

Precalibrated Ion Beam Identification Detector Simulation

1.0

Generated by Doxygen 1.7.4

Fri Jun 15 2012 21:54:54

Contents

1	Main Page	1
1.1	Introduction	1
1.2	Installation	1
1.3	Compiling	1
1.4	Technical Details	1
1.5	Miscellaneous	2
2	Todo List	3
3	Class Index	5
3.1	Class List	5
4	File Index	7
4.1	File List	7
5	Class Documentation	9
5.1	AllAccumulatedStatistics Class Reference	9
5.1.1	Detailed Description	10
5.1.2	Constructor & Destructor Documentation	10
5.1.2.1	AllAccumulatedStatistics	10
5.1.3	Member Function Documentation	10
5.1.3.1	AddGeDetectorEvent	10
5.1.3.2	AddScintillatorPlateEvent	11
5.1.3.3	AddScintillatorTubeEvent	11
5.1.3.4	GetEventHits	12
5.1.3.5	timeCheck	12
5.2	PrimaryGeneratorMessenger::AllAccumulatedStatistics Class Reference	13

5.2.1	Detailed Description	14
5.2.2	Constructor & Destructor Documentation	14
5.2.2.1	AllAccumulatedStatistics	14
5.2.3	Member Function Documentation	14
5.2.3.1	AddGeDetectorEvent	14
5.2.3.2	AddScintillatorPlateEvent	14
5.2.3.3	AddScintillatorTubeEvent	15
5.2.3.4	GetEventHits	15
5.2.3.5	timeCheck	15
5.3	DetectorConstruction Class Reference	16
5.3.1	Detailed Description	24
5.3.2	Member Function Documentation	24
5.3.2.1	GetBackScintillatorPlateSizeY	24
5.3.2.2	SetBeamBlockerMaterial	24
5.3.2.3	SetBeamBlockerThickness	25
5.3.2.4	SetBetaFlipping	25
5.3.2.5	SetEffectiveSlowdownThickness	25
5.3.2.6	SetGapMaterial	26
5.3.2.7	SetGeDetectorLength	26
5.3.2.8	SetGeDetectorRadius	27
5.3.2.9	SetGeDetectorShellInnerRadius	27
5.3.2.10	SetGeDetectorShellLength	27
5.3.2.11	SetGeDetectorShellOuterRadius	28
5.3.2.12	SetGermaniumDetectorMaterial	28
5.3.2.13	SetNumberOfScintillatorBoxes	28
5.3.2.14	SetNumberOfScintillatorsPerBox	29
5.3.2.15	SetScintillatorDiameter	29
5.3.2.16	SetScintillatorMaterial	30
5.3.2.17	SetScintillatorShieldingGapThickness	30
5.3.2.18	SetShieldingBeamBlockerMaterial	30
5.3.2.19	SetShieldingBeamBlockerThickness	31
5.3.2.20	SetShieldingMaterial	31
5.3.2.21	SetShieldingThickness	32
5.3.2.22	SetSlowdownLength	32

5.3.2.23	SetSlowdownMaterial	32
5.3.2.24	SetSlowdownThickness	33
5.3.2.25	SetTopShieldingThickness	33
5.4	PrimaryGeneratorMessenger::DetectorConstruction Class Reference	34
5.4.1	Detailed Description	42
5.4.2	Member Function Documentation	42
5.4.2.1	GetBackScintillatorPlateSizeY	42
5.4.2.2	SetBeamBlockerThickness	42
5.4.2.3	SetEffectiveSlowdownThickness	43
5.4.2.4	SetGeDetectorLength	43
5.4.2.5	SetGeDetectorRadius	43
5.4.2.6	SetGeDetectorShellInnerRadius	43
5.4.2.7	SetGeDetectorShellLength	43
5.4.2.8	SetGeDetectorShellOuterRadius	44
5.4.2.9	SetShieldingBeamBlockerThickness	44
5.4.2.10	SetShieldingThickness	44
5.4.2.11	SetSlowdownLength	44
5.4.2.12	SetSlowdownThickness	44
5.4.2.13	SetTopShieldingThickness	45
5.5	DetectorMessenger Class Reference	45
5.5.1	Detailed Description	46
5.5.2	Constructor & Destructor Documentation	47
5.5.2.1	DetectorMessenger	47
5.5.3	Member Function Documentation	51
5.5.3.1	SetNewValue	51
5.6	PrimaryGeneratorMessenger::DetectorMessenger Class Reference	53
5.6.1	Detailed Description	55
5.7	EventAction Class Reference	55
5.7.1	Detailed Description	56
5.7.2	Constructor & Destructor Documentation	56
5.7.2.1	EventAction	56
5.7.3	Member Function Documentation	56
5.7.3.1	AddGeDetectorEvent	56
5.7.3.2	AddScintillatorPlateEvent	57

5.7.3.3	AddScintillatorTubeEvent	57
5.7.3.4	SetPrintModulo	58
5.8	EventActionMessenger Class Reference	58
5.8.1	Detailed Description	59
5.8.2	Constructor & Destructor Documentation	59
5.8.2.1	EventActionMessenger	59
5.8.3	Member Function Documentation	59
5.8.3.1	SetNewValue	59
5.9	EventHit Class Reference	60
5.9.1	Detailed Description	60
5.9.2	Constructor & Destructor Documentation	61
5.9.2.1	EventHit	61
5.10	PrimaryGeneratorMessenger::EventHit Class Reference	61
5.10.1	Detailed Description	62
5.10.2	Constructor & Destructor Documentation	62
5.10.2.1	EventHit	62
5.10.2.2	EventHit	62
5.11	G4EmStandardFysik Class Reference	63
5.11.1	Detailed Description	63
5.11.2	Constructor & Destructor Documentation	63
5.11.2.1	G4EmStandardFysik	63
5.12	MyPhysicsList< T > Class Template Reference	64
5.12.1	Detailed Description	64
5.12.2	Constructor & Destructor Documentation	64
5.12.2.1	MyPhysicsList	64
5.13	PrimaryGeneratorMessenger::PrimaryGeneratorAction Class Reference	65
5.13.1	Detailed Description	66
5.13.2	Member Function Documentation	66
5.13.2.1	SetParticle	66
5.14	PrimaryGeneratorAction Class Reference	67
5.14.1	Detailed Description	68
5.14.2	Constructor & Destructor Documentation	68
5.14.2.1	PrimaryGeneratorAction	68
5.14.3	Member Function Documentation	68

5.14.3.1	SetParticle	68
5.15	PrimaryGeneratorMessenger Class Reference	69
5.15.1	Detailed Description	70
5.15.2	Constructor & Destructor Documentation	71
5.15.2.1	PrimaryGeneratorMessenger	71
5.16	RunAction Class Reference	72
5.16.1	Detailed Description	73
5.16.2	Member Function Documentation	73
5.16.2.1	SetNewResultsFileName	73
5.16.2.2	SetNewVerboseResultsFileName	73
5.16.2.3	SetVerboseFileUsage	74
5.16.2.4	WriteResultsToFile	74
5.17	RunActionMessenger Class Reference	75
5.17.1	Detailed Description	76
5.17.2	Constructor & Destructor Documentation	76
5.17.2.1	RunActionMessenger	76
5.17.3	Member Function Documentation	77
5.17.3.1	SetNewValue	77
5.18	PrimaryGeneratorMessenger::RunActionMessenger Class Reference	77
5.18.1	Detailed Description	78
5.19	SteppingAction Class Reference	78
5.19.1	Detailed Description	79
5.19.2	Constructor & Destructor Documentation	79
5.19.2.1	SteppingAction	79
5.19.3	Member Function Documentation	79
5.19.3.1	UserSteppingAction	79
5.20	SteppingVerbose Class Reference	81
5.20.1	Detailed Description	82
5.20.2	Constructor & Destructor Documentation	82
5.20.2.1	~SteppingVerbose	82
6	File Documentation	83
6.1	AllAccumulatedStatistics.cc File Reference	83
6.1.1	Detailed Description	83

6.2	AllAccumulatedStatistics.hh File Reference	83
6.2.1	Detailed Description	84
6.3	DetectorConstruction.cc File Reference	84
6.3.1	Detailed Description	84
6.4	DetectorConstruction.hh File Reference	84
6.4.1	Detailed Description	85
6.5	DetectorMessenger.cc File Reference	85
6.5.1	Detailed Description	85
6.6	DetectorMessenger.hh File Reference	86
6.6.1	Detailed Description	86
6.7	EventAction.cc File Reference	86
6.7.1	Detailed Description	87
6.8	EventAction.hh File Reference	87
6.8.1	Detailed Description	88
6.8.2	Define Documentation	88
6.8.2.1	STDIO	88
6.9	EventActionMessenger.cc File Reference	88
6.9.1	Detailed Description	88
6.10	EventActionMessenger.hh File Reference	88
6.10.1	Detailed Description	89
6.11	EventHit.cc File Reference	89
6.11.1	Detailed Description	89
6.12	EventHit.hh File Reference	89
6.12.1	Detailed Description	90
6.13	G4EmStandardFysik.cc File Reference	90
6.13.1	Detailed Description	91
6.14	G4EmStandardFysik.hh File Reference	91
6.14.1	Detailed Description	92
6.15	MyPhysicsList.hh File Reference	92
6.15.1	Detailed Description	92
6.16	MyPhysicsList.icc File Reference	92
6.16.1	Detailed Description	93
6.17	PIBIDS.cc File Reference	93
6.17.1	Detailed Description	94

6.17.2	Function Documentation	94
6.17.2.1	main	94
6.18	PrimaryGeneratorAction.cc File Reference	96
6.18.1	Detailed Description	97
6.19	PrimaryGeneratorAction.hh File Reference	97
6.19.1	Detailed Description	97
6.20	PrimaryGeneratorMessenger.cc File Reference	98
6.20.1	Detailed Description	98
6.21	PrimaryGeneratorMessenger.hh File Reference	98
6.21.1	Detailed Description	99
6.22	RunAction.cc File Reference	99
6.22.1	Detailed Description	99
6.23	RunAction.hh File Reference	99
6.23.1	Detailed Description	100
6.24	RunActionMessenger.cc File Reference	100
6.24.1	Detailed Description	100
6.25	RunActionMessenger.hh File Reference	101
6.25.1	Detailed Description	101
6.26	SteppingAction.cc File Reference	101
6.26.1	Detailed Description	101
6.27	SteppingAction.hh File Reference	102
6.27.1	Detailed Description	102
6.27.2	Define Documentation	102
6.27.2.1	STDIO_H	102
6.28	SteppingVerbose.cc File Reference	103
6.28.1	Detailed Description	103
6.29	SteppingVerbose.hh File Reference	103
6.29.1	Detailed Description	103

Chapter 1

Main Page

1.1 Introduction

The Precalibrated Ion Beam Identification Detector Simulation software has the purpose of simulating a proposed detector design. It is written in Geant4. We would like to thank the authors of Geant4 and particularly the authors of the ExampleN03, which have been of great help in producing the software provided here.

1.2 Installation

In order to use the PIBIDS software, you need to have a correctly installed and configured copy of [Geant4](#) on your computer. When you have that you can simply download and compile the software from [here](#).

1.3 Compiling

Compiling should be no problem, just use the makefile provided. This makefile however contains some targets used for deploying it on some of Chalmers's remote servers. Just ignore these targets, and use the standard G4make and G4run commands.

1.4 Technical Details

The physics list QGSP_BERT was used. QGSP_BERT takes decay physics into account and uses the Bertini cascade model to compute spread of primary protons, neutrons, pions and kaons with energies below 10 GeV, which according to Geant4's reference webpage should give simulations in good accordance with measured data at these energies.

1.5 Miscellaneous

Todo

Complete this documentation.

Chapter 2

Todo List

page [Main Page](#) Complete this documentation.

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

AllAccumulatedStatistics (Small class to keep track of the accumulated statistics)	9
PrimaryGeneratorMessenger::AllAccumulatedStatistics (Small class to keep track of the accumulated statistics)	13
DetectorConstruction (This class constructs the detector, according to specifications)	16
PrimaryGeneratorMessenger::DetectorConstruction (This class constructs the detector, according to specifications)	34
DetectorMessenger (Creates commands that the user can invoke via the command line)	45
PrimaryGeneratorMessenger::DetectorMessenger (Creates commands that the user can invoke via the command line)	53
EventAction (Holds information about what to do at the beginning and end of each event)	55
EventActionMessenger (Creates commands that can be used to control the EventActions via the command line)	58
EventHit (Holds accumulated statistics for a single event hit)	60
PrimaryGeneratorMessenger::EventHit (Holds accumulated statistics for a single event hit)	61
G4EmStandardFysik (This class is extending the G4EmStandardPhysics class in order to apply a few changes to this physics list)	63
MyPhysicsList< T > (Customized physics list class)	64
PrimaryGeneratorMessenger::PrimaryGeneratorAction (Mandatory user class providing the primary particle generator)	65
PrimaryGeneratorAction (Mandatory user class providing the primary particle generator)	67
PrimaryGeneratorMessenger (Creates commands that allows the user to control the PrimaryGenerator via the command line)	69
RunAction (Here we can put stuff to be executed before and after each run) .	72

RunActionMessenger (Creates commands that can be used to control the RunAction via the command line)	75
PrimaryGeneratorMessenger::RunActionMessenger (Creates commands that can be used to control the RunAction via the command line)	77
SteppingAction (Contains information about what to do at each step in the simulation)	78
SteppingVerbose (Inclusion guard)	81

Chapter 4

File Index

4.1 File List

Here is a list of all documented files with brief descriptions:

AllAccumulatedStatistics.cc (Source file for the AllAccumulatedStatistics class)	83
AllAccumulatedStatistics.hh (Header file for the AllAccumulatedStatistics class)	83
DetectorConstruction.cc (Source file for the DetectorConstruction class) . . .	84
DetectorConstruction.hh (Header file for the DetectorConstruction class) . . .	84
DetectorMessenger.cc (Source file for the DetectorMessenger class)	85
DetectorMessenger.hh (Header file for the DetectorMessenger class)	86
EventAction.cc (Source file for the EventAction class)	86
EventAction.hh (Header file for the EventAction class)	87
EventActionMessenger.cc (Source file for the EventActionMessenger class) .	88
EventActionMessenger.hh (Header file for the EventActionMessenger class) .	88
EventHit.cc (Source file for the EventHit class)	89
EventHit.hh (Header file for the EventHit class)	89
G4EmStandardFysik.cc (Source file for the G4EmStandardFysik class)	90
G4EmStandardFysik.hh (Header file for the G4EmStandardFysik class) . . .	91
MyPhysicsList.hh (Header file for the MyPhysicsList class)	92
MyPhysicsList.icc (Source file for the MyPhysicsList class)	92
PIBIDS.cc (Main file, starting the simulation)	93
PrimaryGeneratorAction.cc (Source file for the PrimaryGeneratorAction class)	96
PrimaryGeneratorAction.hh (Header file for the PrimaryGeneratorAction class)	97
PrimaryGeneratorMessenger.cc (Source file for the PrimaryGeneratorMessen- ger class)	98
PrimaryGeneratorMessenger.hh (Header file for the PrimaryGeneratorMes- senger class)	98
RunAction.cc (Source file for the RunAction class)	99
RunAction.hh (<Inclusion guard)	99
RunActionMessenger.cc (Source file for the RunActionMessenger class) . . .	100
RunActionMessenger.hh (Header file for the RunActionMessenger class) . . .	101
SteppingAction.cc (Source file for the SteppingAction class)	101
SteppingAction.hh (Header file for the SteppingAction class)	102

SteppingVerbose.cc (Source file for the SteppingVerbose class)	103
SteppingVerbose.hh (Header file for the SteppingVerbose class)	103

Chapter 5

Class Documentation

5.1 AllAccumulatedStatistics Class Reference

Small class to keep track of the accumulated statistics.

```
#include <AllAccumulatedStatistics.hh>
```

Public Types

- enum **ScintillatorPlate** { **UpperFront** = 0, **LowerFront** = 1, **Back** = 2 }

Public Member Functions

- [AllAccumulatedStatistics](#) (G4double numberOfScintillatorTubes, G4double summationTime)
Constructor, constructs the class.
- [~AllAccumulatedStatistics](#) ()
Destructor.
- void [AddScintillatorTubeEvent](#) (G4int scintillatorID, G4double time, G4double energy)
Add a scintillator event when such occurs.
- void [AddGeDetectorEvent](#) (G4double time, G4double energy)
Add a germanium detector event when such occurs.
- void [AddScintillatorPlateEvent](#) (ScintillatorPlate plate, G4double time, G4double energy)
Add a scintillator plate event.
- std::vector< [EventHit](#) > [GetEventHits](#) ()
Returns the eventHits we have accumulated.
- bool [timeCheck](#) (G4double time)
Check if the time falls within the frame of the current [EventHit](#) object, if not it creates a new [EventHit](#) with the current time as the start time.

Private Attributes

- `std::vector< EventHit > eventHits`
The event hits for this accumulated statistics object.
- `G4double scintillatorTubes`
- `G4double sumTime`

5.1.1 Detailed Description

Small class to keep track of the accumulated statistics.

Definition at line 29 of file AllAccumulatedStatistics.hh.

5.1.2 Constructor & Destructor Documentation

5.1.2.1 AllAccumulatedStatistics::AllAccumulatedStatistics (G4double numberOfScintillatorTubes, G4double summationTime)

Constructor, constructs the class.

Parameters

<i>numberOf-Scintillator-Tubes</i>	Number of scintillator tubes.
<i>summation-Time</i>	Summation time.

Definition at line 3 of file AllAccumulatedStatistics.cc.

```

:scintillatorTubes(numberOfScintillatorTubes), sumTime(summationTime)
{
}

```

5.1.3 Member Function Documentation

5.1.3.1 void AllAccumulatedStatistics::AddGeDetectorEvent (G4double time, G4double energy)

Add a germanium detector event when such occurs.

Parameters

<i>time</i>	The time of the event.
<i>energy</i>	The energy of deposited.

Definition at line 33 of file AllAccumulatedStatistics.cc.

References eventHits, and timeCheck().

Referenced by EventAction::AddGeDetectorEvent().

```
{
    if(energy>0.1*eV) //don't add non-existent energies.
    {
        timeCheck(time);
        eventHits.back().geDetectorEnergy+=energy;
    }
}
```

5.1.3.2 void AllAccumulatedStatistics::AddScintillatorPlateEvent (ScintillatorPlate *plate*, G4double *time*, G4double *energy*)

Add a scintillator plate event.

Parameters

<i>plate</i>	The plate the event occurred in
<i>time</i>	The time of the event.
<i>energy</i>	The energy of deposited.

Definition at line 42 of file AllAccumulatedStatistics.cc.

References eventHits, and timeCheck().

Referenced by EventAction::AddScintillatorPlateEvent().

```
{
    if(energy>0.1*eV) //don't add non-existent energies.
    {
        timeCheck(time);
        if(plate==UpperFront)
            eventHits.back().upperFrontScintillatorPanelEnergy+=energy;
        if(plate==LowerFront)
            eventHits.back().lowerFrontScintillatorPanelEnergy+=energy;
        if(plate==Back)
            eventHits.back().backScintillatorPanelEnergy+=energy;
    }
}
```

5.1.3.3 void AllAccumulatedStatistics::AddScintillatorTubeEvent (G4int *scintillatorID*, G4double *time*, G4double *energy*)

Add a scintillator event when such occurs.

Parameters

<i>scintillatorID</i>	The scintillator ID
<i>time</i>	The time of the event.
<i>energy</i>	Deposited energy.

Definition at line 24 of file AllAccumulatedStatistics.cc.

References eventHits, and timeCheck().

Referenced by EventAction::AddScintillatorTubeEvent().

```
{
    if(energy>0.1*eV) //don't add non-existent energies.
    {
        timeCheck(time);
        eventHits.back().scintillatorTubeEnergies.at(ScintillatorID)+=energy;
    }
}
```

5.1.3.4 std::vector< EventHit > AllAccumulatedStatistics::GetEventHits ()

Returns the eventHits we have accumulated.

Returns

the EventHits accumulated this far.

Definition at line 56 of file AllAccumulatedStatistics.cc.

References eventHits.

Referenced by RunAction::WriteResultsToFile().

```
{
    return eventHits;
}
```

5.1.3.5 bool AllAccumulatedStatistics::timeCheck (G4double time)

Check if the time falls within the frame of the current [EventHit](#) object, if not it creates a new [EventHit](#) with the current time as the start time.

Returns

true if a new [EventHit](#) object was created as a result of the check, false if not.

Parameters

<i>time</i>	Time to check for.
-------------	--------------------

Definition at line 14 of file AllAccumulatedStatistics.cc.

References eventHits.

