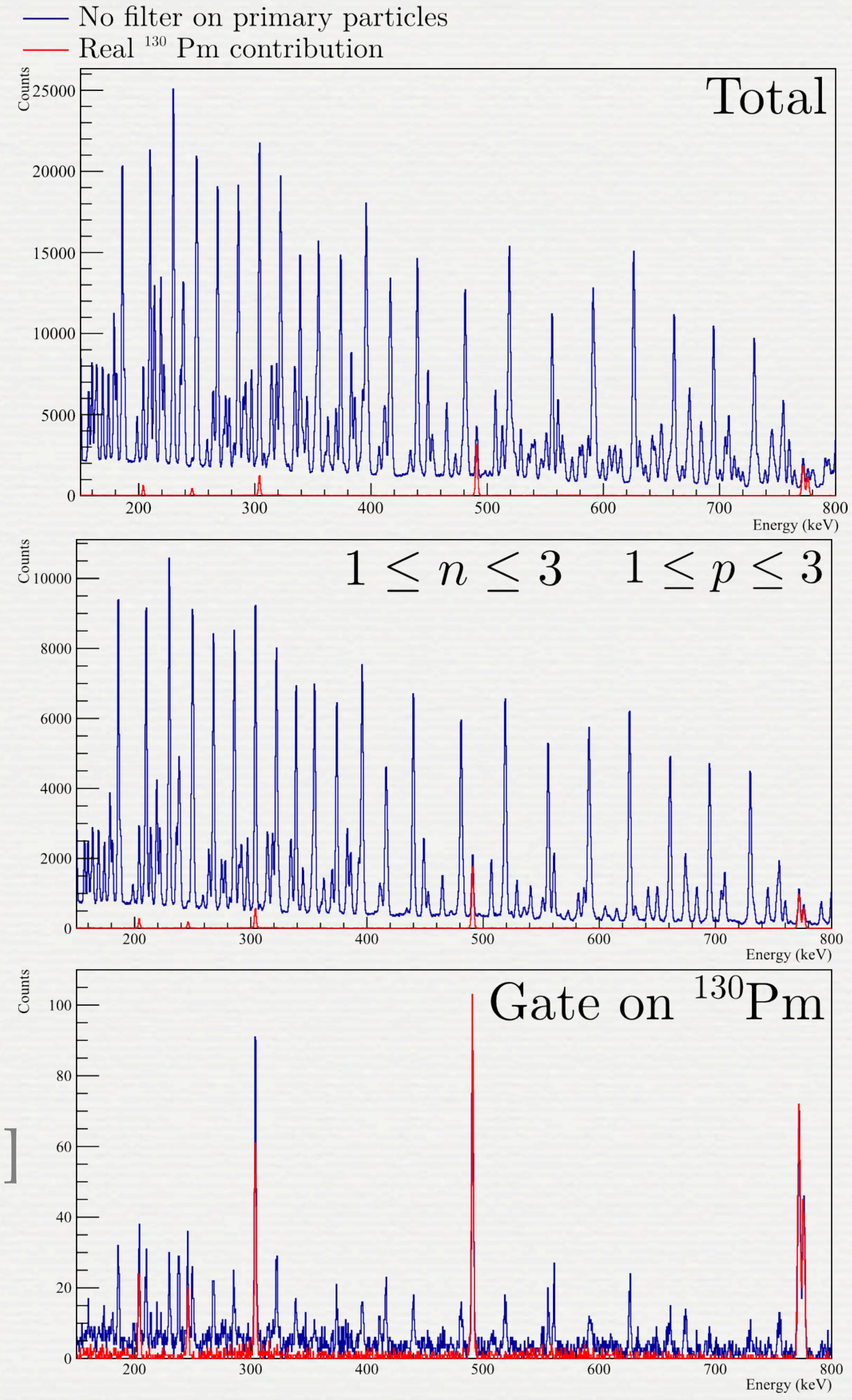
```
ction
```

```
private:
```

*geometries/primary generator

An example of what could be done ...





neutrons / protons / alpha

from energy/angular distributions [pace]

gamma

from level scheme + cross sections

Conclusions / perspectives

STоGS developed to:

- deal with setups ‘easily’
- to be as much as possible open and connected
  to other simulation packages
  to analysis packages
- Still a young package (growing)

Future:

- more stability
- exchanged of geometries - ICC common objective
- toward a mixed GPS / toROOTGPS generator
  - ➥ i.e. primary variables from root branches + other from distributions
- put SToGS_Beam in action
- put hands in the physics of GEANT4