Referenced by AddGeDetectorEvent(), AddScintillatorPlateEvent(), and AddScintillatorTubeEvent().

```
{
```

5.2 PrimaryGeneratorMessenger::AllAccumulatedStatistics Class Reference 13

```
if(eventHits.empty() || time > (eventHits.back().time + sumTime ))
{
    eventHits.push_back(EventHit(scintillatorTubes,time));
    return true;
}
return false;
}
```

The documentation for this class was generated from the following files:

- [AllAccumulatedStatistics.hh](#)
- [AllAccumulatedStatistics.cc](#)

5.2 PrimaryGeneratorMessenger::AllAccumulatedStatistics Class Reference

Small class to keep track of the accumulated statistics.

```
#include <PrimaryGeneratorMessenger.hh>
```

Public Types

- enum **ScintillatorPlate** { **UpperFront** = 0, **LowerFront** = 1, **Back** = 2 }

Public Member Functions

- [AllAccumulatedStatistics](#) (G4double numberOfScintillatorTubes, G4double summationTime)
Constructor, constructs the class.
- [~AllAccumulatedStatistics](#) ()
Destructor.
- void [AddScintillatorTubeEvent](#) (G4int scintillatorID, G4double time, G4double energy)
Add a scintillator event when such occurs.
- void [AddGeDetectorEvent](#) (G4double time, G4double energy)
Add a germanium detector event when such occurs.
- void [AddScintillatorPlateEvent](#) (ScintillatorPlate plate, G4double time, G4double energy)
Add a scintillator plate event.
- std::vector< [EventHit](#) > [GetEventHits](#) ()
Returns the eventHits we have accumulated.
- bool [timeCheck](#) (G4double time)
Check if the time falls within the frame of the current [EventHit](#) object, if not it creates a new [EventHit](#) with the current time as the start time.

Private Attributes

- `std::vector< EventHit > eventHits`
The event hits for this accumulated statistics object.
- `G4double scintillatorTubes`
- `G4double sumTime`

5.2.1 Detailed Description

Small class to keep track of the accumulated statistics.

Definition at line 29 of file PrimaryGeneratorMessenger.hh.

5.2.2 Constructor & Destructor Documentation

5.2.2.1 `PrimaryGeneratorMessenger::AllAccumulatedStatistics::AllAccumulatedStatistics (G4double numberOfScintillatorTubes, G4double summationTime)`

Constructor, constructs the class.

Parameters

<i>numberOf-Scintillator-Tubes</i>	Number of scintillator tubes.
<i>summation-Time</i>	Summation time.

5.2.3 Member Function Documentation

5.2.3.1 `void PrimaryGeneratorMessenger::AllAccumulatedStatistics::AddGeDetectorEvent (G4double time, G4double energy)`

Add a germanium detector event when such occurs.

Parameters

<i>time</i>	The time of the event.
<i>energy</i>	The energy of deposited.

5.2.3.2 `void PrimaryGeneratorMessenger::AllAccumulatedStatistics::AddScintillatorPlateEvent (ScintillatorPlate plate, G4double time, G4double energy)`

Add a scintillator plate event.

5.2 PrimaryGeneratorMessenger::AllAccumulatedStatistics Class Reference 15

Parameters

<i>plate</i>	The plate the event occurred in
<i>time</i>	The time of the event.
<i>energy</i>	The energy of deposited.

5.2.3.3 void PrimaryGeneratorMessenger::AllAccumulatedStatistics::AddScintillatorTubeEvent (G4int *scintillatorID*, G4double *time*, G4double *energy*)

Add a scintillator event when such occurs.

Parameters

<i>scintillatorID</i>	The scintillator ID
<i>time</i>	The time of the event.
<i>energy</i>	Deposited energy.

5.2.3.4 std::vector<EventHit> PrimaryGeneratorMessenger::AllAccumulatedStatistics::GetEventHits ()

Returns the eventHits we have accumulated.

Returns

the EventHits accumulated this far.

5.2.3.5 bool PrimaryGeneratorMessenger::AllAccumulatedStatistics::timeCheck (G4double *time*)

Check if the time falls within the frame of the current [EventHit](#) object, if not it creates a new [EventHit](#) with the current time as the start time.

Returns

true if a new [EventHit](#) object was created as a result of the check, false if not.

Parameters

<i>time</i>	Time to check for.
-------------	--------------------

The documentation for this class was generated from the following file:

- [AllAccumulatedStatistics.hh](#)

5.3 DetectorConstruction Class Reference

This class constructs the detector, according to specifications.

```
#include <DetectorConstruction.hh>
```

Public Member Functions

- [DetectorConstruction \(\)](#)
Constructor, constructs the object.
- [~DetectorConstruction \(\)](#)
Destructor, destroys the object.
- void **SetDefaultDetectorParameters ()**
- void [SetScintillatorMaterial](#) (G4String)
Sets the scintillator material.
- void [SetScintillatorDiameter](#) (G4double)
Sets the scintillator diameter.
- void [SetShieldingMaterial](#) (G4String)
Sets the shielding material.
- void [SetGeDetectorShellMaterial](#) (G4String)
Sets the shell material of the Germanium detector.
- void [SetBeamBlockerMaterial](#) (G4String)
Sets the beam blocker material.
- void [SetShieldingBeamBlockerMaterial](#) (G4String)
Sets the shielding beam blocker material.
- void [SetGermaniumDetectorMaterial](#) (G4String)
Sets the Germanium detector material.
- void [SetEffectiveSlowdownThickness](#) (G4double thickness)
Sets the effective slowdown thickness.
- void [SetSlowdownLength](#) (G4double length)
Sets the slowdown length.
- void [SetSlowdownThickness](#) (G4double thickness)
Sets the slowdown thickness.
- void [SetShieldingThickness](#) (G4double thickness)
Sets the shielding thickness.
- void [SetTopShieldingThickness](#) (G4double thickness)
Sets the top shielding thickness.
- void [SetBeamBlockerThickness](#) (G4double thickness)
Sets beamblocker thickness.
- void [SetShieldingBeamBlockerThickness](#) (G4double thickness)
Sets shielding beam blocker thickness.
- void [SetGeDetectorRadius](#) (G4double radius)
Set Ge detector radius.
- void [SetGeDetectorShellLength](#) (G4double length)

- Set Ge detector shell length.*

 - void [SetGeDetectorLength](#) (G4double length)
- Set Ge detector length.*

 - void [SetGeDetectorShellInnerRadius](#) (G4double radius)
- Set Ge detector shell inner radius.*

 - void [SetGeDetectorShellOuterRadius](#) (G4double radius)
- Set Ge detector shell outer radius.*

 - void [SetGapMaterial](#) (G4String)
- Sets the gap material.*

 - void [SetScintillatorShieldingGapThickness](#) (G4double)
- Sets the gap thickness.*

 - void [SetSlowdownMaterial](#) (G4String)
- Sets the slowdown material.*

 - void [SetSlowdownHeightGap](#) (G4double)
- Sets the slowdown height gap.*

 - void [SetNumberOfScintillatorBoxes](#) (G4int)
- Sets the number of scintillator boxes.*

 - void [SetNumberOfScintillatorsPerBox](#) (G4int)
- Sets the number of layers of scintillators.*

 - void [SetBetaFlipping](#) (G4bool)
- Sets beta detector flipping.*

 - G4VPhysicalVolume * [Construct](#) ()
- Constructs the detector.*

 - void [UpdateGeometry](#) ()
- Updates the geometry of the detector.*

 - G4double [GetWorldSizeX](#) ()
- Returns the X size of the world.*

 - G4double [GetWorldSizeY](#) ()
- Returns the Y size of the world.*

 - G4double [GetWorldSizeZ](#) ()
- Returns the Z size of the world.*

 - G4double [GetScintillatorBoxSizeX](#) ()
- <Returns the X size of the scintillator box.*

 - G4double [GetScintillatorBoxSizeY](#) ()
- Returns the Y size of the scintillator box.*

 - G4double [GetScintillatorBoxSizeZ](#) ()
- Returns the Z size of the scintillator box.*

 - G4double [GetBackScintillatorPlateSizeY](#) ()
- Returns Y size of back scintillator plate.*

 - G4int [GetNumberOfScintillatorBoxes](#) ()
- Returns the number of layers.*

 - G4int [GetNumberOfScintillatorsPerBox](#) ()
- Returns the number of layers.*

 - G4Material * [GetScintillatorMaterial](#) ()

Returns the scintillator material.

- G4double [GetScintillatorDiameter](#) ()

Returns the scintillator diameter.

- G4Material * [GetGapMaterial](#) ()

Returns the gap material.

- G4double [GetScintillatorShieldingGapThickness](#) ()

Returns the gap thickness.

- const G4VPhysicalVolume * [GetphysiWorld](#) ()

Returns the physical world.

- const G4VPhysicalVolume * [GetAbsorber](#) ()

Returns the absorber.

- const G4VPhysicalVolume * [GetGap](#) ()

Returns the gap.

- const G4VPhysicalVolume * [GetScintillator](#) ()

Returns the scintillator.

- const G4VPhysicalVolume * [GetUpperBetaDetector](#) ()

Returns the upper beta detector.

- const G4VPhysicalVolume * [GetLowerBetaDetector](#) ()

Returns the lower Beta detector.

- const G4VPhysicalVolume * [GetGermaniumDetector](#) ()

Returns the germanium detector.

- const G4VPhysicalVolume * [GetUpperFrontScintillatorPlate](#) ()

Returns the upper front scintillator plate.

- const G4VPhysicalVolume * [GetLowerFrontScintillatorPlate](#) ()

Returns the lower front scintillator plate.

- const G4VPhysicalVolume * [GetBackScintillatorPlate](#) ()

Returns the back scintillator plate.

Private Member Functions

- void **ConstructWorld** ()
- void **ConstructBetaDetector** ()
- void **ConstructScintillatorBox** ()
- void **ConstructScintillatorSlot** ()
- void **ConstructScintillator** ()
- void **ConstructShielding** ()
- void **ConstructBeamBlocker** ()
- void **ConstructScintillatorPlates** ()
- void **ConstructSlowdown** ()
- void **ConstructGermaniumDetector** ()
- void **ConstructShieldingBeamBlocker** ()
- void **SetVisualizationAttributes** ()
- void [DefineMaterials](#) ()

Defines the materials.

- void [ComputeDependentDetectorParameters](#) ()
Computes the scintillator parameters.
- G4VPhysicalVolume * [ConstructDetector](#) ()
Constructs the detector.

Private Attributes

- G4Material * [ScintillatorMaterial](#)
Scintillator material.
- G4double [ScintillatorThickness](#)
Scintillator thickness.
- G4Material * [GapMaterial](#)
Gap material.
- G4double [ScintillatorShieldingGapThickness](#)
Gap thickness.
- G4int [NumberOfScintillatorsPerBox](#)
Number of layers of scintillators (typically 10).
- G4int [NumberOfScintillatorBoxes](#)
Number of scintillator boxes (typically 2).
- G4double [ScintillatorBoxSizeX](#)
Extent in X-direction of the scintillator box.
- G4double [ScintillatorBoxSizeY](#)
Extent in Y-direction of the scintillator box.
- G4double [ScintillatorBoxSizeZ](#)
Extent in Z-direction of the scintillator box.
- G4double [SlotWidthY](#)
Width of slot in Y-direction.
- G4double [SlotWidthZ](#)
Width of slot.
- G4double [ScintillatorDiameter](#)
Diameter of scintillator.
- G4double [ScintillatorLength](#)
Length of scintillator.
- G4double [ScintillatorPlateThickness](#)
Thickness of scintillator plate.
- G4double [FrontScintillatorPlateSizeY](#)
Extent in Y-direction of front scintillator plate.
- G4double [FrontScintillatorPlateSizeZ](#)
Extent in Z-direction of front scintillator plate.
- G4double [BackScintillatorPlateSizeY](#)
Extent in X-direction of back scintillator plate.
- G4double [GeDetectorRadius](#)
Radius of Ge detector.

- G4double [GeDetectorLength](#)
Length of Ge detector.
- G4double [GeDetectorShellLength](#)
Length of Ge detector shell.
- G4double [GeDetectorShellInnerRadius](#)
Inner radius of Ge detector shell.
- G4double [GeDetectorShellOuterRadius](#)
Outer radius of Ge detector shell.
- G4double [BeamBlockerThickness](#)
Thickness of beam blocker.
- G4double [ShieldingBeamBlockerThickness](#)
Shielding beam blocker thickness.
- G4double [GeDetectorContainerSizeX](#)
Size of the detector container in the Z-direction.
- G4Material * [defaultMaterial](#)
Default material.
- G4double [WorldSizeZ](#)
Size of world in the Z-direction.
- G4double [WorldSizeY](#)
Size of world in the Y-direction.
- G4double [WorldSizeX](#)
Size of world in the X-direction.
- G4Material * [ShieldingMaterial](#)
Shielding material.
- G4double [ShieldingThickness](#)
Thickness of shielding.
- G4double [TopShieldingThickness](#)
Thickness of top shielding.
- G4Material * [BeamBlockerMaterial](#)
The beam blocker material.
- G4Material * [ShieldingBeamBlockerMaterial](#)
Shielding beam blocker material.
- G4Box * [solidWorld](#)
Pointer to the solid World.
- G4LogicalVolume * [logicWorld](#)
Pointer to the logical World.
- G4VPhysicalVolume * [physiWorld](#)
Pointer to the physical World.
- G4Material * [GeDetectorShellMaterial](#)
The Ge detector shell material.
- G4Material * [GermaniumDetectorMaterial](#)
The Ge detector material.
- G4Box * [solidScintillatorBox](#)

- Pointer to the solid Calor.*
 - G4LogicalVolume * [logicScintillatorBox](#)
- Pointer to the logical Calor.*
 - G4VPhysicalVolume * [physiScintillatorBox](#)
- Pointer to the physical Calor.*
 - G4Box * [solidSlot](#)
- Pointer to the solid Layer.*
 - G4LogicalVolume * [logicSlot](#)
- Pointer to the logical Layer.*
 - G4VPhysicalVolume * [physiSlot](#)
- Pointer to the physical Layer.*
 - G4Tubs * [solidScintillator](#)
- Pointer to the solid Scintillator.*
 - G4LogicalVolume * [logicScintillator](#)
- Pointer to the logical Scintillator.*
 - G4VPhysicalVolume * [physiScintillator](#)
- Pointer to the physical Scintillator.*
 - G4Box * [solidGap](#)
- Pointer to the solid Gap.*
 - G4LogicalVolume * [logicGap](#)
- Pointer to the logical Gap.*
 - G4VPhysicalVolume * [physiGap](#)
- Pointer to the physical Gap.*
 - G4Box * [solidShielding](#)
- Pointer to the solid shielding.*
 - G4LogicalVolume * [logicShielding](#)
- Pointer to the logical shielding.*
 - G4VPhysicalVolume * [physiShielding](#)
- Pointer to the physical shielding.*
 - G4Box * [solidTopShielding](#)
- Pointer to the solid shielding.*
 - G4LogicalVolume * [logicTopShielding](#)
- Pointer to the logical shielding.*
 - G4VPhysicalVolume * [physiTopShielding](#)
- Pointer to the physical shielding.*
 - G4Box * [solidShieldingBeamBlocker](#)
- Pointer to the solid shielding.*
 - G4LogicalVolume * [logicShieldingBeamBlocker](#)
- Pointer to the logical shielding.*
 - G4VPhysicalVolume * [physiShieldingBeamBlocker](#)
- Pointer to the physical shielding.*
 - G4Box * [solidTopShieldingBeamBlocker](#)
- Pointer to the solid shielding.*

- G4LogicalVolume * [logicTopShieldingBeamBlocker](#)
Pointer to the logical shielding.
- G4VPhysicalVolume * [physiTopShieldingBeamBlocker](#)
Pointer to the physical shielding.
- G4Box * [solidBetaDetector](#)
Pointer to the solid UpperBetaDetector.
- G4LogicalVolume * [logicBetaDetector](#)
Pointer to the logical UpperBetaDetector.
- G4VPhysicalVolume * [physiUpperBetaDetector](#)
Pointer to the physical UpperBetaDetector.
- G4Box * [solidLowerBetaDetector](#)
Pointer to the solid LowerBetaDetector.
- G4LogicalVolume * [logicLowerBetaDetector](#)
Pointer to the logical LowerBetaDetector.
- G4VPhysicalVolume * [physiLowerBetaDetector](#)
Pointer to the physical LowerBetaDetector.
- [DetectorMessenger](#) * [detectorMessenger](#)
Pointer to the Messenger.
- G4Box * [solidFrontScintillatorPlate](#)
Pointer to the solid FrontScintillatorPlate.
- G4LogicalVolume * [logicFrontScintillatorPlate](#)
Pointer to the logical FrontScintillatorPlate.
- G4VPhysicalVolume * [physiUpperFrontScintillatorPlate](#)
Pointer to the physical UpperFrontScintillatorPlate.
- G4VPhysicalVolume * [physiLowerFrontScintillatorPlate](#)
Pointer to the physical LowerFrontScintillatorPlate.
- G4Box * [solidBackScintillatorPlate](#)
Pointer to the solid BackScintillatorPlate.
- G4LogicalVolume * [logicBackScintillatorPlate](#)
Pointer to the logical BackScintillatorPlate.
- G4VPhysicalVolume * [physiBackScintillatorPlate](#)
Pointer to the physical BackScintillatorPlate.
- G4double [SlowdownThickness](#)
Thickness of slowdown.
- G4double [SlowdownLength](#)
Length of slowdown.
- G4double [SlowdownHeightGap](#)
Peight of slowdown gap.
- G4double [EffectiveSlowdownThickness](#)
The thickness that the particles from the gun meet.
- G4Trap * [solidSlowdown](#)
Pointer to the solid FrontScintillatorPlate.
- G4LogicalVolume * [logicSlowdown](#)

- Pointer to the logical Slowdown.*
- G4VPhysicalVolume * [physiUpperSlowdown](#)
Pointer to the physical UpperSlowdown.
- G4VPhysicalVolume * [physiLowerSlowdown](#)
Pointer to the physical LowerSlowdown.
- G4Material * [SlowdownMaterial](#)
The slowdown material.
- G4Tubs * [solidGermaniumDetector](#)
Pointer to the solid germanium detector.
- G4LogicalVolume * [logicGermaniumDetector](#)
Pointer to the logical germanium detector.
- G4VPhysicalVolume * [physiGermaniumDetector](#)
Pointer to the physical germanium detector.
- G4Tubs * [solidGeDetectorShell](#)
Pointer to the solid germanium detector shell.
- G4LogicalVolume * [logicGeDetectorShell](#)
Pointer to the logical germanium detector shell.
- G4VPhysicalVolume * [physiGeDetectorShell](#)
Pointer to the physical germanium detector shell.
- G4Box * [solidGeDetectorContainer](#)
Pointer to the solid germanium detector container.
- G4LogicalVolume * [logicGeDetectorContainer](#)
Pointer to the logical germanium detector container.
- G4VPhysicalVolume * [physiGeDetectorContainer](#)
Pointer to the physical germanium detector container.
- G4Tubs * [solidGeDetectorShellTop](#)
Pointer to the solid germanium detector shell.
- G4LogicalVolume * [logicGeDetectorShellTop](#)
Pointer to the logical germanium detector shell.
- G4VPhysicalVolume * [physiGeDetectorShellTop](#)
Pointer to the physical germanium detector shell.
- G4Tubs * [solidGeDetectorShellBottom](#)
Pointer to the solid germanium detector shell.
- G4LogicalVolume * [logicGeDetectorShellBottom](#)
Pointer to the logical germanium detector shell.
- G4VPhysicalVolume * [physiGeDetectorShellBottom](#)
Pointer to the physical germanium detector shell.
- G4Box * [solidBeamBlocker](#)
Pointer to the solid BeamBlocker.
- G4LogicalVolume * [logicBeamBlocker](#)
Pointer to the logical BeamBlocker.
- G4VPhysicalVolume * [physiBeamBlocker](#)
Pointer to the physical BeamBlocker.

- G4double [BetaDetectorPositionY](#)
Y position of the Beta detectors.
- G4double [BeamBlockerYWidth](#)
Y width of beam blocker.
- G4double [BeamBlockerZWidth](#)
Z width of beam blocker.
- G4double [BeamBlockerXPosition](#)
X position of beam blocker.
- G4double **ScintillatorPositionX**
- G4double **ScintillatorPositionY**
- G4double **ScintillatorPositionZ**
- G4double **ShieldingPositionX**
- G4double **ShieldingPositionY**
- G4double **ShieldingPositionZ**
- G4double **TopShieldingPositionX**
- G4double **TopShieldingPositionY**
- G4double **TopShieldingPositionZ**
- G4double **PlateShieldingSpacing**
- G4double **ShieldingBeamBlockerPositionX**
- G4bool **FlipBeta**

5.3.1 Detailed Description

This class constructs the detector, according to specifications.

Definition at line 54 of file DetectorConstruction.hh.

5.3.2 Member Function Documentation

5.3.2.1 G4double DetectorConstruction::GetBackScintillatorPlateSizeY () [inline]

Returns Y size of back scintillator plate.

Returns the number of scintillator boxes.

Definition at line 165 of file DetectorConstruction.hh.

References [BackScintillatorPlateSizeY](#).

Referenced by [PrimaryGeneratorAction::GeneratePrimaries\(\)](#), and [PrimaryGeneratorAction::PrintTargetSize\(\)](#).

5.3.2.2 void DetectorConstruction::SetBeamBlockerMaterial (G4String materialChoice)

Sets the beam blocker material.

Parameters

<i>material-Choice</i>	The name of the beam blocker material.
------------------------	--

Definition at line 839 of file DetectorConstruction.cc.

References BeamBlockerMaterial.

Referenced by DetectorMessenger::SetNewValue().

```
{
    // search the material by its name
    G4Material* pttoMaterial = G4Material::GetMaterial(materialChoice);
    if (pttoMaterial) BeamBlockerMaterial = pttoMaterial;
}
```

5.3.2.3 void DetectorConstruction::SetBeamBlockerThickness (G4double *thickness*)

Sets beamblocker thickness.

Parameters

<i>thickness</i>	BeamBlocker thickness.
------------------	------------------------

Definition at line 872 of file DetectorConstruction.cc.

References BeamBlockerThickness.

Referenced by DetectorMessenger::SetNewValue().

```
{
    BeamBlockerThickness = thickness;
}
```

5.3.2.4 void DetectorConstruction::SetBetaFlipping (G4bool *flip*)

Sets beta detector flipping.

Parameters

<i>flip</i>	Flip (true/false)
-------------	-------------------

Definition at line 931 of file DetectorConstruction.cc.

Referenced by DetectorMessenger::SetNewValue().

```
{
    FlipBeta = flip;
}
```

5.3.2.5 void DetectorConstruction::SetEffectiveSlowdownThickness (G4double *thickness*)

Sets the effective slowdown thickness.

Parameters

<i>thickness</i>	The thickness of effective slowdown.
------------------	--------------------------------------

Definition at line 802 of file DetectorConstruction.cc.

References EffectiveSlowdownThickness.

Referenced by DetectorMessenger::SetNewValue().

```
{
    EffectiveSlowdownThickness = thickness;
}
```

5.3.2.6 void DetectorConstruction::SetGapMaterial (G4String *materialChoice*)

Sets the gap material.

Parameters

<i>material-Choice</i>	The name of the gap material.
------------------------	-------------------------------

Definition at line 818 of file DetectorConstruction.cc.

References GapMaterial.

Referenced by DetectorMessenger::SetNewValue().

```
{
    // search the material by its name
    G4Material* pttoMaterial = G4Material::GetMaterial(materialChoice);
    if (pttoMaterial) GapMaterial = pttoMaterial;
}
```

5.3.2.7 void DetectorConstruction::SetGeDetectorLength (G4double *length*)

Set Ge detector length.

Parameters

<i>length</i>	Length
---------------	--------

Definition at line 904 of file DetectorConstruction.cc.

References GeDetectorLength.

Referenced by DetectorMessenger::SetNewValue().

```
{
    GeDetectorLength = length;
}
```

```
}
```

5.3.2.8 void DetectorConstruction::SetGeDetectorRadius (G4double *radius*)

Set Ge detector radius.

Parameters

<i>radius</i>	Detector radius.
---------------	------------------

Definition at line 894 of file DetectorConstruction.cc.

References GeDetectorRadius.

Referenced by DetectorMessenger::SetNewValue().

```
{  
    GeDetectorRadius = radius;  
}
```

5.3.2.9 void DetectorConstruction::SetGeDetectorShellInnerRadius (G4double *radius*)

Set Ge detector shell inner radius.

Parameters

<i>radius</i>	Detector radius
---------------	-----------------

Definition at line 909 of file DetectorConstruction.cc.

References GeDetectorShellInnerRadius.

Referenced by DetectorMessenger::SetNewValue().

```
{  
    GeDetectorShellInnerRadius = radius;  
}
```

5.3.2.10 void DetectorConstruction::SetGeDetectorShellLength (G4double *length*)

Set Ge detector shell length.

Parameters

<i>length</i>	Length
---------------	--------

Definition at line 899 of file DetectorConstruction.cc.

References GeDetectorShellLength.

Referenced by DetectorMessenger::SetNewValue().

```
{
    GeDetectorShellLength = length;
}
```

5.3.2.11 void DetectorConstruction::SetGeDetectorShellOuterRadius (G4double *radius*)

Set Ge detector shell outer radius.

Parameters

<i>radius</i>	Radius.
---------------	---------

Definition at line 914 of file DetectorConstruction.cc.

References GeDetectorShellOuterRadius.

Referenced by DetectorMessenger::SetNewValue().

```
{
    GeDetectorShellOuterRadius = radius;
}
```

5.3.2.12 void DetectorConstruction::SetGermaniumDetectorMaterial (G4String *materialChoice*)

Sets the Germanium detector material.

Parameters

<i>material-Choice</i>	The name of the Germanium detector material.
------------------------	--

Definition at line 853 of file DetectorConstruction.cc.

References GermaniumDetectorMaterial.

Referenced by DetectorMessenger::SetNewValue().

```
{
    // search the material by its name
    G4Material* pttoMaterial = G4Material::GetMaterial(materialChoice);
    if (pttoMaterial) GermaniumDetectorMaterial = pttoMaterial;
}
```

5.3.2.13 void DetectorConstruction::SetNumberOfScintillatorBoxes (G4int *val*)

Sets the number of scintillator boxes.

Parameters

<i>val</i>	Number of boxes.
------------	------------------

Definition at line 925 of file DetectorConstruction.cc.

References NumberOfScintillatorBoxes.

Referenced by DetectorMessenger::SetNewValue().

```
{  
    NumberOfScintillatorBoxes = val;  
}
```

5.3.2.14 void DetectorConstruction::SetNumberOfScintillatorsPerBox (G4int *val*)

Sets the number of layers of scintillators.

Parameters

<i>val</i>	The number of scintillator layers.
------------	------------------------------------

Definition at line 920 of file DetectorConstruction.cc.

References NumberOfScintillatorsPerBox.

Referenced by DetectorMessenger::SetNewValue().

```
{  
    NumberOfScintillatorsPerBox = val;  
}
```

5.3.2.15 void DetectorConstruction::SetScintillatorDiameter (G4double *val*)

Sets the scintillator diameter.

Parameters

<i>val</i>	The diameter of the scintillator.
------------	-----------------------------------

Definition at line 860 of file DetectorConstruction.cc.

References ScintillatorDiameter.

Referenced by DetectorMessenger::SetNewValue().

```
{  
    // change Absorber thickness and recompute the calorimeter parameters  
    ScintillatorDiameter = val;  
}
```

5.3.2.16 void DetectorConstruction::SetScintillatorMaterial (G4String *materialChoice*)

Sets the scintillator material.

Parameters

<i>material-Choice</i>	The name of the material to set.
------------------------	----------------------------------

Definition at line 788 of file DetectorConstruction.cc.

References ScintillatorMaterial.

Referenced by DetectorMessenger::SetNewValue().

```
{
  // search the material by its name
  G4Material* pttoMaterial = G4Material::GetMaterial(materialChoice);
  if (pttoMaterial) ScintillatorMaterial = pttoMaterial;
}
```

5.3.2.17 void DetectorConstruction::SetScintillatorShieldingGapThickness (G4double *val*)

Sets the gap thickness.

Parameters

<i>val</i>	The gap thickness.
------------	--------------------

Definition at line 866 of file DetectorConstruction.cc.

References ScintillatorShieldingGapThickness.

Referenced by DetectorMessenger::SetNewValue().

```
{
  // change Gap thickness and recompute the calorimeter parameters
  ScintillatorShieldingGapThickness = val;
}
```

5.3.2.18 void DetectorConstruction::SetShieldingBeamBlockerMaterial (G4String *materialChoice*)

Sets the shielding beam blocker material.

Parameters

<i>material-Choice</i>	The material of the shielding beam blocker.
------------------------	---

Definition at line 847 of file DetectorConstruction.cc.

References ShieldingBeamBlockerMaterial.

Referenced by DetectorMessenger::SetNewValue().

```
{  
    G4Material* pttoMaterial = G4Material::GetMaterial(materialChoice);  
    if (pttoMaterial) ShieldingBeamBlockerMaterial = pttoMaterial;  
}
```

5.3.2.19 void DetectorConstruction::SetShieldingBeamBlockerThickness (G4double *thickness*)

Sets shielding beam blocker thickness.

Parameters

<i>thickness</i>	Shielding beam blocker thickness
------------------	----------------------------------

Definition at line 877 of file DetectorConstruction.cc.

References ShieldingBeamBlockerThickness.

Referenced by DetectorMessenger::SetNewValue().

```
{  
    ShieldingBeamBlockerThickness = thickness;  
}
```

5.3.2.20 void DetectorConstruction::SetShieldingMaterial (G4String *materialChoice*)

Sets the shielding material.

Parameters

<i>material-Choice</i>	The name of the shielding material.
------------------------	-------------------------------------

Definition at line 832 of file DetectorConstruction.cc.

References ShieldingMaterial.

Referenced by DetectorMessenger::SetNewValue().

```
{  
    // search the material by its name  
    G4Material* pttoMaterial = G4Material::GetMaterial(materialChoice);  
    if (pttoMaterial) ShieldingMaterial = pttoMaterial;  
}
```

5.3.2.21 void DetectorConstruction::SetShieldingThickness (G4double *thickness*)

Sets the shielding thickness.

Parameters

<i>thickness</i>	The shielding thickness
------------------	-------------------------

Definition at line 882 of file DetectorConstruction.cc.

References ShieldingThickness.

Referenced by DetectorMessenger::SetNewValue().

```
{  
    ShieldingThickness = thickness;  
}
```

5.3.2.22 void DetectorConstruction::SetSlowdownLength (G4double *length*)

Sets the slowdown length.

Parameters

<i>length</i>	The slowdown length
---------------	---------------------

Definition at line 808 of file DetectorConstruction.cc.

References SlowdownLength.

Referenced by DetectorMessenger::SetNewValue().

```
{  
    SlowdownLength = length;  
}
```

5.3.2.23 void DetectorConstruction::SetSlowdownMaterial (G4String *materialChoice*)

Sets the slowdown material.

Parameters

<i>material-Choice</i>	The slowdown material.
------------------------	------------------------

Definition at line 795 of file DetectorConstruction.cc.

References SlowdownMaterial.

Referenced by DetectorMessenger::SetNewValue().

```
{  
  // search the material by its name  
  G4Material* pttoMaterial = G4Material::GetMaterial(materialChoice);  
  if (pttoMaterial) SlowdownMaterial = pttoMaterial;  
}
```

5.3.2.24 void DetectorConstruction::SetSlowdownThickness (G4double *thickness*)

Sets the slowdown thickness.

Parameters

<i>thickness</i>	The slowdown thickness
------------------	------------------------

Definition at line 813 of file DetectorConstruction.cc.

References SlowdownThickness.

Referenced by DetectorMessenger::SetNewValue().

```
{  
  SlowdownThickness = thickness;  
}
```

5.3.2.25 void DetectorConstruction::SetTopShieldingThickness (G4double *thickness*)

Sets the top shielding thickness.

Parameters

<i>thickness</i>	The shielding thickness
------------------	-------------------------

Definition at line 888 of file DetectorConstruction.cc.

References TopShieldingThickness.

Referenced by DetectorMessenger::SetNewValue().

```
{  
  TopShieldingThickness = thickness;  
}
```

The documentation for this class was generated from the following files:

- [DetectorConstruction.hh](#)
- [DetectorConstruction.cc](#)

5.4 PrimaryGeneratorMessenger::DetectorConstruction Class Reference

This class constructs the detector, according to specifications.

```
#include <PrimaryGeneratorMessenger.hh>
```

Public Member Functions

- [DetectorConstruction](#) ()
Constructor, constructs the object.
- [~DetectorConstruction](#) ()
Destructor, destroys the object.
- void **SetDefaultDetectorParameters** ()
- void [SetScintillatorMaterial](#) (G4String)
Sets the scintillator material.
- void [SetScintillatorDiameter](#) (G4double)
Sets the scintillator diameter.
- void [SetShieldingMaterial](#) (G4String)
Sets the shielding material.
- void [SetGeDetectorShellMaterial](#) (G4String)
Sets the shell material of the Germanium detector.
- void [SetBeamBlockerMaterial](#) (G4String)
Sets the beam blocker material.
- void [SetShieldingBeamBlockerMaterial](#) (G4String)
Sets the shielding beam blocker material.
- void [SetGermaniumDetectorMaterial](#) (G4String)
Sets the Germanium detector material.
- void [SetEffectiveSlowdownThickness](#) (G4double thickness)
Sets the effective slowdown thickness.
- void [SetSlowdownLength](#) (G4double length)
Sets the slowdown length.
- void [SetSlowdownThickness](#) (G4double thickness)
Sets the slowdown thickness.
- void [SetShieldingThickness](#) (G4double thickness)
Sets the shielding thickness.
- void [SetTopShieldingThickness](#) (G4double thickness)
Sets the top shielding thickness.
- void [SetBeamBlockerThickness](#) (G4double thickness)
Sets beamblocker thickness.
- void [SetShieldingBeamBlockerThickness](#) (G4double thickness)
Sets shielding beam blocker thickness.
- void [SetGeDetectorRadius](#) (G4double radius)

- Set Ge detector radius.*

 - void [SetGeDetectorShellLength](#) (G4double length)

Set Ge detector shell length.
- void [SetGeDetectorLength](#) (G4double length)

Set Ge detector length.
- void [SetGeDetectorShellInnerRadius](#) (G4double radius)

Set Ge detector shell inner radius.
- void [SetGeDetectorShellOuterRadius](#) (G4double radius)

Set Ge detector shell outer radius.
- void [SetGapMaterial](#) (G4String)

Sets the gap material.
- void [SetScintillatorShieldingGapThickness](#) (G4double)

Sets the gap thickness.
- void [SetSlowdownMaterial](#) (G4String)

Sets the slowdown material.
- void [SetSlowdownHeightGap](#) (G4double)

Sets the slowdown height gap.
- void [SetNumberOfScintillatorBoxes](#) (G4int)

Sets the number of scintillator boxes.
- void [SetNumberOfScintillatorsPerBox](#) (G4int)

Sets the number of layers of scintillators.
- void [SetBetaFlipping](#) (G4bool)

Sets beta detector flipping.
- G4VPhysicalVolume * [Construct](#) ()

Constructs the detector.
- void [UpdateGeometry](#) ()

Updates the geometry of the detector.
- G4double [GetWorldSizeX](#) ()

Returns the X size of the world.
- G4double [GetWorldSizeY](#) ()

Returns the Y size of the world.
- G4double [GetWorldSizeZ](#) ()

Returns the Z size of the world.
- G4double [GetScintillatorBoxSizeX](#) ()

<Returns the X size of the scintillator box.
- G4double [GetScintillatorBoxSizeY](#) ()

Returns the Y size of the scintillator box.
- G4double [GetScintillatorBoxSizeZ](#) ()

Returns the Z size of the scintillator box.
- G4double [GetBackScintillatorPlateSizeY](#) ()

Returns Y size of back scintillator plate.
- G4int [GetNumberOfScintillatorBoxes](#) ()
- G4int [GetNumberOfScintillatorsPerBox](#) ()

Returns the number of layers.

- G4Material * [GetScintillatorMaterial](#) ()

Returns the scintillator material.

- G4double [GetScintillatorDiameter](#) ()

Returns the scintillator diameter.

- G4Material * [GetGapMaterial](#) ()

Returns the gap material.

- G4double [GetScintillatorShieldingGapThickness](#) ()

Returns the gap thickness.

- const G4VPhysicalVolume * [GetphysiWorld](#) ()

Returns the physical world.

- const G4VPhysicalVolume * [GetAbsorber](#) ()

Returns the absorber.

- const G4VPhysicalVolume * [GetGap](#) ()

Returns the gap.

- const G4VPhysicalVolume * [GetScintillator](#) ()

Returns the scintillator.

- const G4VPhysicalVolume * [GetUpperBetaDetector](#) ()

Returns the upper beta detector.

- const G4VPhysicalVolume * [GetLowerBetaDetector](#) ()

Returns the lower Beta detector.

- const G4VPhysicalVolume * [GetGermaniumDetector](#) ()

Returns the germanium detector.

- const G4VPhysicalVolume * [GetUpperFrontScintillatorPlate](#) ()

Returns the upper front scintillator plate.

- const G4VPhysicalVolume * [GetLowerFrontScintillatorPlate](#) ()

Returns the lower front scintillator plate.

- const G4VPhysicalVolume * [GetBackScintillatorPlate](#) ()

Returns the back scintillator plate.

Private Member Functions

- void **ConstructWorld** ()
- void **ConstructBetaDetector** ()
- void **ConstructScintillatorBox** ()
- void **ConstructScintillatorSlot** ()
- void **ConstructScintillator** ()
- void **ConstructShielding** ()
- void **ConstructBeamBlocker** ()
- void **ConstructScintillatorPlates** ()
- void **ConstructSlowdown** ()
- void **ConstructGermaniumDetector** ()
- void **ConstructShieldingBeamBlocker** ()
- void **SetVisualizationAttributes** ()

- void [DefineMaterials](#) ()
Defines the materials.
- void [ComputeDependentDetectorParameters](#) ()
Computes the scintillator parameters.
- G4VPhysicalVolume * [ConstructDetector](#) ()
Constructs the detector.

Private Attributes

- G4Material * [ScintillatorMaterial](#)
Scintillator material.
- G4double [ScintillatorThickness](#)
Scintillator thickness.
- G4Material * [GapMaterial](#)
Gap material.
- G4double [ScintillatorShieldingGapThickness](#)
Gap thickness.
- G4int [NumberOfScintillatorsPerBox](#)
Number of layers of scintillators (typically 10).
- G4int [NumberOfScintillatorBoxes](#)
Number of scintillator boxes (typically 2).
- G4double [ScintillatorBoxSizeX](#)
Extent in X-direction of the scintillator box.
- G4double [ScintillatorBoxSizeY](#)
Extent in Y-direction of the scintillator box.
- G4double [ScintillatorBoxSizeZ](#)
Extent in Z-direction of the scintillator box.
- G4double [SlotWidthY](#)
Width of slot in Y-direction.
- G4double [SlotWidthZ](#)
Width of slot.
- G4double [ScintillatorDiameter](#)
Diameter of scintillator.
- G4double [ScintillatorLength](#)
Length of scintillator.
- G4double [ScintillatorPlateThickness](#)
Thickness of scintillator plate.
- G4double [FrontScintillatorPlateSizeY](#)
Extent in Y-direction of front scintillator plate.
- G4double [FrontScintillatorPlateSizeZ](#)
Extent in Z-direction of front scintillator plate.
- G4double [BackScintillatorPlateSizeY](#)
Extent in X-direction of back scintillator plate.

- G4double [GeDetectorRadius](#)
Radius of Ge detector.
- G4double [GeDetectorLength](#)
Length of Ge detector.
- G4double [GeDetectorShellLength](#)
Length of Ge detector shell.
- G4double [GeDetectorShellInnerRadius](#)
Inner radius of Ge detector shell.
- G4double [GeDetectorShellOuterRadius](#)
Outer radius of Ge detector shell.
- G4double [BeamBlockerThickness](#)
Thickness of beam blocker.
- G4double [ShieldingBeamBlockerThickness](#)
Shielding beam blocker thickness.
- G4double [GeDetectorContainerSizeX](#)
Size of the detector container in the Z-direction.
- G4Material * [defaultMaterial](#)
Default material.
- G4double [WorldSizeZ](#)
Size of world in the Z-direction.
- G4double [WorldSizeY](#)
Size of world in the Y-direction.
- G4double [WorldSizeX](#)
Size of world in the X-direction.
- G4Material * [ShieldingMaterial](#)
Shielding material.
- G4double [ShieldingThickness](#)
Thickness of shielding.
- G4double [TopShieldingThickness](#)
Thickness of top shielding.
- G4Material * [BeamBlockerMaterial](#)
The beam blocker material.
- G4Material * [ShieldingBeamBlockerMaterial](#)
Shielding beam blocker material.
- G4Box * [solidWorld](#)
Pointer to the solid World.
- G4LogicalVolume * [logicWorld](#)
Pointer to the logical World.
- G4VPhysicalVolume * [physiWorld](#)
Pointer to the physical World.
- G4Material * [GeDetectorShellMaterial](#)
The Ge detector shell material.
- G4Material * [GermaniumDetectorMaterial](#)

The Ge detector material.

- G4Box * [solidScintillatorBox](#)
Pointer to the solid Calor.
- G4LogicalVolume * [logicScintillatorBox](#)
Pointer to the logical Calor.
- G4VPhysicalVolume * [physiScintillatorBox](#)
Pointer to the physical Calor.
- G4Box * [solidSlot](#)
Pointer to the solid Layer.
- G4LogicalVolume * [logicSlot](#)
Pointer to the logical Layer.
- G4VPhysicalVolume * [physiSlot](#)
Pointer to the physical Layer.
- G4Tubs * [solidScintillator](#)
Pointer to the solid Scintillator.
- G4LogicalVolume * [logicScintillator](#)
Pointer to the logical Scintillator.
- G4VPhysicalVolume * [physiScintillator](#)
Pointer to the physical Scintillator.
- G4Box * [solidGap](#)
Pointer to the solid Gap.
- G4LogicalVolume * [logicGap](#)
Pointer to the logical Gap.
- G4VPhysicalVolume * [physiGap](#)
Pointer to the physical Gap.
- G4Box * [solidShielding](#)
Pointer to the solid shielding.
- G4LogicalVolume * [logicShielding](#)
Pointer to the logical shielding.
- G4VPhysicalVolume * [physiShielding](#)
Pointer to the physical shielding.
- G4Box * [solidTopShielding](#)
Pointer to the solid shielding.
- G4LogicalVolume * [logicTopShielding](#)
Pointer to the logical shielding.
- G4VPhysicalVolume * [physiTopShielding](#)
Pointer to the physical shielding.
- G4Box * [solidShieldingBeamBlocker](#)
Pointer to the solid shielding.
- G4LogicalVolume * [logicShieldingBeamBlocker](#)
Pointer to the logical shielding.
- G4VPhysicalVolume * [physiShieldingBeamBlocker](#)
Pointer to the physical shielding.

- G4Box * [solidTopShieldingBeamBlocker](#)
Pointer to the solid shielding.
- G4LogicalVolume * [logicTopShieldingBeamBlocker](#)
Pointer to the logical shielding.
- G4VPhysicalVolume * [physiTopShieldingBeamBlocker](#)
Pointer to the physical shielding.
- G4Box * [solidBetaDetector](#)
Pointer to the solid UpperBetaDetector.
- G4LogicalVolume * [logicBetaDetector](#)
Pointer to the logical UpperBetaDetector.
- G4VPhysicalVolume * [physiUpperBetaDetector](#)
Pointer to the physical UpperBetaDetector.
- G4Box * [solidLowerBetaDetector](#)
Pointer to the solid LowerBetaDetector.
- G4LogicalVolume * [logicLowerBetaDetector](#)
Pointer to the logical LowerBetaDetector.
- G4VPhysicalVolume * [physiLowerBetaDetector](#)
Pointer to the physical LowerBetaDetector.
- [DetectorMessenger](#) * [detectorMessenger](#)
Pointer to the Messenger.
- G4Box * [solidFrontScintillatorPlate](#)
Pointer to the solid FrontScintillatorPlate.
- G4LogicalVolume * [logicFrontScintillatorPlate](#)
Pointer to the logical FrontScintillatorPlate.
- G4VPhysicalVolume * [physiUpperFrontScintillatorPlate](#)
Pointer to the physical UpperFrontScintillatorPlate.
- G4VPhysicalVolume * [physiLowerFrontScintillatorPlate](#)
Pointer to the physical LowerFrontScintillatorPlate.
- G4Box * [solidBackScintillatorPlate](#)
Pointer to the solid BackScintillatorPlate.
- G4LogicalVolume * [logicBackScintillatorPlate](#)
Pointer to the logical BackScintillatorPlate.
- G4VPhysicalVolume * [physiBackScintillatorPlate](#)
Pointer to the physical BackScintillatorPlate.
- G4double [SlowdownThickness](#)
Thickness of slowdown.
- G4double [SlowdownLength](#)
Length of slowdown.
- G4double [SlowdownHeightGap](#)
Peight of slowdown gap.
- G4double [EffectiveSlowdownThickness](#)
The thickness that the particles from the gun meet.
- G4Trap * [solidSlowdown](#)

- Pointer to the solid FrontScintillatorPlate.*
- G4LogicalVolume * [logicSlowdown](#)
- Pointer to the logical Slowdown.*
- G4VPhysicalVolume * [physiUpperSlowdown](#)
- Pointer to the physical UpperSlowdown.*
- G4VPhysicalVolume * [physiLowerSlowdown](#)
- Pointer to the physical LowerSlowdown.*
- G4Material * [SlowdownMaterial](#)
- The slowdown material.*
- G4Tubs * [solidGermaniumDetector](#)
- Pointer to the solid germanium detector.*
- G4LogicalVolume * [logicGermaniumDetector](#)
- Pointer to the logical germanium detector.*
- G4VPhysicalVolume * [physiGermaniumDetector](#)
- Pointer to the physical germanium detector.*
- G4Tubs * [solidGeDetectorShell](#)
- Pointer to the solid germanium detector shell.*
- G4LogicalVolume * [logicGeDetectorShell](#)
- Pointer to the logical germanium detector shell.*
- G4VPhysicalVolume * [physiGeDetectorShell](#)
- Pointer to the physical germanium detector shell.*
- G4Box * [solidGeDetectorContainer](#)
- Pointer to the solid germanium detector container.*
- G4LogicalVolume * [logicGeDetectorContainer](#)
- Pointer to the logical germanium detector container.*
- G4VPhysicalVolume * [physiGeDetectorContainer](#)
- Pointer to the physical germanium detector container.*
- G4Tubs * [solidGeDetectorShellTop](#)
- Pointer to the solid germanium detector shell.*
- G4LogicalVolume * [logicGeDetectorShellTop](#)
- Pointer to the logical germanium detector shell.*
- G4VPhysicalVolume * [physiGeDetectorShellTop](#)
- Pointer to the physical germanium detector shell.*
- G4Tubs * [solidGeDetectorShellBottom](#)
- Pointer to the solid germanium detector shell.*
- G4LogicalVolume * [logicGeDetectorShellBottom](#)
- Pointer to the logical germanium detector shell.*
- G4VPhysicalVolume * [physiGeDetectorShellBottom](#)
- Pointer to the physical germanium detector shell.*
- G4Box * [solidBeamBlocker](#)
- Pointer to the solid BeamBlocker.*
- G4LogicalVolume * [logicBeamBlocker](#)
- Pointer to the logical BeamBlocker.*

- G4VPhysicalVolume * [physiBeamBlocker](#)
Pointer to the physical BeamBlocker.
- G4double [BetaDetectorPositionY](#)
Y position of the Beta detectors.
- G4double [BeamBlockerYWidth](#)
Y width of beam blocker.
- G4double [BeamBlockerZWidth](#)
Z width of beam blocker.
- G4double [BeamBlockerXPosition](#)
X position of beam blocker.
- G4double **ScintillatorPositionX**
- G4double **ScintillatorPositionY**
- G4double **ScintillatorPositionZ**
- G4double **ShieldingPositionX**
- G4double **ShieldingPositionY**
- G4double **ShieldingPositionZ**
- G4double **TopShieldingPositionX**
- G4double **TopShieldingPositionY**
- G4double **TopShieldingPositionZ**
- G4double **PlateShieldingSpacing**
- G4double **ShieldingBeamBlockerPositionX**
- G4bool **FlipBeta**

5.4.1 Detailed Description

This class constructs the detector, according to specifications.

Definition at line 54 of file PrimaryGeneratorMessenger.hh.

5.4.2 Member Function Documentation

5.4.2.1 G4double PrimaryGeneratorMessenger::DetectorConstruction::GetBackScintillatorPlateSizeY
() [inline]

Returns Y size of back scintillator plate.

Returns the number of scintillator boxes.

Definition at line 165 of file PrimaryGeneratorMessenger.hh.

5.4.2.2 void PrimaryGeneratorMessenger::DetectorConstruction::SetBeamBlockerThickness (G4double *thickness*)

Sets beamblocker thickness.

Parameters

<i>thickness</i>	BeamBlocker thickness.
------------------	------------------------

5.4.2.3 void PrimaryGeneratorMessenger::DetectorConstruction::SetEffectiveSlowdownThickness (G4double *thickness*)

Sets the effective slowdown thickness.

Parameters

<i>thickness</i>	The thickness of effective slowdown.
------------------	--------------------------------------

5.4.2.4 void PrimaryGeneratorMessenger::DetectorConstruction::SetGeDetectorLength (G4double *length*)

Set Ge detector length.

Parameters

<i>length</i>	Length
---------------	--------

5.4.2.5 void PrimaryGeneratorMessenger::DetectorConstruction::SetGeDetectorRadius (G4double *radius*)

Set Ge detector radius.

Parameters

<i>radius</i>	Detector radius.
---------------	------------------

5.4.2.6 void PrimaryGeneratorMessenger::DetectorConstruction::SetGeDetectorShellInnerRadius (G4double *radius*)

Set Ge detector shell inner radius.

Parameters

<i>radius</i>	Detector radius
---------------	-----------------

5.4.2.7 void PrimaryGeneratorMessenger::DetectorConstruction::SetGeDetectorShellLength (G4double *length*)

Set Ge detector shell length.

Parameters

<i>length</i>	Length
---------------	--------

5.4.2.8 void PrimaryGeneratorMessenger::DetectorConstruction::SetGeDetectorShellOuterRadius (G4double *radius*)

Set Ge detector shell outer radius.

Parameters

<i>radius</i>	Radius.
---------------	---------

5.4.2.9 void PrimaryGeneratorMessenger::DetectorConstruction::SetShieldingBeamBlockerThickness (G4double *thickness*)

Sets shielding beam blocker thickness.

Parameters

<i>thickness</i>	Shielding beam blocker thickness
------------------	----------------------------------

5.4.2.10 void PrimaryGeneratorMessenger::DetectorConstruction::SetShieldingThickness (G4double *thickness*)

Sets the shielding thickness.

Parameters

<i>thickness</i>	The shielding thickness
------------------	-------------------------

5.4.2.11 void PrimaryGeneratorMessenger::DetectorConstruction::SetSlowdownLength (G4double *length*)

Sets the slowdown length.

Parameters

<i>length</i>	The slowdown length
---------------	---------------------

5.4.2.12 void PrimaryGeneratorMessenger::DetectorConstruction::SetSlowdownThickness (G4double *thickness*)

Sets the slowdown thickness.

Parameters

<i>thickness</i>	The slowdown thickness
------------------	------------------------

5.4.2.13 void PrimaryGeneratorMessenger::DetectorConstruction::SetTopShieldingThickness (G4double *thickness*)

Sets the top shielding thickness.

Parameters

<i>thickness</i>	The shielding thickness
------------------	-------------------------

The documentation for this class was generated from the following file:

- [DetectorConstruction.hh](#)

5.5 DetectorMessenger Class Reference

Creates commands that the user can invoke via the command line.

```
#include <DetectorMessenger.hh>
```

Public Member Functions

- [DetectorMessenger](#) ([DetectorConstruction](#) *)
Constructor, constructs the object.
- [~DetectorMessenger](#) ()
Destructor, destroys the object.
- void [SetNewValue](#) (G4UIcommand *, G4String)
This method is invoked when the user (or a script, etc.) enters a command.

Private Attributes

- [DetectorConstruction](#) * [Detector](#)
The detector associated with this messenger.
- G4UIdirectory * [PIBIDSDir](#)
Directory containing some commands wich can be invoked.
- G4UIdirectory * [detDir](#)
Directory containing some commands wich can be invoked.
- G4UIcmdWithAString * [ScintillatorMaterialCmd](#)
A command which can be invoked.
- G4UIcmdWithAString * [GapMaterCmd](#)
A command which can be invoked.

- G4UlcmdWithADoubleAndUnit * [ScintillatorDiameterCmd](#)
A command which can be invoked.
- G4UlcmdWithADoubleAndUnit * [ScintillatorShieldingGapThicknessCmd](#)
A command which can be invoked.
- G4UlcmdWithAnInteger * [NumberOfScintillatorsPerBoxCmd](#)
A command which can be invoked.
- G4UlcmdWithAnInteger * [NumberOfScintillatorBoxesCmd](#)
Number of scintillator boxes.
- G4UlcmdWithoutParameter * [UpdateCmd](#)
A command which can be invoked.
- G4UlcmdWithADoubleAndUnit * [EffectiveSlowdownThicknessCmd](#)
Set the effective slowdown thickness.
- G4UlcmdWithADoubleAndUnit * [SlowdownLengthCmd](#)
Set the slowdown length.
- G4UlcmdWithADoubleAndUnit * [SlowdownThicknessCmd](#)
Set the slowdown thickness.
- G4UlcmdWithADoubleAndUnit * [BeamBlockerThicknessCmd](#)
Sets the beam blocker thickness.
- G4UlcmdWithADoubleAndUnit * [ShieldingBeamBlockerThicknessCmd](#)
Sets the beam blocker thickness.
- G4UlcmdWithADoubleAndUnit * [ShieldingThicknessCmd](#)
Sets the shielding thickness.
- G4UlcmdWithADoubleAndUnit * [TopShieldingThicknessCmd](#)
Sets the shielding thickness.
- G4UlcmdWithADoubleAndUnit * **GeDetectorRadiusCmd**
- G4UlcmdWithADoubleAndUnit * **GeDetectorShellLengthCmd**
- G4UlcmdWithADoubleAndUnit * **GeDetectorLengthCmd**
- G4UlcmdWithADoubleAndUnit * **GeDetectorShellInnerRadiusCmd**
- G4UlcmdWithADoubleAndUnit * **GeDetectorShellOuterRadiusCmd**
- G4UlcmdWithAString * [ShieldingMaterialCmd](#)
A command which can be invoked.
- G4UlcmdWithAString * **GeDetectorShellMaterialCmd**
- G4UlcmdWithAString * **GermaniumDetectorMaterialCmd**
- G4UlcmdWithAString * **BeamBlockerMaterialCmd**
- G4UlcmdWithAString * **ShieldingBeamBlockerMaterialCmd**
- G4UlcmdWithAString * **SlowdownMaterialCmd**
- G4UlcmdWithABool * **FlipBetaCmd**

5.5.1 Detailed Description

Creates commands that the user can invoke via the command line.

Definition at line 35 of file DetectorMessenger.hh.

5.5.2 Constructor & Destructor Documentation

5.5.2.1 DetectorMessenger::DetectorMessenger (DetectorConstruction * Det)

Constructor, constructs the object.

Parameters

<i>Det</i>	The detector for which we are constructing the messenger.
------------	---

Definition at line 3 of file DetectorMessenger.cc.

References BeamBlockerThicknessCmd, detDir, EffectiveSlowdownThicknessCmd, GapMaterCmd, NumberOfScintillatorBoxesCmd, NumberOfScintillatorsPerBoxCmd, PIBIDSDir, ScintillatorDiameterCmd, ScintillatorMaterialCmd, ScintillatorShieldingGapThicknessCmd, ShieldingBeamBlockerThicknessCmd, ShieldingMaterialCmd, ShieldingThicknessCmd, SlowdownLengthCmd, SlowdownThicknessCmd, TopShieldingThicknessCmd, and UpdateCmd.

```
:Detector(Det)
{
    PIBIDSDir = new G4UIdirectory("/PIBIDS/");
    PIBIDSDir->SetGuidance("UI commands for the PIBIDS simulation.");

    detDir = new G4UIdirectory("/PIBIDS/det/");
    detDir->SetGuidance("Detector properties control.");

    ScintillatorMaterialCmd = new G4UIcmdWithAString("/PIBIDS/det/setAbsMat",this);

    ScintillatorMaterialCmd->SetGuidance("Set material in the scintillators.");
    ScintillatorMaterialCmd->SetParameterName("choice",false);
    ScintillatorMaterialCmd->AvailableForStates(G4State_PreInit,G4State_Idle);

    GapMaterCmd = new G4UIcmdWithAString("/PIBIDS/det/setGapMat",this);
    GapMaterCmd->SetGuidance("Select material of the gap between the scintillators.");
    GapMaterCmd->SetParameterName("choice",false);
    GapMaterCmd->AvailableForStates(G4State_PreInit,G4State_Idle);

    ScintillatorDiameterCmd = new G4UIcmdWithADoubleAndUnit("/PIBIDS/det/setScintillatorDiameter",this);
    ScintillatorDiameterCmd->SetGuidance("Set the scintillator diameter.");
    ScintillatorDiameterCmd->SetParameterName("Size",false);
    ScintillatorDiameterCmd->SetRange("Size>=0.");
    ScintillatorDiameterCmd->SetUnitCategory("Length");
    ScintillatorDiameterCmd->AvailableForStates(G4State_PreInit,G4State_Idle);

    ScintillatorShieldingGapThicknessCmd = new G4UIcmdWithADoubleAndUnit("/PIBIDS/det/setScintillatorShieldingGapThickness",this);
    ScintillatorShieldingGapThicknessCmd->SetGuidance("Set thickness of the gap between scintillators.");
    ScintillatorShieldingGapThicknessCmd->SetParameterName("Size",false);
    ScintillatorShieldingGapThicknessCmd->SetRange("Size>=0.");
    ScintillatorShieldingGapThicknessCmd->SetUnitCategory("Length");
    ScintillatorShieldingGapThicknessCmd->AvailableForStates(G4State_PreInit,G4State_Idle);

    NumberOfScintillatorsPerBoxCmd = new G4UIcmdWithAnInteger("/PIBIDS/det/setNumbe
```

```

        rOfScintillatorsPerBox",this);
NumberOfScintillatorsPerBoxCmd->SetGuidance("Set number of layers of scintillat
    ors.");
NumberOfScintillatorsPerBoxCmd->SetParameterName("NbLayers",false);
NumberOfScintillatorsPerBoxCmd->SetRange("NbLayers>0 && NbLayers<500");
NumberOfScintillatorsPerBoxCmd->AvailableForStates(G4State_PreInit,G4State_Idle
    );

NumberOfScintillatorBoxesCmd = new G4UICmdWithAnInteger("/PIBIDS/det/setNumberO
    fScintillatorBoxes",this);
NumberOfScintillatorBoxesCmd->SetGuidance("Set number of scintillator boxes.");

NumberOfScintillatorBoxesCmd->SetParameterName("NbLayers",false);
NumberOfScintillatorBoxesCmd->SetRange("NbLayers>0 && NbLayers<20");
NumberOfScintillatorBoxesCmd->AvailableForStates(G4State_PreInit,G4State_Idle);

EffectiveSlowdownThicknessCmd = new G4UICmdWithADoubleAndUnit("/PIBIDS/det/setE
    ffectiveSlowdownThickness",this);
EffectiveSlowdownThicknessCmd->SetGuidance("Set effective slowdown thickness of
    wedges.");
EffectiveSlowdownThicknessCmd->SetParameterName("Thickness",false);
EffectiveSlowdownThicknessCmd->SetRange("Thickness>0.");
EffectiveSlowdownThicknessCmd->SetUnitCategory("Length");
EffectiveSlowdownThicknessCmd->AvailableForStates(G4State_PreInit,G4State_Idle)
    ;

SlowdownLengthCmd = new G4UICmdWithADoubleAndUnit("/PIBIDS/det/setSlowdownLengt
    h",this);
SlowdownLengthCmd->SetGuidance("Set slowdown (wedges) length.");
SlowdownLengthCmd->SetParameterName("Length",false);
SlowdownLengthCmd->SetRange("Length>0.");
SlowdownLengthCmd->SetUnitCategory("Length");
SlowdownLengthCmd->AvailableForStates(G4State_PreInit,G4State_Idle);

GeDetectorRadiusCmd = new G4UICmdWithADoubleAndUnit("/PIBIDS/det/setGeDetectorR
    adius",this);
GeDetectorRadiusCmd->SetGuidance("Set Ge detector radius.");
GeDetectorRadiusCmd->SetParameterName("Radius",false);
GeDetectorRadiusCmd->SetRange("Radius>0.");
GeDetectorRadiusCmd->SetUnitCategory("Length");
GeDetectorRadiusCmd->AvailableForStates(G4State_PreInit,G4State_Idle);

GeDetectorShellLengthCmd = new G4UICmdWithADoubleAndUnit("/PIBIDS/det/setGeDete
    ctorShellLength",this);
GeDetectorShellLengthCmd->SetGuidance("Set Ge detector shell length.");
GeDetectorShellLengthCmd->SetParameterName("Length",false);
GeDetectorShellLengthCmd->SetRange("Length>0.");
GeDetectorShellLengthCmd->SetUnitCategory("Length");
GeDetectorShellLengthCmd->AvailableForStates(G4State_PreInit,G4State_Idle);

GeDetectorLengthCmd = new G4UICmdWithADoubleAndUnit("/PIBIDS/det/setGeDetectorL
    ength",this);
GeDetectorLengthCmd->SetGuidance("Set Ge detector length.");
GeDetectorLengthCmd->SetParameterName("Length",false);
GeDetectorLengthCmd->SetRange("Length>0.");
GeDetectorLengthCmd->SetUnitCategory("Length");
GeDetectorLengthCmd->AvailableForStates(G4State_PreInit,G4State_Idle);

GeDetectorShellInnerRadiusCmd = new G4UICmdWithADoubleAndUnit("/PIBIDS/det/setG
    eDetectorShellInnerRadius",this);

```

```

GeDetectorShellInnerRadiusCmd->SetGuidance("Set Ge detector shell inner radius"
);
GeDetectorShellInnerRadiusCmd->SetParameterName("Radius", false);
GeDetectorShellInnerRadiusCmd->SetRange("Radius>0.");
GeDetectorShellInnerRadiusCmd->SetUnitCategory("Length");
GeDetectorShellInnerRadiusCmd->AvailableForStates(G4State_PreInit, G4State_Idle)
;

GeDetectorShellOuterRadiusCmd = new G4UIcmdWithADoubleAndUnit("/PIBIDS/det/setGeDetectorShellOuterRadius", this);
GeDetectorShellOuterRadiusCmd->SetGuidance("Set Ge detector shell outer radius.
");
GeDetectorShellOuterRadiusCmd->SetParameterName("Radius", false);
GeDetectorShellOuterRadiusCmd->SetRange("Radius>0.");
GeDetectorShellOuterRadiusCmd->SetUnitCategory("Length");
GeDetectorShellOuterRadiusCmd->AvailableForStates(G4State_PreInit, G4State_Idle)
;

SlowdownThicknessCmd = new G4UIcmdWithADoubleAndUnit("/PIBIDS/det/setSlowdownThickness", this);
SlowdownThicknessCmd->SetGuidance("Set slowdown (wedges) length.");
SlowdownThicknessCmd->SetParameterName("Thickness", false);
SlowdownThicknessCmd->SetRange("Thickness>0.");
SlowdownThicknessCmd->SetUnitCategory("Length");
SlowdownThicknessCmd->AvailableForStates(G4State_PreInit, G4State_Idle);

BeamBlockerThicknessCmd = new G4UIcmdWithADoubleAndUnit("/PIBIDS/det/setBeamBlockerThickness", this);
BeamBlockerThicknessCmd->SetGuidance("Set beam blocker thickness.");
BeamBlockerThicknessCmd->SetParameterName("Thickness", false);
BeamBlockerThicknessCmd->SetRange("Thickness>0.");
BeamBlockerThicknessCmd->SetUnitCategory("Length");
BeamBlockerThicknessCmd->AvailableForStates(G4State_PreInit, G4State_Idle);

ShieldingBeamBlockerThicknessCmd = new G4UIcmdWithADoubleAndUnit("/PIBIDS/det/setShieldingBeamBlockerThickness", this);
ShieldingBeamBlockerThicknessCmd->SetGuidance("Set beam blocker thickness.");
ShieldingBeamBlockerThicknessCmd->SetParameterName("Thickness", false);
ShieldingBeamBlockerThicknessCmd->SetRange("Thickness>0.");
ShieldingBeamBlockerThicknessCmd->SetUnitCategory("Length");
ShieldingBeamBlockerThicknessCmd->AvailableForStates(G4State_PreInit, G4State_Idle);

ShieldingThicknessCmd = new G4UIcmdWithADoubleAndUnit("/PIBIDS/det/setShieldingThickness", this);
ShieldingThicknessCmd->SetGuidance("Set shielding thickness.");
ShieldingThicknessCmd->SetParameterName("Thickness", false);
ShieldingThicknessCmd->SetRange("Thickness>0.");
ShieldingThicknessCmd->SetUnitCategory("Length");
ShieldingThicknessCmd->AvailableForStates(G4State_PreInit, G4State_Idle);

TopShieldingThicknessCmd = new G4UIcmdWithADoubleAndUnit("/PIBIDS/det/setTopShieldingThickness", this);
TopShieldingThicknessCmd->SetGuidance("Set shielding thickness.");

```

```

TopShieldingThicknessCmd->SetParameterName("Thickness", false);
TopShieldingThicknessCmd->SetRange("Thickness>0.");
TopShieldingThicknessCmd->SetUnitCategory("Length");
TopShieldingThicknessCmd->AvailableForStates(G4State_PreInit, G4State_Idle);

ShieldingMaterialCmd = new G4UIcmdWithAString("/PIBIDS/det/setShieldingMaterial", this);
ShieldingMaterialCmd->SetGuidance("Set the shielding material.");
ShieldingMaterialCmd->SetParameterName("choice", false);
ShieldingMaterialCmd->AvailableForStates(G4State_PreInit, G4State_Idle);

GeDetectorShellMaterialCmd = new G4UIcmdWithAString("/PIBIDS/det/setGeDetectorShellMaterial", this);
GeDetectorShellMaterialCmd->SetGuidance("Set the germanium detector shell material.");
GeDetectorShellMaterialCmd->SetParameterName("choice", false);
GeDetectorShellMaterialCmd->AvailableForStates(G4State_PreInit, G4State_Idle);

GermaniumDetectorMaterialCmd = new G4UIcmdWithAString("/PIBIDS/det/setGermaniumDetectorMaterial", this);
GermaniumDetectorMaterialCmd->SetGuidance("Set the germanium detector material.");
GermaniumDetectorMaterialCmd->SetParameterName("choice", false);
GermaniumDetectorMaterialCmd->AvailableForStates(G4State_PreInit, G4State_Idle);

BeamBlockerMaterialCmd = new G4UIcmdWithAString("/PIBIDS/det/setBeamBlockerMaterial", this);
BeamBlockerMaterialCmd->SetGuidance("Set the beam blocker material.");
BeamBlockerMaterialCmd->SetParameterName("choice", false);
BeamBlockerMaterialCmd->AvailableForStates(G4State_PreInit, G4State_Idle);

ShieldingBeamBlockerMaterialCmd = new G4UIcmdWithAString("/PIBIDS/det/setShieldingBeamBlockerMaterial", this);
ShieldingBeamBlockerMaterialCmd->SetGuidance("Set the beam blocker material.");

ShieldingBeamBlockerMaterialCmd->SetParameterName("choice", false);
ShieldingBeamBlockerMaterialCmd->AvailableForStates(G4State_PreInit, G4State_Idle);

SlowdownMaterialCmd = new G4UIcmdWithAString("/PIBIDS/det/setSlowdownMaterial", this);
SlowdownMaterialCmd->SetGuidance("Set the slowdown material.");
SlowdownMaterialCmd->SetParameterName("choice", false);
SlowdownMaterialCmd->AvailableForStates(G4State_PreInit, G4State_Idle);

FlipBetaCmd = new G4UIcmdWithABool("/PIBIDS/det/flipBeta", this);
FlipBetaCmd->SetGuidance("Use this to switch beta detector flipping on or off.");
FlipBetaCmd->SetParameterName("Flip", false);

UpdateCmd = new G4UIcmdWithoutParameter("/PIBIDS/det/update", this);
UpdateCmd->SetGuidance("Update detector geometry.");
UpdateCmd->SetGuidance("This command MUST be applied before \"beamOn\" ");

```

```

UpdateCmd->SetGuidance("if you changed geometrical value(s).");
UpdateCmd->AvailableForStates(G4State_Idle);
}

```

5.5.3 Member Function Documentation

5.5.3.1 void DetectorMessenger::SetNewValue (G4UIcommand * *command*, G4String *newValue*)

This method is invoked when the user (or a script, etc.) enters a command.

Parameters

<i>command</i>	The command invoked.
<i>newValue</i>	The value which was sent as a parameter to the command.

Definition at line 219 of file DetectorMessenger.cc.

References BeamBlockerThicknessCmd, Detector, EffectiveSlowdownThicknessCmd, GapMaterCmd, NumberOfScintillatorBoxesCmd, NumberOfScintillatorsPerBoxCmd, ScintillatorDiameterCmd, ScintillatorMaterialCmd, ScintillatorShieldingGapThicknessCmd, DetectorConstruction::SetBeamBlockerMaterial(), DetectorConstruction::SetBeamBlockerThickness(), DetectorConstruction::SetBetaFlipping(), DetectorConstruction::SetEffectiveSlowdownThickness(), DetectorConstruction::SetGapMaterial(), DetectorConstruction::SetGeDetectorLength(), DetectorConstruction::SetGeDetectorRadius(), DetectorConstruction::SetGeDetectorShellInnerRadius(), DetectorConstruction::SetGeDetectorShellLength(), DetectorConstruction::SetGeDetectorShellMaterial(), DetectorConstruction::SetGeDetectorShellOuterRadius(), DetectorConstruction::SetGermaniumDetectorMaterial(), DetectorConstruction::SetNumberOfScintillatorBoxes(), DetectorConstruction::SetNumberOfScintillatorsPerBox(), DetectorConstruction::SetScintillatorDiameter(), DetectorConstruction::SetScintillatorMaterial(), DetectorConstruction::SetScintillatorShieldingGapThickness(), DetectorConstruction::SetShieldingBeamBlockerMaterial(), DetectorConstruction::SetShieldingBeamBlockerThickness(), DetectorConstruction::SetShieldingMaterial(), DetectorConstruction::SetShieldingThickness(), DetectorConstruction::SetSlowdownLength(), DetectorConstruction::SetSlowdownMaterial(), DetectorConstruction::SetSlowdownThickness(), DetectorConstruction::SetTopShieldingThickness(), ShieldingBeamBlockerThicknessCmd, ShieldingMaterialCmd, ShieldingThicknessCmd, SlowdownLengthCmd, SlowdownThicknessCmd, TopShieldingThicknessCmd, UpdateCmd, and DetectorConstruction::UpdateGeometry().

```

{
    if( command == ScintillatorMaterialCmd )
        { Detector->SetScintillatorMaterial(newValue); }

    if( command == GapMaterCmd )
        { Detector->SetGapMaterial(newValue); }

    if( command == ShieldingMaterialCmd )
        { Detector->SetShieldingMaterial(newValue); }

    if( command == GeDetectorShellMaterialCmd )
        { Detector->SetGeDetectorShellMaterial(newValue); }

    if( command == GermaniumDetectorMaterialCmd )
        { Detector->SetGermaniumDetectorMaterial(newValue); }
}

```

```

if( command == BeamBlockerMaterialCmd)
    {Detector->SetBeamBlockerMaterial(newValue);}

if( command == ShieldingBeamBlockerMaterialCmd)
    {Detector->SetShieldingBeamBlockerMaterial(newValue);}

if( command == SlowdownMaterialCmd)
    {Detector->SetSlowdownMaterial(newValue);}

if( command == ScintillatorDiameterCmd )
    { Detector->SetScintillatorDiameter(ScintillatorDiameterCmd
                                     ->GetNewDoubleValue(newValue)); }

if( command == ScintillatorShieldingGapThicknessCmd )
    { Detector->SetScintillatorShieldingGapThickness(
      ScintillatorShieldingGapThicknessCmd->GetNewDoubleValue(newValue)); }

if( command == NumberOfScintillatorsPerBoxCmd )
    { Detector->SetNumberOfScintillatorsPerBox(NumberOfScintillatorsPerBoxCmd->Get
      NewIntValue(newValue)); }

if(command == NumberOfScintillatorBoxesCmd)
    {      Detector->SetNumberOfScintillatorBoxes(NumberOfScintillatorBoxesCmd->G
      etNewIntValue(newValue));      }

if( command == EffectiveSlowdownThicknessCmd)
    { Detector->SetEffectiveSlowdownThickness(EffectiveSlowdownThicknessCmd->GetN
      ewDoubleValue(newValue)); }

if( command == SlowdownLengthCmd)
    { Detector->SetSlowdownLength(SlowdownLengthCmd->GetNewDoubleValue(newValue))
    ; }

if( command == GeDetectorRadiusCmd)
    { Detector->SetGeDetectorRadius(GeDetectorRadiusCmd->GetNewDoubleValue(newVal
      ue)); }

if( command == GeDetectorShellLengthCmd)
    { Detector->SetGeDetectorShellLength(GeDetectorShellLengthCmd->GetNewDoubleVa
      lue(newValue)); }

if( command == GeDetectorLengthCmd)
    { Detector->SetGeDetectorLength(GeDetectorLengthCmd->GetNewDoubleValue(newVal
      ue)); }

if( command == GeDetectorShellInnerRadiusCmd)
    { Detector->SetGeDetectorShellInnerRadius(GeDetectorShellInnerRadiusCmd->GetN
      ewDoubleValue(newValue)); }

if( command == GeDetectorShellOuterRadiusCmd)
    { Detector->SetGeDetectorShellOuterRadius(GeDetectorShellOuterRadiusCmd->GetN
      ewDoubleValue(newValue)); }

if( command == SlowdownThicknessCmd)
    { Detector->SetSlowdownThickness(SlowdownThicknessCmd->GetNewDoubleValue(newV
      alue)); }

if( command == BeamBlockerThicknessCmd)
    { Detector->SetBeamBlockerThickness(BeamBlockerThicknessCmd->GetNewDoubleValu
      e(newValue)); }

if( command == ShieldingBeamBlockerThicknessCmd)

```

```

    { Detector->SetShieldingBeamBlockerThickness(
      ShieldingBeamBlockerThicknessCmd->GetNewDoubleValue(newValue)); }

    if( command == ShieldingThicknessCmd)
    { Detector->SetShieldingThickness(ShieldingThicknessCmd->GetNewDoubleValue(ne
      wValue)); }

    if( command == TopShieldingThicknessCmd)
    { Detector->SetTopShieldingThickness(TopShieldingThicknessCmd->GetNewDoubleVa
      lue(newValue)); }

    if (command == FlipBetaCmd)
    {Detector->SetBetaFlipping(FlipBetaCmd->GetNewBoolValue(newValue)); }

    if( command == UpdateCmd )
    { Detector->UpdateGeometry(); }

}

```

The documentation for this class was generated from the following files:

- [DetectorMessenger.hh](#)
- [DetectorMessenger.cc](#)

5.6 PrimaryGeneratorMessenger::DetectorMessenger Class Reference

Creates commands that the user can invoke via the command line.

```
#include <PrimaryGeneratorMessenger.hh>
```

Public Member Functions

- [DetectorMessenger](#) ([DetectorConstruction](#) *)
Constructor, constructs the object.
- [~DetectorMessenger](#) ()
Destructor, destroys the object.
- void [SetNewValue](#) (G4UIcommand *, G4String)
This method is invoked when the user (or a script, etc.) enters a command.

Private Attributes

- [DetectorConstruction](#) * [Detector](#)
The detector associated with this messenger.
- G4UIDirectory * [PIBIDSDir](#)
Directory containing some commands wich can be invoked.
- G4UIDirectory * [detDir](#)
Directory containing some commands wich can be invoked.

- G4UlcmdWithAString * [ScintillatorMaterialCmd](#)
A command which can be invoked.
- G4UlcmdWithAString * [GapMaterCmd](#)
A command which can be invoked.
- G4UlcmdWithADoubleAndUnit * [ScintillatorDiameterCmd](#)
A command which can be invoked.
- G4UlcmdWithADoubleAndUnit * [ScintillatorShieldingGapThicknessCmd](#)
A command which can be invoked.
- G4UlcmdWithAnInteger * [NumberOfScintillatorsPerBoxCmd](#)
A command which can be invoked.
- G4UlcmdWithAnInteger * [NumberOfScintillatorBoxesCmd](#)
Number of scintillator boxes.
- G4UlcmdWithoutParameter * [UpdateCmd](#)
A command which can be invoked.
- G4UlcmdWithADoubleAndUnit * [EffectiveSlowdownThicknessCmd](#)
Set the effective slowdown thickness.
- G4UlcmdWithADoubleAndUnit * [SlowdownLengthCmd](#)
Set the slowdown length.
- G4UlcmdWithADoubleAndUnit * [SlowdownThicknessCmd](#)
Set the slowdown thickness.
- G4UlcmdWithADoubleAndUnit * [BeamBlockerThicknessCmd](#)
Sets the beam blocker thickness.
- G4UlcmdWithADoubleAndUnit * [ShieldingBeamBlockerThicknessCmd](#)
Sets the beam blocker thickness.
- G4UlcmdWithADoubleAndUnit * [ShieldingThicknessCmd](#)
Sets the shielding thickness.
- G4UlcmdWithADoubleAndUnit * [TopShieldingThicknessCmd](#)
Sets the shielding thickness.
- G4UlcmdWithADoubleAndUnit * **GeDetectorRadiusCmd**
- G4UlcmdWithADoubleAndUnit * **GeDetectorShellLengthCmd**
- G4UlcmdWithADoubleAndUnit * **GeDetectorLengthCmd**
- G4UlcmdWithADoubleAndUnit * **GeDetectorShellInnerRadiusCmd**
- G4UlcmdWithADoubleAndUnit * **GeDetectorShellOuterRadiusCmd**
- G4UlcmdWithAString * [ShieldingMaterialCmd](#)
A command which can be invoked.
- G4UlcmdWithAString * **GeDetectorShellMaterialCmd**
- G4UlcmdWithAString * **GermaniumDetectorMaterialCmd**
- G4UlcmdWithAString * **BeamBlockerMaterialCmd**
- G4UlcmdWithAString * **ShieldingBeamBlockerMaterialCmd**
- G4UlcmdWithAString * **SlowdownMaterialCmd**
- G4UlcmdWithABool * **FlipBetaCmd**

5.6.1 Detailed Description

Creates commands that the user can invoke via the command line.

Definition at line 36 of file PrimaryGeneratorMessenger.hh.

The documentation for this class was generated from the following file:

- [DetectorMessenger.hh](#)

5.7 EventAction Class Reference

Holds information about what to do at the beginning and end of each event.

```
#include <EventAction.hh>
```

Public Member Functions

- [EventAction](#) ([RunAction](#) *run, [DetectorConstruction](#) *det, G4double accumulationtime=1000 *ns)
Constructor, constructs the [EventAction](#) object.
- virtual [~EventAction](#) ()
Destructor, destroys the [EventAction](#) object.
- void [BeginOfEventAction](#) (const G4Event *)
Actions that are taken at the beginning of each event.
- void [EndOfEventAction](#) (const G4Event *)
Actions that are taken at the end of each event.
- void [AddScintillatorTubeEvent](#) (G4int scintillatorID, G4double time, G4double energy)
Add a scintillator event when such occurs.
- void [AddGeDetectorEvent](#) (G4double time, G4double energy)
Add a germanium detector event when such occurs.
- void [AddScintillatorPlateEvent](#) (AllAccumulatedStatistics::ScintillatorPlate plate, G4double time, G4double energy)
Add a scintillator plate event.
- void [PrintToVerboseFile](#) (G4String messageToPrint)
Set the PrintModulo to a specific number.
- void [SetPrintModulo](#) (G4int val)

Private Attributes

- [RunAction](#) * runAct
The Pointer to the [RunAction](#) associated with this [EventAction](#) object.
- G4int printModulo
Print a message when the event number modulo this number equals zero.

- [EventActionMessenger](#) * [eventMessenger](#)
The messenger.
- [DetectorConstruction](#) * [detector](#)
The detector construction.
- G4double [myAccumulationTime](#)
Default time to accumulate statistics over, used when constructing the myStatistics object.
- [AllAccumulatedStatistics](#) * [myStatistics](#)
Contains the statistics accumulated during the event.

5.7.1 Detailed Description

Holds information about what to do at the beginning and end of each event.

Accumulates statistics for the ongoing run.

Definition at line 51 of file EventAction.hh.

5.7.2 Constructor & Destructor Documentation

5.7.2.1 EventAction::EventAction ([RunAction](#) * [run](#), [DetectorConstruction](#) * [det](#), G4double [accumulationtime](#) = 1000*ns)

Constructor, constructs the [EventAction](#) object.

Parameters

run	The RunAction to associate with this EventAction object.
det	The DetectorConstruction which we are using.
accumulationtime	The accumulation time, default value 20 us.

Definition at line 3 of file EventAction.cc.

References [detector](#), and [eventMessenger](#).

```

:runAct(run),printModulo(100),eventMessenger(0), myAccumulationTime(accumulationtime)
{
    detector=det;
    eventMessenger = new EventActionMessenger(this);
}

```

5.7.3 Member Function Documentation

5.7.3.1 void EventAction::AddGeDetectorEvent (G4double *time*, G4double *energy*)

Add a germanium detector event when such occurs.

Parameters

<i>time</i>	The time of the event.
<i>energy</i>	The energy of deposited.

Definition at line 50 of file EventAction.cc.

References AllAccumulatedStatistics::AddGeDetectorEvent(), and myStatistics.

Referenced by SteppingAction::UserSteppingAction().

```
{
    myStatistics->AddGeDetectorEvent (time, energy) ;
}
```

5.7.3.2 void EventAction::AddScintillatorPlateEvent (AllAccumulatedStatistics::ScintillatorPlate *plate*, G4double *time*, G4double *energy*)

Add a scintillator plate event.

Parameters

<i>plate</i>	The plate the event occurred in
<i>time</i>	The time of the event.
<i>energy</i>	The energy of deposited.

Definition at line 55 of file EventAction.cc.

References AllAccumulatedStatistics::AddScintillatorPlateEvent(), and myStatistics.

Referenced by SteppingAction::UserSteppingAction().

```
{
    myStatistics->AddScintillatorPlateEvent (plate, time, energy) ;
}
```

5.7.3.3 void EventAction::AddScintillatorTubeEvent (G4int *scintillatorID*, G4double *time*, G4double *energy*)

Add a scintillator event when such occurs.

Parameters

<i>scintillatorID</i>	The scintillator ID
<i>time</i>	The time of the event.
<i>energy</i>	Deposited energy.

Definition at line 45 of file EventAction.cc.

References AllAccumulatedStatistics::AddScintillatorTubeEvent(), and myStatistics.

Referenced by `SteppingAction::UserSteppingAction()`.

```
{
    myStatistics->AddScintillatorTubeEvent(scintillatorID,time,energy); //a range c
    heck would probable be good here.
}
```

5.7.3.4 void EventAction::SetPrintModulo (G4int val) [inline]

Parameters

<i>val</i>	The value to set PrintModulo to.
------------	----------------------------------

Definition at line 82 of file EventAction.hh.

References `printModulo`.

Referenced by `EventActionMessenger::SetNewValue()`.

```
{printModulo = val;};
```

The documentation for this class was generated from the following files:

- [EventAction.hh](#)
- [EventAction.cc](#)

5.8 EventActionMessenger Class Reference

Creates commands that can be used to control the EventActions via the command line.

```
#include <EventActionMessenger.hh>
```

Public Member Functions

- [EventActionMessenger](#) ([EventAction](#) *)
Constructor, constructs the object.
- virtual [~EventActionMessenger](#) ()
Destructor, destroys the object.
- void [SetNewValue](#) (G4UCommand *, G4String)
This method is invoked when a command that alters [EventAction](#) is typed in via the command line.

Private Attributes

- [EventAction](#) * `eventAction`
The [EventAction](#) associated with this [EventActionMessenger](#).

- G4UIdirectory * [eventDir](#)

The directory in which the command will be contained.

- G4UIcmdWithAnInteger * [PrintCmd](#)

The command itself, in this case, print events modulo a number.

5.8.1 Detailed Description

Creates commands that can be used to control the EventActions via the command line.

Definition at line 33 of file EventActionMessenger.hh.

5.8.2 Constructor & Destructor Documentation

5.8.2.1 EventActionMessenger::EventActionMessenger (EventAction * EvAct)

Constructor, constructs the object.

Parameters

<i>EvAct</i>	The EventAction to associate with this EventActionMessenger .
--------------	---

Definition at line 3 of file EventActionMessenger.cc.

References [eventDir](#), and [PrintCmd](#).

```

:~EventActionMessenger()
{
    eventDir = new G4UIdirectory("/PIBIDS/event/");
    eventDir->SetGuidance("event control");

    PrintCmd = new G4UIcmdWithAnInteger("/PIBIDS/event/printModulo",this);
    PrintCmd->SetGuidance("Print events modulo n");
    PrintCmd->SetParameterName("EventNb",false);
    PrintCmd->SetRange("EventNb>0");
}

```

5.8.3 Member Function Documentation

5.8.3.1 void EventActionMessenger::SetNewValue (G4UIcommand * command, G4String newValue)

This method is invoked when a command that alters [EventAction](#) is typed in via the command line.

Parameters

<i>command</i>	The G4UIcommand which we want to change value in.
<i>newValue</i>	The string containing the new value.

Definition at line 22 of file EventActionMessenger.cc.

References eventAction, PrintCmd, and EventAction::SetPrintModulo().

```
{
    if(command == PrintCmd)
        {eventAction->SetPrintModulo(PrintCmd->GetNewIntValue(newValue));}
}
```

The documentation for this class was generated from the following files:

- [EventActionMessenger.hh](#)
- [EventActionMessenger.cc](#)

5.9 EventHit Class Reference

Holds accumulated statistics for a single event hit.

```
#include <EventHit.hh>
```

Public Member Functions

- [EventHit](#) (G4int numberOfScintillatorTubes, G4double defaultTime=0)
Constructor, constructs the object.

Public Attributes

- G4double [time](#)
The time of the event start.
- G4double [upperFrontScintillatorPanelEnergy](#)
Energy in the upper front scintillator panel.
- G4double [lowerFrontScintillatorPanelEnergy](#)
Energy in the lower front scintillator panel.
- G4double [backScintillatorPanelEnergy](#)
Energy in the back scintillator panel.
- G4double [geDetectorEnergy](#)
Energy in the Ge detector.
- std::vector< G4double > [scintillatorTubeEnergies](#)
Energies in the scintillator tubes.

5.9.1 Detailed Description

Holds accumulated statistics for a single event hit.

Definition at line 24 of file EventHit.hh.

5.9.2 Constructor & Destructor Documentation

5.9.2.1 EventHit::EventHit (G4int *numberOfScintillatorTubes*, G4double *defaultTime* = 0)

Constructor, constructs the object.

Parameters

<i>numberOfScintillatorTubes</i>	Number of scintillator tubes. This is used to preinitialize the scintillator tube vector with the correct number of scintillator tubes.
<i>defaultTime</i>	The default time to set.

Definition at line 3 of file EventHit.cc.

References backScintillatorPanelEnergy, geDetectorEnergy, lowerFrontScintillatorPanelEnergy, scintillatorTubeEnergies, time, and upperFrontScintillatorPanelEnergy.

```
{
    scintillatorTubeEnergies = std::vector<G4double>(numberOfScintillatorTubes,0);
    time=defaultTime;
    upperFrontScintillatorPanelEnergy = 0;
    lowerFrontScintillatorPanelEnergy = 0;
    backScintillatorPanelEnergy = 0;
    geDetectorEnergy = 0;
}
```

The documentation for this class was generated from the following files:

- [EventHit.hh](#)
- [EventHit.cc](#)

5.10 PrimaryGeneratorMessenger::EventHit Class Reference

Holds accumulated statistics for a single event hit.

```
#include <PrimaryGeneratorMessenger.hh>
```

Public Member Functions

- [EventHit](#) (G4int numberOfScintillatorTubes, G4double defaultTime=0)
Constructor, constructs the object.
- [EventHit](#) (G4int numberOfScintillatorTubes, G4double defaultTime=0)
Constructor, constructs the object.

Public Attributes

- G4double [time](#)

The time of the event start.

- G4double [upperFrontScintillatorPanelEnergy](#)
Energy in the upper front scintillator panel.
- G4double [lowerFrontScintillatorPanelEnergy](#)
Energy in the lower front scintillator panel.
- G4double [backScintillatorPanelEnergy](#)
Energy in the back scintillator panel.
- G4double [geDetectorEnergy](#)
Energy in the Ge detector.
- std::vector< G4double > [scintillatorTubeEnergies](#)
Energies in the scintillator tubes.

5.10.1 Detailed Description

Holds accumulated statistics for a single event hit.

Definition at line 25 of file PrimaryGeneratorMessenger.hh.

5.10.2 Constructor & Destructor Documentation

5.10.2.1 PrimaryGeneratorMessenger::EventHit::EventHit (G4int *numberOfScintillatorTubes*, G4double *defaultTime* = 0)

Constructor, constructs the object.

Parameters

<i>numberOfScintillatorTubes</i>	Number of scintillator tubes. This is used to preinitialize the scintillator tube vector with the correct number of scintillator tubes.
<i>defaultTime</i>	The default time to set.

5.10.2.2 PrimaryGeneratorMessenger::EventHit::EventHit (G4int *numberOfScintillatorTubes*, G4double *defaultTime* = 0)

Constructor, constructs the object.

Parameters

<i>numberOfScintillatorTubes</i>	Number of scintillator tubes. This is used to preinitialize the scintillator tube vector with the correct number of scintillator tubes.
<i>defaultTime</i>	The default time to set.

The documentation for this class was generated from the following file:

- [EventHit.hh](#)

5.11 G4EmStandardFysik Class Reference

This class is extending the G4EmStandardPhysics class in order to apply a few changes to this physics list.

```
#include <G4EmStandardFysik.hh>
```

Public Member Functions

- [G4EmStandardFysik](#) (G4int ver)
Constructor, constructs the object.
- virtual void [ConstructProcess](#) ()
Destructor, destroys the object.

Private Attributes

- G4int [verbose](#)
Used to indicate verbose behaviour.

5.11.1 Detailed Description

This class is extending the G4EmStandardPhysics class in order to apply a few changes to this physics list.

Definition at line 16 of file G4EmStandardFysik.hh.

5.11.2 Constructor & Destructor Documentation

5.11.2.1 G4EmStandardFysik::G4EmStandardFysik (G4int ver)

Constructor, constructs the object.

Parameters

<i>ver</i>	Tells the verbosity level.
------------	----------------------------

Definition at line 50 of file G4EmStandardFysik.cc.

References [verbose](#).

```
{
    verbose = ver;
}
```

The documentation for this class was generated from the following files:

- [G4EmStandardFysik.hh](#)
- [G4EmStandardFysik.cc](#)

5.12 MyPhysicsList< T > Class Template Reference

Customized physics list class.

```
#include <MyPhysicsList.hh>
```

Public Member Functions

- [MyPhysicsList](#) (G4int ver=0)
Constructor, constructs the physics list.
- virtual [~MyPhysicsList](#) ()
Destructor, destroys the object.
- virtual void [SetCuts](#) ()
Sets the cuts.

Private Types

- enum { **ok** = CompileTimeConstraints::IsA<T, **ok** = CompileTimeConstraints::IsA<T
}

5.12.1 Detailed Description

```
template<class T>class MyPhysicsList< T >
```

Customized physics list class.

Definition at line 26 of file MyPhysicsList.hh.

5.12.2 Constructor & Destructor Documentation

5.12.2.1 `template<class T > MyPhysicsList< T >::MyPhysicsList (G4int ver = 0)`

Constructor, constructs the physics list.

Parameters

<code>ver</code>	Sets the verbosity level. Default is 0.
------------------	---

Definition at line 28 of file MyPhysicsList.icc.

5.13 PrimaryGeneratorMessenger::PrimaryGeneratorAction Class Reference 65

```

: T()

{

    G4DataQuestionnaire it(photon);
    G4cout << "APAAPAAPA"<<G4endl;
    G4cout <<G4endl;

    this->defaultCutValue = 0.7*mm;
    this->SetVerboseLevel(ver);

    // EM Physics
    this->RegisterPhysics( new G4EmStandardPhysics(ver));

    // Synchrotron Radiation & GN Physics
    this->RegisterPhysics( new G4EmExtraPhysics("extra EM"));

    // Decays
    this->RegisterPhysics( new G4DecayPhysics("decay",ver) );

    // Hadron Elastic scattering
    this-> RegisterPhysics( new G4HadronElasticPhysics("elastic",ver,false));

    // Hadron Physics
    G4bool quasiElastic;
    this->RegisterPhysics( new HadronPhysicsQGSP_BERT("hadron",quasiElastic=true));

    // Stopping Physics
    this->RegisterPhysics( new G4QStoppingPhysics("stopping"));

    // Ion Physics
    this->RegisterPhysics( new G4IonPhysics("ion"));

    // Neutron tracking cut
    this->RegisterPhysics( new G4NeutronTrackingCut("Neutron tracking cut", ver));

    // Radioactive decay physics
    this->RegisterPhysics(new G4RadioactiveDecayPhysics(5));

}

```

The documentation for this class was generated from the following files:

- [MyPhysicsList.hh](#)
- [MyPhysicsList.icc](#)

5.13 PrimaryGeneratorMessenger::PrimaryGeneratorAction Class Reference

Mandatory user class providing the primary particle generator.

```
#include <PrimaryGeneratorMessenger.hh>
```

Public Member Functions

- [PrimaryGeneratorAction](#) ([DetectorConstruction](#) *, [RunAction](#) *)

Constructor. Sets the particle gun position, energy and direction.

- void [SetParticle](#) (G4int Z=6, G4int A=17, G4double E=0., G4double Q=0.)
Obtains pointers to particle and ion tables, and retrieves the particle we want from them.
- virtual [~PrimaryGeneratorAction](#) ()
Destructor, destroys the object.
- void [GeneratePrimaries](#) (G4Event *)
Randoms the particle gun's position (if desired) and generates an event.
- void [SetRndmFlag](#) (G4String val)
If rndmFlag is true, the gun's position is randomized at the beginning of each event.
- G4ParticleGun * [GetParticleGun](#) ()
Returns the ParticleGun.
- void [PrintTargetSize](#) ()
Print target size.

Private Attributes

- G4ParticleGun * [particleGun](#)
Pointer a to the G4ParticleGun class.
- [DetectorConstruction](#) * [Detector](#)
Pointer to DetectorConstruction.
- [RunAction](#) * [myRunAction](#)
- [PrimaryGeneratorMessenger](#) * [gunMessenger](#)
Messenger for this class.
- G4String [rndmFlag](#)
Flag for random gun position.

5.13.1 Detailed Description

Mandatory user class providing the primary particle generator.

Definition at line 42 of file PrimaryGeneratorMessenger.hh.

5.13.2 Member Function Documentation

- 5.13.2.1 void [PrimaryGeneratorMessenger::PrimaryGeneratorAction::SetParticle](#) (G4int Z = 6, G4int A = 17, G4double E = 0. , G4double Q = 0.)

Obtains pointers to particle and ion tables, and retrieves the particle we want from them.

Parameters

Z	The atomic number of the particle.
A	The atomic mass of the particle.
E	The excitation energy of the particle.
Q	The charge of the particle.

Referenced by PrimaryGeneratorMessenger::SetNewValue().

The documentation for this class was generated from the following file:

- [PrimaryGeneratorAction.hh](#)

5.14 PrimaryGeneratorAction Class Reference

Mandatory user class providing the primary particle generator.

```
#include <PrimaryGeneratorAction.hh>
```

Public Member Functions

- [PrimaryGeneratorAction](#) ([DetectorConstruction](#) *, [RunAction](#) *)
Constructor. Sets the particle gun position, energy and direction.
- void [SetParticle](#) (G4int Z=6, G4int A=17, G4double E=0., G4double Q=0.)
Obtains pointers to particle and ion tables, and retrieves the particle we want from them.
- virtual [~PrimaryGeneratorAction](#) ()
Destructor, destroys the object.
- void [GeneratePrimaries](#) (G4Event *)
Randoms the particle gun's position (if desired) and generates an event.
- void [SetRndmFlag](#) (G4String val)
If rndmFlag is true, the gun's position is randomized at the beginning of each event.
- G4ParticleGun * [GetParticleGun](#) ()
Returns the ParticleGun.
- void [PrintTargetSize](#) ()
Print target size.

Private Attributes

- G4ParticleGun * [particleGun](#)
Pointer a to the G4ParticleGun class.
- [DetectorConstruction](#) * [Detector](#)
Pointer to [DetectorConstruction](#).
- [RunAction](#) * **myRunAction**
- [PrimaryGeneratorMessenger](#) * [gunMessenger](#)
Messenger for this class.
- G4String [rndmFlag](#)
Flag for random gun position.

5.14.1 Detailed Description

Mandatory user class providing the primary particle generator.

Definition at line 42 of file PrimaryGeneratorAction.hh.

5.14.2 Constructor & Destructor Documentation

5.14.2.1 PrimaryGeneratorAction::PrimaryGeneratorAction (DetectorConstruction * *DC*, RunAction * *run*)

Constructor. Sets the particle gun position, energy and direction.

Parameters

<i>DC</i>	Object containing the information the constructor needs.
<i>run</i>	Pointer to the RunAction , to inform this when we change the particle in the gun.

Definition at line 27 of file PrimaryGeneratorAction.cc.

References [Detector](#), [DetectorConstruction::GetWorldSizeX\(\)](#), [gunMessenger](#), and [particleGun](#).

```

:Detector (DC) , myRunAction (run) , rndmFlag ("on")
{
    G4int n_particle = 1;
    particleGun = new G4ParticleGun(n_particle);

    //create a messenger for this class
    gunMessenger = new PrimaryGeneratorMessenger(this);

    // default particle kinematic
    particleGun->SetParticleMomentumDirection(G4ThreeVector(1.,0.,0.));
    particleGun->SetParticleEnergy(11*GeV);           // old value: 2.2
                                                    GeV

    //default particle gun position.
    G4double position = -(Detector->GetWorldSizeX());
    particleGun->SetParticlePosition(G4ThreeVector(position,0.*cm,0.*cm));
}

```

5.14.3 Member Function Documentation

5.14.3.1 void PrimaryGeneratorAction::SetParticle (G4int *Z* = 6, G4int *A* = 17, G4double *E* = 0., G4double *Q* = 0.)

Obtains pointers to particle and ion tables, and retrieves the particle we want from them.

Parameters

<i>Z</i>	The atomic number of the particle.
<i>A</i>	The atomic mass of the particle.

E	The excitation energy of the particle.
Q	The charge of the particle.

Definition at line 11 of file PrimaryGeneratorAction.cc.

References `particleGun`.

Referenced by `main()`.

```
{
  G4ParticleTable* particleTable = G4ParticleTable::GetParticleTable();
  particleTable->GetIonTable()->CreateAllIon();

  G4ParticleDefinition* particle = particleTable->GetIon(Z, A, E);
  if(particle!=NULL)
  {
    particleGun->SetParticleCharge(Q);
    particleGun->SetParticleDefinition(particle);
    myRunAction->SetParticleZ(Z);
    myRunAction->SetParticleA(A);
  }
}
```

The documentation for this class was generated from the following files:

- [PrimaryGeneratorAction.hh](#)
- [PrimaryGeneratorAction.cc](#)

5.15 PrimaryGeneratorMessenger Class Reference

Creates commands that allows the user to control the PrimaryGenerator via the command line.

```
#include <PrimaryGeneratorMessenger.hh>
```

Classes

- class [AllAccumulatedStatistics](#)
Small class to keep track of the accumulated statistics.
- class [DetectorConstruction](#)
This class constructs the detector, according to specifications.
- class [DetectorMessenger](#)
Creates commands that the user can invoke via the command line.
- class [EventHit](#)
Holds accumulated statistics for a single event hit.
- class [PrimaryGeneratorAction](#)
Mandatory user class providing the primary particle generator.
- class [RunActionMessenger](#)
Creates commands that can be used to control the [RunAction](#) via the command line.

Public Member Functions

- void **DetectorConstruction::ComputeDependentDetectorParameters** ()
- [PrimaryGeneratorMessenger](#) ([PrimaryGeneratorAction](#) *)
Constructor, constructs the object.
- virtual [~PrimaryGeneratorMessenger](#) ()
Destructor, destroys the object.
- void [SetNewValue](#) (G4UCommand *, G4String)
This method is invoked when a command to the primarygenerator is typed in via the command line.

Private Attributes

- [PrimaryGeneratorAction](#) * [Action](#)
The [PrimaryGeneratorAction](#) associated with this [PrimaryGeneratorMessenger](#).
- G4Udirectory * [gunDir](#)
The directory used for the gun settings.
- G4UlcmdWithAString * [RndmCmd](#)
Lets us choose if the incident particle should be shot randomly or not.
- G4UlcmdWithAnInteger * [gunZ](#)
Command for setting Z, the atomic number of the particle.
- G4UlcmdWithAnInteger * [gunA](#)
Command for setting A, the atomic mass of the particle.
- G4UlcmdWithADouble * [gunE](#)
Command for setting E, the excitation energy of the particle.
- G4UlcmdWithADouble * [gunQ](#)
Command for setting Q, the charge of the particle.
- G4UlcmdWithoutParameter * [targetArea](#)
Show target area.
- G4int [currentGunZ](#)
Current value of gun Z.
- G4int [currentGunA](#)
Current value of gun A.
- G4int [currentGunE](#)
Current value of gun E.
- G4int [currentGunQ](#)
Current value of gun Q.

5.15.1 Detailed Description

Creates commands that allows the user to control the PrimaryGenerator via the command line.

Definition at line 35 of file PrimaryGeneratorMessenger.hh.

5.15.2 Constructor & Destructor Documentation

5.15.2.1 PrimaryGeneratorMessenger::PrimaryGeneratorMessenger (PrimaryGeneratorAction * Gun)

Constructor, constructs the object.

Parameters

<i>Gun</i>	The PrimaryGeneratorAction to be associated with this PrimaryGeneratorMessenger .
------------	---

Definition at line 3 of file PrimaryGeneratorMessenger.cc.

References [currentGunA](#), [currentGunE](#), [currentGunQ](#), [currentGunZ](#), [gunA](#), [gunDir](#), [gunE](#), [gunQ](#), [gunZ](#), [RndmCmd](#), and [targetArea](#).

```
:Action(Gun)
{
    gunDir = new G4UIdirectory("/PIBIDS/gun/");
    gunDir->SetGuidance("PrimaryGenerator control");

    RndmCmd = new G4UIcmdWithAString("/PIBIDS/gun/rndm",this);
    RndmCmd->SetGuidance("Shoot randomly the incident particle.");
    RndmCmd->SetGuidance(" Choice : on(default), off");
    RndmCmd->SetParameterName("choice",true);
    RndmCmd->SetDefaultValue("on");
    RndmCmd->SetCandidates("on off");
    RndmCmd->AvailableForStates(G4State_PreInit,G4State_Idle);

    gunZ = new G4UIcmdWithAnInteger("/PIBIDS/gun/Z",this);
    gunZ->SetGuidance("Set the atomic number of the incident particle");
    gunZ->SetParameterName("Z",true);
    gunZ->SetDefaultValue(6);
    currentGunZ = 6;
    gunZ->SetRange("Z>=0");

    gunA = new G4UIcmdWithAnInteger("/PIBIDS/gun/A",this);
    gunA->SetGuidance("Set the atomic number of the incident particle");
    gunA->SetParameterName("A",true);
    gunA->SetDefaultValue(17);
    currentGunA = 17;
    gunA->SetRange("A>=0");

    gunE = new G4UIcmdWithADouble("/PIBIDS/gun/E",this);
    gunE->SetGuidance("Set the excitation energy of the incident particle");
    gunE->SetParameterName("E",true);
    gunE->SetDefaultValue(0.);
    currentGunE = 0.;

    gunQ = new G4UIcmdWithADouble("/PIBIDS/gun/Q",this);
    gunQ->SetGuidance("Set the charge of the incident particle");
    gunQ->SetParameterName("E",true);
    gunQ->SetDefaultValue(0.);
    currentGunQ = 0.;

    targetArea = new G4UIcmdWithoutParameter("/PIBIDS/gun/area",this);
    targetArea->SetGuidance("Show target area");
}
```

The documentation for this class was generated from the following files:

- [PrimaryGeneratorMessenger.hh](#)
- [DetectorConstruction.hh](#)
- [PrimaryGeneratorMessenger.cc](#)

5.16 RunAction Class Reference

Here we can put stuff to be executed before and after each run.

```
#include <RunAction.hh>
```

Public Member Functions

- [RunAction](#) ()
Constructor, constructs the object.
- virtual [~RunAction](#) ()
Destructor, destroys the object.
- void [BeginOfRunAction](#) (const G4Run *)
Is invoked at the beginning of each run by the RunManager.
- void [EndOfRunAction](#) (const G4Run *)
Is invoked at the end of each run by the RunManager.
- void [SetNewResultsFileName](#) (G4String filename)
Sets the filename of the file to write the results to.
- void [WriteResultsToFile](#) (G4int eventID, [AllAccumulatedStatistics](#) *myEventStatistics)

Invoked after each event by the [EventAction](#) class in order to write the event data from that event to the file.
- void [SetNewVerboseResultsFileName](#) (G4String file)
Sets if we are using a verbose file or not.
- void [SetVerboseFileUsage](#) (G4bool usage)
- void [PrintToVerboseFile](#) (G4String messageToPrint)
- void [SetParticleZ](#) (G4int newZ)
- void [SetParticleA](#) (G4int newA)

Private Attributes

- G4String [resultsFileName](#)
File name of the results file.
- FILE * [resultsFile](#)
Results file pointer.
- G4String [verboseResultsFileName](#)
The name of the verbose results file we might be using.

- G4bool [usingVerboseFile](#)
True if we are using a verbose file, otherwise false.
- FILE * [verboseFile](#)
The file to dump some debug output to, used if we are using a verbose file.
- G4int **particleZ**
- G4int **particleA**
- [RunActionMessenger](#) * **myMessenger**

5.16.1 Detailed Description

Here we can put stuff to be executed before and after each run.

Definition at line 38 of file RunAction.hh.

5.16.2 Member Function Documentation

5.16.2.1 void RunAction::SetNewResultsFileName (G4String filename)

Sets the filename of the file to write the results to.

Parameters

<i>filename</i>	The filename of the file to write the run results to.
-----------------	---

Definition at line 15 of file RunAction.cc.

References resultsFileName.

Referenced by RunActionMessenger::RunActionMessenger(), and RunActionMessenger::SetNewValue().

```
{
    resultsFileName = filename;
}
```

5.16.2.2 void RunAction::SetNewVerboseResultsFileName (G4String file) [inline]

Sets if we are using a verbose file or not.

Parameters

<i>file</i>	The filename to set.
-------------	----------------------

Definition at line 55 of file RunAction.hh.

References verboseResultsFileName.

Referenced by RunActionMessenger::RunActionMessenger(), and RunActionMessenger::SetNewValue().

5.16.2.3 void RunAction::SetVerboseFileUsage (G4bool *usage*) [inline]

Parameters

<i>usage</i>	Tells if we are using a verbose file.
--------------	---------------------------------------

Definition at line 59 of file RunAction.hh.

References usingVerboseFile.

Referenced by RunActionMessenger::RunActionMessenger(), and RunActionMessenger::SetNewValue().

```
{usingVerboseFile = usage;}
```

5.16.2.4 void RunAction::WriteResultsToFile (G4int *eventID*, AllAccumulatedStatistics * *myEventStatistics*)

Invoked after each event by the [EventAction](#) class in order to write the event data from that event to the file.

Sets the new VerboseResultsFileName

Parameters

<i>eventID</i>	The event ID.
<i>myEventStatistics</i>	Write the event data to the file.

Definition at line 30 of file RunAction.cc.

References AllAccumulatedStatistics::GetEventHits(), and resultsFile.

Referenced by EventAction::EndOfEventAction().

```
{
    std::vector<EventHit> eventsToWrite = myEventStatistics->GetEventHits();
    for(std::vector<EventHit>::iterator it = eventsToWrite.begin(); it!=eventsToWrite.end(); it++)
    {
        fprintf(resultsFile, "%d %d %d ", eventID, particleZ, particleA);

        fprintf(resultsFile, "%.15e ",it->time/ns);
        if(it->upperFrontScintillatorPanelEnergy/MeV>1E-100)
            fprintf(resultsFile, "%.15e ",it->upperFrontScintillatorPanelEnergy/MeV);
        else
            fprintf(resultsFile, "%d ",0);

        if(it->lowerFrontScintillatorPanelEnergy/MeV>1E-100)
            fprintf(resultsFile, "%.15e ",it->lowerFrontScintillatorPanelEnergy/MeV);
        else
            fprintf(resultsFile, "%d ",0);
    }
}
```

```

    if(it->backScintillatorPanelEnergy/MeV>1E-100)
        fprintf(resultsFile, "%.15e ",it->backScintillatorPanelEnergy/MeV);
    else
        fprintf(resultsFile, "%d ",0);

    if(it->geDetectorEnergy/MeV>1E-100)
        fprintf(resultsFile, "%.15e",it->geDetectorEnergy/MeV);
    else
        fprintf(resultsFile, "%d",0);

    for(std::vector<G4double>::iterator ig = it->scintillatorTubeEnergies.begin
        (); ig!=it->scintillatorTubeEnergies.end(); ig++)
    {
        if(*ig/MeV>1E-100)
            fprintf(resultsFile, " %.15e",*ig/MeV);
        else
            fprintf(resultsFile, " %d",0);
    }
    fprintf(resultsFile, "\n");
}

```

The documentation for this class was generated from the following files:

- [RunAction.hh](#)
- [RunAction.cc](#)

5.17 RunActionMessenger Class Reference

Creates commands that can be used to control the [RunAction](#) via the command line.

```
#include <RunActionMessenger.hh>
```

Public Member Functions

- [RunActionMessenger](#) ([RunAction](#) *)
Constructor, constructs the object.
- virtual [~RunActionMessenger](#) ()
Destructor, destroys the object.
- void [SetNewValue](#) (G4UIcommand *, G4String)
This method is invoked when a command that alters [RunAction](#) is typed in via the command line.

Private Attributes

- [RunAction](#) * [runAction](#)
The [RunAction](#) associated with this [RunActionMessenger](#).
- G4UIdirectory * [eventDir](#)
The directory in which the command will be contained.

- G4UlcmdWithAString * [SimulationResultsFileCmd](#)

The command itself.

- G4UlcmdWithAString * **VerboseResultsFile**
- G4UlcmdWithABool * **UseVerboseFile**

5.17.1 Detailed Description

Creates commands that can be used to control the [RunAction](#) via the command line.

Definition at line 33 of file RunActionMessenger.hh.

5.17.2 Constructor & Destructor Documentation

5.17.2.1 RunActionMessenger::RunActionMessenger (RunAction * RunAct)

Constructor, constructs the object.

Parameters

<i>RunAct</i>	The RunAction to associate with this RunActionMessenger .
---------------	---

Definition at line 3 of file RunActionMessenger.cc.

References [eventDir](#), [RunAction::SetNewResultsFileName\(\)](#), [RunAction::SetNewVerboseResultsFileName\(\)](#), [RunAction::SetVerboseFileUsage\(\)](#), and [SimulationResultsFileCmd](#).

```
:runAction(RunAct)
{
    eventDir = new G4UIdirectory("/PIBIDS/files");
    eventDir->SetGuidance("File control");

    SimulationResultsFileCmd = new G4UlcmdWithAString("/PIBIDS/files/resultsFile",t
        his);
    SimulationResultsFileCmd->SetGuidance("File to dump the results of the current
        run to.");
    SimulationResultsFileCmd->SetParameterName("FileName", false);
    SimulationResultsFileCmd->SetDefaultValue("SimulationResults.txt");
    RunAct->SetNewResultsFileName("SimulationResults.txt");

    VerboseResultsFile = new G4UlcmdWithAString("/PIBIDS/files/verboseResultsFile",
        this);
    VerboseResultsFile->SetGuidance("Print verbose results to this file.");
    VerboseResultsFile->SetParameterName("FileName", false);
    VerboseResultsFile->SetDefaultValue("VerboseSimulationResults.txt");
    RunAct->SetNewVerboseResultsFileName("VerboseSimulationResults.txt");

    UseVerboseFile = new G4UlcmdWithABool("/PIBIDS/files/useVerboseResultsFile",thi
        s);
    UseVerboseFile->SetGuidance("Use this to switch the usage of a verbose file on
        or off");
    UseVerboseFile->SetParameterName("UseFile", false);
    RunAct->SetVerboseFileUsage(false);
}
```

5.17.3 Member Function Documentation

5.17.3.1 void RunActionMessenger::SetNewValue (G4UIcommand * *command*, G4String *newValue*)

This method is invoked when a command that alters [RunAction](#) is typed in via the command line.

Parameters

<i>command</i>	The G4UIcommand which we want to change value in.
<i>newValue</i>	The string containing the new value.

Definition at line 35 of file RunActionMessenger.cc.

References [runAction](#), [RunAction::SetNewResultsFileName\(\)](#), [RunAction::SetNewVerboseResultsFileName\(\)](#), [RunAction::SetVerboseFileUsage\(\)](#), and [SimulationResultsFileCmd](#).

```
{
    if (command == SimulationResultsFileCmd)
        runAction->SetNewResultsFileName (newValue);
    if (command == VerboseResultsFile)
        {runAction->SetNewVerboseResultsFileName (newValue); }
    if (command == UseVerboseFile)
        {runAction->SetVerboseFileUsage (UseVerboseFile->GetNewBoolValue (newValue)); }
}
```

The documentation for this class was generated from the following files:

- [RunActionMessenger.hh](#)
- [RunActionMessenger.cc](#)

5.18 PrimaryGeneratorMessenger::RunActionMessenger Class Reference

Creates commands that can be used to control the [RunAction](#) via the command line.

```
#include <PrimaryGeneratorMessenger.hh>
```

Public Member Functions

- [RunActionMessenger](#) ([RunAction](#) *)
Constructor, constructs the object.
- virtual [~RunActionMessenger](#) ()
Destructor, destroys the object.
- void [SetNewValue](#) (G4UIcommand *, G4String)
This method is invoked when a command that alters [RunAction](#) is typed in via the command line.

Private Attributes

- [RunAction](#) * [runAction](#)
The [RunAction](#) associated with this [RunActionMessenger](#).
- G4Uldirectory * [eventDir](#)
The directory in which the command will be contained.
- G4UlcmdWithAString * [SimulationResultsFileCmd](#)
The command itself.
- G4UlcmdWithAString * **VerboseResultsFile**
- G4UlcmdWithABool * **UseVerboseFile**

5.18.1 Detailed Description

Creates commands that can be used to control the [RunAction](#) via the command line.

Definition at line 34 of file PrimaryGeneratorMessenger.hh.

The documentation for this class was generated from the following file:

- [RunActionMessenger.hh](#)

5.19 SteppingAction Class Reference

Contains information about what to do at each step in the simulation.

```
#include <SteppingAction.hh>
```

Public Member Functions

- [SteppingAction](#) ([DetectorConstruction](#) *, [EventAction](#) *)
Constructor, constructs the object.
- virtual [~SteppingAction](#) ()
Destructor, destroys the object.
- void [UserSteppingAction](#) (const G4Step *)
Performs all required actions at each step.

Private Attributes

- [DetectorConstruction](#) * [detector](#)
Contains information about the detector.
- [EventAction](#) * [eventaction](#)
Pointer to an associated [EventAction](#) object.

5.19.1 Detailed Description

Contains information about what to do at each step in the simulation.

At each step the class checks if the particle is in either one of the scintillatortubes, in one of the scintillator plates, or in the germanium detector. If it is, it prints (to the screen or to a file) information about how much energy was deposited by what particle and at what time.

Definition at line 46 of file SteppingAction.hh.

5.19.2 Constructor & Destructor Documentation

5.19.2.1 SteppingAction::SteppingAction (DetectorConstruction * *det*, EventAction * *evt*)

Constructor, constructs the object.

Parameters

<i>det</i>	Information about the detector.
<i>evt</i>	Pointer to an associated EventAction object.

Definition at line 4 of file SteppingAction.cc.

```
:detector(det), eventaction(evt)
{
}
```

5.19.3 Member Function Documentation

5.19.3.1 void SteppingAction::UserSteppingAction (const G4Step * *aStep*)

Performs all required actions at each step.

Parameters

<i>aStep</i>	Contains information about the step in question.
--------------	--

Definition at line 14 of file SteppingAction.cc.

References [EventAction::AddGeDetectorEvent\(\)](#), [EventAction::AddScintillatorPlateEvent\(\)](#), [EventAction::AddScintillatorTubeEvent\(\)](#), [detector](#), [eventaction](#), [DetectorConstruction::GetBackScintillatorPlate\(\)](#), [DetectorConstruction::GetGermaniumDetector\(\)](#), [DetectorConstruction::GetLowerBetaDetector\(\)](#), [DetectorConstruction::GetLowerFrontScintillatorPlate\(\)](#), [DetectorConstruction::GetNumberOfScintillatorsPerBox\(\)](#), [DetectorConstruction::GetScintillator\(\)](#), [DetectorConstruction::GetUpperFrontScintillatorPlate\(\)](#), and [EventAction::PrintToVerboseFile\(\)](#).

```
{
```

```

G4cout.precision(15); //increase precision of cout.

// get volume of the current step
G4VPhysicalVolume* volume
= aStep->GetPreStepPoint()->GetTouchableHandle()->GetVolume();

// collect energy and track length step by step
//G4double edep = aStep->GetTotalEnergyDeposit();
G4double edep = aStep->GetTotalEnergyDeposit();

// record the time
G4double time = aStep->GetTrack()->GetGlobalTime();

if (volume == detector->GetScintillator())
{
    //Determine the scintillator ID.
    G4int scintillatorID = aStep->GetPreStepPoint()->GetTouchable()->GetReplicaNumber(1);
    scintillatorID += (aStep->GetPreStepPoint()->GetTouchable()->GetReplicaNumber(2)) * (detector->GetNumberOfScintillatorsPerBox());
    if (aStep->GetPreStepPoint()->GetTouchable()->GetVolume(3) == detector->GetLowerBetaDetector())
        scintillatorID += (detector->GetNumberOfScintillatorBoxes()) * (detector->GetNumberOfScintillatorsPerBox());

    eventaction->AddScintillatorTubeEvent(scintillatorID,time, edep);
    std::stringstream ss;
    ss.precision(15);
    ss << time/ns << " " << scintillatorID << " " << edep/MeV << " " << aStep->GetTrack()->GetDefinition()->GetParticleName().c_str() << " " << aStep->GetTrack()->GetTotalEnergy()/MeV << std::endl;
    eventaction->PrintToVerboseFile(ss.str());
}

if (volume == detector->GetGermaniumDetector())
{
    eventaction->AddGeDetectorEvent(time, edep);

    std::stringstream ss;
    ss.precision(15);
    ss << time/ns << " GermaniumDetector " << edep/MeV << " " << aStep->GetTrack()->GetDefinition()->GetParticleName().c_str() << " " << aStep->GetTrack()->GetTotalEnergy()/MeV << std::endl;
    eventaction->PrintToVerboseFile(ss.str());
}

if (volume == detector->GetLowerFrontScintillatorPlate())
{
    eventaction->AddScintillatorPlateEvent(AllAccumulatedStatistics::LowerFront,time/ns, edep);
}

if (volume == detector->GetUpperFrontScintillatorPlate())
{
    eventaction->AddScintillatorPlateEvent(AllAccumulatedStatistics::UpperFront,time, edep);
}

```

```

if (volume == detector->GetBackScintillatorPlate())
{
    eventaction->AddScintillatorPlateEvent(AllAccumulatedStatistics::Back,time, e
        dep);
}

if (volume == detector->GetUpperFrontScintillatorPlate() || volume == detector-
    >GetBackScintillatorPlate() || volume == detector->
    GetLowerFrontScintillatorPlate())
{
    if (edep/eV > 1)
    {
        std::stringstream ss;
        ss.precision(15);
        ss << time/ns << " " << volume->GetName().c_str() << " " << edep/MeV << "
            " << aStep->GetTrack()->GetDefinition()->GetParticleName().c_str() << " " << aSt
            ep->GetTrack()->GetTotalEnergy()/MeV << std::endl;
        eventaction->PrintToVerboseFile(ss.str());
    }
}

//if (aStep->GetTrack()->GetDefinition()->GetParticleName().compare("gamma")==0
    )
//    fprintf(file, "GAMMA: Time: %.15f, Energy: %f, Region: %s \n", time, aSte
        p->GetTrack()->GetTotalEnergy()/MeV, volume->GetName().c_str());

// G4cout << "Time: " << aStep->GetTrack()->GetGlobalTime()/s << G4endl;
}

```

The documentation for this class was generated from the following files:

- [SteppingAction.hh](#)
- [SteppingAction.cc](#)

5.20 SteppingVerbose Class Reference

Inclusion guard.

```
#include <SteppingVerbose.hh>
```

Public Member Functions

- [~SteppingVerbose](#) ()
Constructor, constructs the object.
- void [StepInfo](#) ()
Prints information with G4Cout when this is called.
- void [TrackingStarted](#) ()
Prints information with G4Cout when this is called.

5.20.1 Detailed Description

Inclusion guard.

Prints information at each step depending on the current verbosity level.

Definition at line 21 of file SteppingVerbose.hh.

5.20.2 Constructor & Destructor Documentation

5.20.2.1 SteppingVerbose::~SteppingVerbose ()

Constructor, constructs the object.

Destructor, destroys the object.

Definition at line 10 of file SteppingVerbose.cc.

```
{ }
```

The documentation for this class was generated from the following files:

- [SteppingVerbose.hh](#)
- [SteppingVerbose.cc](#)

Chapter 6

File Documentation

6.1 AllAccumulatedStatistics.cc File Reference

Source file for the [AllAccumulatedStatistics](#) class.

```
#include "AllAccumulatedStatistics.hh"
```

6.1.1 Detailed Description

Source file for the [AllAccumulatedStatistics](#) class.

Author

Rikard Lundmark

Definition in file [AllAccumulatedStatistics.cc](#).

6.2 AllAccumulatedStatistics.hh File Reference

Header file for the [AllAccumulatedStatistics](#) class.

```
#include "globals.hh"  
#include "DetectorConstruction.hh"  
#include "EventHit.hh"  
#include "G4UnitsTable.hh"  
#include <vector>
```

Classes

- class [AllAccumulatedStatistics](#)

Small class to keep track of the accumulated statistics.

Defines

- `#define VECTOR 1`
- `#define AllAccumulatedStatistics_hh 1`
- `#define VECTOR 1`
- `#define AllAccumulatedStatistics_hh 1`

6.2.1 Detailed Description

Header file for the [AllAccumulatedStatistics](#) class.

Author

Rikard Lundmark

Definition in file [AllAccumulatedStatistics.hh](#).

6.3 DetectorConstruction.cc File Reference

Source file for the [DetectorConstruction](#) class.

```
#include "DetectorConstruction.hh"
```

6.3.1 Detailed Description

Source file for the [DetectorConstruction](#) class.

Author

Philippe Klintefelt Collet, Carl Toft, Rikard Lundmark

Definition in file [DetectorConstruction.cc](#).

6.4 DetectorConstruction.hh File Reference

Header file for the [DetectorConstruction](#) class.

```
#include "G4VUserDetectorConstruction.hh"
#include "globals.hh"
#include "G4FieldManager.hh"
#include "G4TransportationManager.hh"
#include "G4RunManager.hh"
```

```
#include "DetectorMessenger.hh"
#include "G4Material.hh"
#include "G4Tubs.hh"
#include "G4Trap.hh"
#include "G4Box.hh"
#include "G4LogicalVolume.hh"
#include "G4PVPlacement.hh"
#include "G4PVReplica.hh"
#include "G4GeometryManager.hh"
#include "G4PhysicalVolumeStore.hh"
#include "G4LogicalVolumeStore.hh"
#include "G4SolidStore.hh"
#include "G4VisAttributes.hh"
#include "G4Colour.hh"
```

Classes

- class [DetectorConstruction](#)

This class constructs the detector, according to specifications.

6.4.1 Detailed Description

Header file for the [DetectorConstruction](#) class.

Author

Philippe Klintefelt Collet, Carl Toft, Rikard Lundmark

Definition in file [DetectorConstruction.hh](#).

6.5 DetectorMessenger.cc File Reference

Source file for the [DetectorMessenger](#) class.

```
#include "DetectorMessenger.hh"
```

6.5.1 Detailed Description

Source file for the [DetectorMessenger](#) class.

Author

Carl Toft, Rikard Lundmark

Definition in file [DetectorMessenger.cc](#).

6.6 DetectorMessenger.hh File Reference

Header file for the [DetectorMessenger](#) class.

```
#include "globals.hh"
#include "G4UIessenger.hh"
#include "DetectorConstruction.hh"
#include "G4UIDirectory.hh"
#include "G4UIcmdWithAString.hh"
#include "G4UIcmdWithAnInteger.hh"
#include "G4UIcmdWithADoubleAndUnit.hh"
#include "G4UIcmdWithoutParameter.hh"
#include "G4UIcmdWithABool.hh"
```

Classes

- class [DetectorMessenger](#)

Creates commands that the user can invoke via the command line.

6.6.1 Detailed Description

Header file for the [DetectorMessenger](#) class. <Inclusion guard

Author

Carl Toft, Rikard Lundmark

Carl Toft, Rikard Lundmark Header file for the [DetectorMessenger](#) class

Definition in file [DetectorMessenger.hh](#).

6.7 EventAction.cc File Reference

Source file for the [EventAction](#) class.

```
#include "EventAction.hh"
```


6.7.1 Detailed Description

Source file for the [EventAction](#) class.

Author

Carl Toft, Rikard Lundmark

Definition in file [EventAction.cc](#).

6.8 EventAction.hh File Reference

Header file for the [EventAction](#) class.

```
#include "G4UserEventAction.hh"
#include "globals.hh"
#include "RunAction.hh"
#include "EventActionMessenger.hh"
#include "G4Event.hh"
#include "G4TrajectoryContainer.hh"
#include "G4VTrajectory.hh"
#include "G4VVisManager.hh"
#include "G4UnitsTable.hh"
#include "DetectorConstruction.hh"
#include "EventHit.hh"
#include "AllAccumulatedStatistics.hh"
#include "Randomize.hh"
#include <iomanip>
#include <stdio.h>
```

Classes

- class [EventAction](#)

Holds information about what to do at the beginning and end of each event.

Defines

- #define [STDIO](#)

<Inclusion guard

6.8.1 Detailed Description

Header file for the [EventAction](#) class.

Author

Carl Toft, Rikard Lundmark

Definition in file [EventAction.hh](#).

6.8.2 Define Documentation

6.8.2.1 #define STDIO

<Inclusion guard

Inclusion guard

Definition at line 39 of file EventAction.hh.

6.9 EventActionMessenger.cc File Reference

Source file for the [EventActionMessenger](#) class.

```
#include "EventActionMessenger.hh"
```

6.9.1 Detailed Description

Source file for the [EventActionMessenger](#) class.

Author

Carl Toft, Rikard Lundmark

Definition in file [EventActionMessenger.cc](#).

6.10 EventActionMessenger.hh File Reference

Header file for the [EventActionMessenger](#) class.

```
#include "globals.hh"
#include "G4UImessenger.hh"
#include "EventAction.hh"
#include "G4UIDirectory.hh"
#include "G4UIcmdWithAnInteger.hh"
#include "G4UIcmdWithAString.hh"
```

Classes

- class [EventActionMessenger](#)

Creates commands that can be used to control the EventActions via the command line.

6.10.1 Detailed Description

Header file for the [EventActionMessenger](#) class.

Author

Carl Toft, Rikard Lundmark

Definition in file [EventActionMessenger.hh](#).

6.11 EventHit.cc File Reference

Source file for the [EventHit](#) class.

```
#include "EventHit.hh"
```

6.11.1 Detailed Description

Source file for the [EventHit](#) class.

Author

Rikard Lundmark

Definition in file [EventHit.cc](#).

6.12 EventHit.hh File Reference

Header file for the [EventHit](#) class.

```
#include "globals.hh"
#include "G4UnitsTable.hh"
#include <vector>
```

Classes

- class [EventHit](#)

Holds accumulated statistics for a single event hit.

Defines

- `#define EventHit_hh 1`

6.12.1 Detailed Description

Header file for the [EventHit](#) class.

Author

Rikard Lundmark

Definition in file [EventHit.hh](#).

6.13 G4EmStandardFysik.cc File Reference

Source file for the [G4EmStandardFysik](#) class.

```
#include "G4EmStandardFysik.hh"
#include "G4EmStandardPhysics.hh"
#include "G4ParticleDefinition.hh"
#include "G4ProcessManager.hh"
#include "G4LossTableManager.hh"
#include "G4EmProcessOptions.hh"
#include "G4ComptonScattering.hh"
#include "G4GammaConversion.hh"
#include "G4PhotoElectricEffect.hh"
#include "G4eMultipleScattering.hh"
#include "G4MuMultipleScattering.hh"
#include "G4hMultipleScattering.hh"
#include "G4CoulombScattering.hh"
#include "G4WentzelVIModel.hh"
#include "G4eIonisation.hh"
#include "G4eBremsstrahlung.hh"
#include "G4eplusAnnihilation.hh"
#include "G4MuIonisation.hh"
#include "G4MuBremsstrahlung.hh"
#include "G4MuPairProduction.hh"
#include "G4hBremsstrahlung.hh"
```

```
#include "G4hPairProduction.hh"
#include "G4hIonisation.hh"
#include "G4ionIonisation.hh"
#include "G4alphaIonisation.hh"
#include "G4Gamma.hh"
#include "G4Electron.hh"
#include "G4Positron.hh"
#include "G4MuonPlus.hh"
#include "G4MuonMinus.hh"
#include "G4PionPlus.hh"
#include "G4PionMinus.hh"
#include "G4KaonPlus.hh"
#include "G4KaonMinus.hh"
#include "G4Proton.hh"
#include "G4AntiProton.hh"
#include "G4Deuteron.hh"
#include "G4Triton.hh"
#include "G4He3.hh"
#include "G4Alpha.hh"
#include "G4GenericIon.hh"
```

6.13.1 Detailed Description

Source file for the [G4EmStandardFysik](#) class.

Author

Rikard Lundmark

Definition in file [G4EmStandardFysik.cc](#).

6.14 G4EmStandardFysik.hh File Reference

Header file for the [G4EmStandardFysik](#) class.

```
#include "G4EmStandardPhysics.hh"
```

Classes

- class [G4EmStandardFysik](#)

This class is extending the [G4EmStandardPhysics](#) class in order to apply a few changes to this physics list.

6.14.1 Detailed Description

Header file for the [G4EmStandardFysik](#) class.

Author

Rikard Lundmark

Definition in file [G4EmStandardFysik.hh](#).

6.15 MyPhysicsList.hh File Reference

Header file for the [MyPhysicsList](#) class.

```
#include "G4VModularPhysicsList.hh"
#include "globals.hh"
#include "CompileTimeConstraints.hh"
#include "../src/MyPhysicsList.icc"
```

Classes

- class [MyPhysicsList< T >](#)

Customized physics list class.

6.15.1 Detailed Description

Header file for the [MyPhysicsList](#) class.

Author

Rikard Lundmark

Definition in file [MyPhysicsList.hh](#).

6.16 MyPhysicsList.icc File Reference

Source file for the [MyPhysicsList](#) class.

```
#include "globals.hh"
#include "G4ParticleDefinition.hh"
#include "G4ParticleWithCuts.hh"
#include "G4ProcessManager.hh"
#include "G4ProcessVector.hh"
#include "G4ParticleTypes.hh"
#include "G4ParticleTable.hh"
#include "G4Material.hh"
#include "G4MaterialTable.hh"
#include "G4ios.hh"
#include <iomanip>
#include "G4DecayPhysics.hh"
#include "G4EmStandardPhysics.hh"
#include "G4EmExtraPhysics.hh"
#include "G4IonPhysics.hh"
#include "G4RadioactiveDecayPhysics.hh"
#include "G4QStoppingPhysics.hh"
#include "G4HadronElasticPhysics.hh"
#include "G4NeutronTrackingCut.hh"
#include "G4EmStandardFysik.hh"
#include "G4DataQuestionnaire.hh"
#include "HadronPhysicsQGSP_BERT.hh"
```

6.16.1 Detailed Description

Source file for the [MyPhysicsList](#) class.

Author

Rikard Lundmark

Definition in file [MyPhysicsList.icc](#).

6.17 PIBIDS.cc File Reference

Main file, starting the simulation.

```
#include "G4RunManager.hh"
```

```
#include "G4UImanager.hh"
#include "Randomize.hh"
#include <ctime>
#include "MyPhysicsList.hh"
#include "DetectorConstruction.hh"
#include "PrimaryGeneratorAction.hh"
#include "RunAction.hh"
#include "EventAction.hh"
#include "SteppingAction.hh"
#include "SteppingVerbose.hh"
#include <stdio.h>
```

Defines

- `#define` [STDIO](#)
Inclusion guard.

Functions

- `int` [main](#) (int argc, char **argv)
Main function, starting the program.

6.17.1 Detailed Description

Main file, starting the simulation.

Author

Rikard Lundmark

Definition in file [PIBIDS.cc](#).

6.17.2 Function Documentation

6.17.2.1 `int main (int argc, char ** argv)`

Main function, starting the program.

Parameters

<i>argc</i>	Number of arguments.
<i>argv</i>	The actual arguments.

Definition at line 38 of file PIBIDS.cc.

References PrimaryGeneratorAction::SetParticle().

```
{
    // Choose the Random engine
    //
    srand(time(0));
    CLHEP::HepRandom::setTheEngine(new CLHEP::RanecuEngine);
    CLHEP::HepRandom::setTheSeed(rand()%10000+1);

    // User Verbose output class
    //
    G4VSteppingVerbose::SetInstance(new SteppingVerbose);

    // Construct the default run manager
    //
    G4RunManager * runManager = new G4RunManager;

    // Set mandatory initialization classes
    //
    DetectorConstruction* detector = new DetectorConstruction;
    runManager->SetUserInitialization(detector);
    //
    G4VUserPhysicsList* physics = new MyPhysicsList<G4VModularPhysicsList>;
    runManager->SetUserInitialization(physics);

    // Set user action classes
    //

    //
    RunAction* run_action = new RunAction;
    //

    PrimaryGeneratorAction * myAction =
        new PrimaryGeneratorAction(detector, run_action);

    G4VUserPrimaryGeneratorAction* gen_action = myAction;

    runManager->SetUserAction(gen_action);

    runManager->SetUserAction(run_action);

    EventAction* event_action = new EventAction(run_action, detector);
    runManager->SetUserAction(event_action);
    //
    G4UserSteppingAction* stepping_action =
        new SteppingAction(detector, event_action);
    runManager->SetUserAction(stepping_action);

    // Initialize G4 kernel
    //
    runManager->Initialize();

    myAction->SetParticle(6,17,0.,0.);

#ifdef G4VIS_USE
    // Initialize visualization
    //
    G4VisManager* visManager = new G4VisExecutive;
```

```

    visManager->Initialize();
#endif

// Get the pointer to the User Interface manager
//
G4UImanager* UImanager = G4UImanager::GetUIpointer();

if (argc==2)    // batch mode
{
    G4String command = "/control/execute ";
    G4String fileName = argv[1];
    UImanager->ApplyCommand(command+fileName);
}
else
{
    // interactive mode : define UI session
#ifdef G4UI_USE
    G4UIExecutive* ui = new G4UIExecutive(argc, argv);
#endif
#ifdef G4VIS_USE
    if(argc<2)
        UImanager->ApplyCommand("/control/execute vis.mac");
    else //other init file.
    {
        G4String command = "/control/execute ";
        G4String fileName = argv[1];
        UImanager->ApplyCommand(command+fileName);
    }
#endif
    if (ui->IsGUI())
        UImanager->ApplyCommand("/control/execute visTutor/gui.mac");
    ui->SessionStart();
    delete ui;
#endif
}

// Job termination
// Free the store: user actions, physics_list and detector_description are
// owned and deleted by the run manager, so they should not
// be deleted in the main() program !
#ifdef G4VIS_USE
    delete visManager;
#endif
delete runManager;

return 0;
}

```

6.18 PrimaryGeneratorAction.cc File Reference

Source file for the [PrimaryGeneratorAction](#) class.

```

#include "PrimaryGeneratorAction.hh"
#include "DetectorConstruction.hh"
#include "PrimaryGeneratorMessenger.hh"
#include "G4ios.hh"

```

6.18.1 Detailed Description

Source file for the [PrimaryGeneratorAction](#) class.

Author

Carl Toft, Rikard Lundmark

Definition in file [PrimaryGeneratorAction.cc](#).

6.19 PrimaryGeneratorAction.hh File Reference

Header file for the [PrimaryGeneratorAction](#) class.

```
#include "G4VUserPrimaryGeneratorAction.hh"
#include "globals.hh"
#include "G4ParticleTable.hh"
#include "G4UImessenger.hh"
#include "G4ParticleMessenger.hh"
#include "G4IonTable.hh"
#include "G4ShortLivedTable.hh"
#include "G4Ions.hh"
#include "G4Event.hh"
#include "G4ParticleGun.hh"
#include "G4ParticleDefinition.hh"
#include "Randomize.hh"
#include "RunAction.hh"
```

Classes

- class [PrimaryGeneratorAction](#)
Mandatory user class providing the primary particle generator.

6.19.1 Detailed Description

Header file for the [PrimaryGeneratorAction](#) class.

Author

Carl Toft, Rikard Lundmark

Definition in file [PrimaryGeneratorAction.hh](#).

6.20 PrimaryGeneratorMessenger.cc File Reference

Source file for the [PrimaryGeneratorMessenger](#) class.

```
#include "PrimaryGeneratorMessenger.hh"
```

6.20.1 Detailed Description

Source file for the [PrimaryGeneratorMessenger](#) class.

Author

Carl Toft, Rikard Lundmark

Definition in file [PrimaryGeneratorMessenger.cc](#).

6.21 PrimaryGeneratorMessenger.hh File Reference

Header file for the [PrimaryGeneratorMessenger](#) class.

```
#include "G4UImessenger.hh"  
#include "globals.hh"  
#include "PrimaryGeneratorAction.hh"  
#include "G4UIDirectory.hh"  
#include "G4UIcmdWithAString.hh"  
#include "G4UIcmdWithAnInteger.hh"  
#include "G4UIcmdWithADouble.hh"  
#include "G4UIcmdWithoutParameter.hh"
```

Classes

- class [PrimaryGeneratorMessenger](#)
Creates commands that allows the user to control the PrimaryGenerator via the command line.
- class [PrimaryGeneratorMessenger::RunActionMessenger](#)
Creates commands that can be used to control the [RunAction](#) via the command line.
- class [PrimaryGeneratorMessenger::DetectorMessenger](#)
Creates commands that the user can invoke via the command line.
- class [PrimaryGeneratorMessenger::DetectorConstruction](#)
This class constructs the detector, according to specifications.
- class [PrimaryGeneratorMessenger::EventHit](#)
Holds accumulated statistics for a single event hit.

- class [PrimaryGeneratorMessenger::AllAccumulatedStatistics](#)
Small class to keep track of the accumulated statistics.
- class [PrimaryGeneratorMessenger::EventHit](#)
Holds accumulated statistics for a single event hit.
- class [PrimaryGeneratorMessenger::PrimaryGeneratorAction](#)
Mandatory user class providing the primary particle generator.

6.21.1 Detailed Description

Header file for the [PrimaryGeneratorMessenger](#) class.

Author

Carl Toft, Rikard Lundmark

Definition in file [PrimaryGeneratorMessenger.hh](#).

6.22 RunAction.cc File Reference

Source file for the [RunAction](#) class.

```
#include "RunAction.hh"
```

6.22.1 Detailed Description

Source file for the [RunAction](#) class.

Author

Carl Toft, Rikard Lundmark

Definition in file [RunAction.cc](#).

6.23 RunAction.hh File Reference

<Inclusion guard.

```
#include <stdio.h>
#include "G4Run.hh"
#include "G4RunManager.hh"
#include "G4UnitsTable.hh"
#include "RunActionMessenger.hh"
#include "AllAccumulatedStatistics.hh"
```

```
#include "EventHit.hh"
#include "G4UserRunAction.hh"
#include "globals.hh"
```

Classes

- class [RunAction](#)

Here we can put stuff to be executed before and after each run.

Defines

- #define [RunAction_h](#) 1

Inclusion guard.

6.23.1 Detailed Description

<Inclusion guard. Header file for the [RunAction](#) class.

Author

Carl Toft, Rikard Lundmark Header file for the [RunAction](#) class
Carl Toft, Rikard Lundmark

Definition in file [RunAction.hh](#).

6.24 RunActionMessenger.cc File Reference

Source file for the [RunActionMessenger](#) class.

```
#include "RunActionMessenger.hh"
```

6.24.1 Detailed Description

Source file for the [RunActionMessenger](#) class.

Author

Rikard Lundmark

Definition in file [RunActionMessenger.cc](#).

6.25 RunActionMessenger.hh File Reference

Header file for the [RunActionMessenger](#) class.

```
#include "globals.hh"
#include "G4UI messenger.hh"
#include "RunAction.hh"
#include "G4UI directory.hh"
#include "G4UIcmdWithAString.hh"
#include "G4UIcmdWithABool.hh"
```

Classes

- class [RunActionMessenger](#)

Creates commands that can be used to control the [RunAction](#) via the command line.

6.25.1 Detailed Description

Header file for the [RunActionMessenger](#) class.

Author

Rikard Lundmark

Definition in file [RunActionMessenger.hh](#).

6.26 SteppingAction.cc File Reference

Source file for the [SteppingAction](#) class.

```
#include "SteppingAction.hh"
```

6.26.1 Detailed Description

Source file for the [SteppingAction](#) class.

Author

Carl Toft, Rikard Lundmark

Definition in file [SteppingAction.cc](#).

6.27 SteppingAction.hh File Reference

Header file for the [SteppingAction](#) class.

```
#include "G4UserSteppingAction.hh"
#include "DetectorConstruction.hh"
#include "EventAction.hh"
#include "G4UnitsTable.hh"
#include "G4Step.hh"
#include <stdio.h>
#include <sstream>
```

Classes

- class [SteppingAction](#)

Contains information about what to do at each step in the simulation.

Defines

- #define [STDIO_H](#)
<Inclusion guard.
- #define **SSTREAM**

6.27.1 Detailed Description

Header file for the [SteppingAction](#) class.

Author

Carl Toft, Rikard Lundmark

Definition in file [SteppingAction.hh](#).

6.27.2 Define Documentation

6.27.2.1 #define STDIO_H

<Inclusion guard.

Inclusion guard.

Definition at line 26 of file SteppingAction.hh.

6.28 SteppingVerbose.cc File Reference

Source file for the [SteppingVerbose](#) class.

```
#include "SteppingVerbose.hh"  
#include "G4SteppingManager.hh"  
#include "G4UnitsTable.hh"
```

6.28.1 Detailed Description

Source file for the [SteppingVerbose](#) class.

Author

Carl Toft, Rikard Lundmark

Definition in file [SteppingVerbose.cc](#).

6.29 SteppingVerbose.hh File Reference

Header file for the [SteppingVerbose](#) class.

```
#include "G4SteppingVerbose.hh"
```

Classes

- class [SteppingVerbose](#)
Inclusion guard.

6.29.1 Detailed Description

Header file for the [SteppingVerbose](#) class.

Author

Carl Toft, Rikard Lundmark

Definition in file [SteppingVerbose.hh](#).