

beam_dump Reference Manual

0.1

Generated by Doxygen 1.5.3

Thu Sep 25 17:52:06 2008

Contents

1	beam_dump Hierarchical Index	1
1.1	beam_dump Class Hierarchy	1
2	beam_dump Class Index	3
2.1	beam_dump Class List	3
3	beam_dump File Index	5
3.1	beam_dump File List	5
4	beam_dump Class Documentation	7
4.1	Cone Class Reference	7
4.2	Cylinder Class Reference	9
4.3	Element Class Reference	10
4.4	Geometry Class Reference	11
4.5	Node Class Reference	13
4.6	Particle Class Reference	14
4.7	Simulation Class Reference	15
4.8	TwoPlates Class Reference	16
5	beam_dump File Documentation	17
5.1	src/cone.h File Reference	17
5.2	src/cylinder.h File Reference	19
5.3	src/element.h File Reference	20
5.4	src/geometry.h File Reference	21
5.5	src/node.h File Reference	22
5.6	src/particle.h File Reference	23
5.7	src/simulation.h File Reference	24
5.8	src/twoplates.h File Reference	25

Chapter 1

beam_dump Hierarchical Index

1.1 beam_dump Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Element	10
Geometry	11
Cone	7
Cylinder	9
TwoPlates	16
Node	13
Particle	14
Simulation	15

Chapter 2

beam_dump Class Index

2.1 beam_dump Class List

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

Cone	7
Cylinder	9
Element	10
Geometry	11
Node	13
Particle	14
Simulation	15
TwoPlates	16

Chapter 3

beam_dump File Index

3.1 beam_dump File List

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

src/cone.h (Pure conical geometry)	17
src/cylinder.h (Pure cylindrical geometry)	19
src/element.h (Facet planar geometry for graphical representation purposes)	20
src/geometry.h (Base class for the different geometries)	21
src/node.h (Node class for graphical representation purposes)	22
src/particle.h (Represents each of the finite charged particles)	23
src/simulation.h (Procedures for the program workflow)	24
src/twoplates.h (Two symmetrical plates geometry)	25

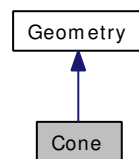
Chapter 4

beam_dump Class Documentation

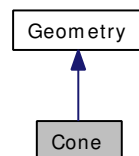
4.1 Cone Class Reference

```
#include </home/dani/Programs/beam_dump/src/cone.h>
```

Inheritance diagram for Cone:



Collaboration diagram for Cone:



Public Member Functions

- **Cone** (std::string, double, double, int)
- void **computeEnergy** (double, std::vector< [Particle](#) * > &)
- void **setSections** (double)
- void **computeGeometry** ()
- void **outputTable** ()
- void **outputEnergyFile** ()
- void **outputPowerFile** ()

4.1.1 Detailed Description

Author:

Daniel Iglesias <daniel.iglesias@ciemat.es>

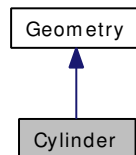
The documentation for this class was generated from the following files:

- [src/cone.h](#)
- [src/cone.cpp](#)

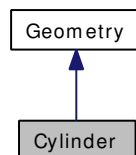
4.2 Cylinder Class Reference

```
#include </home/dani/Programs/beam_dump/src/cylinder.h>
```

Inheritance diagram for Cylinder:



Collaboration diagram for Cylinder:



Public Member Functions

- **Cylinder** (std::string, double, double, int)
- void **computeEnergy** (double, std::vector< [Particle](#) * > &)
- void **setSections** (double)
- void **computeGeometry** ()
- void **outputTable** ()
- void **outputEnergyFile** ()
- void **outputPowerFile** ()

4.2.1 Detailed Description

Author:

Daniel Iglesias <daniel.iglesias@ciemat.es>

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

- [src/cylinder.h](#)
- [src/cylinder.cpp](#)

4.3 Element Class Reference

```
#include </home/dani/Programs/beam_dump/src/element.h>
```

Public Member Functions

- **Element** (int type_in)
- void **setNumber** (int number_in)
- void **setArea** (double area_in)
- void **addNode** (int node_number)
- int **getType** ()
- int **getNumber** ()
- double **getArea** ()
- int **getNumberOfNodes** ()
- std::vector< int > & **getConnectivity** ()
- void **generateGeometry** ()

Public Attributes

- vtkCell * **geometry**

4.3.1 Detailed Description

Author:

Daniel Iglesias <daniel.iglesias@ciemat.es>

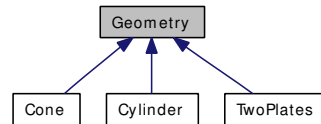
The documentation for this class was generated from the following files:

- src/[element.h](#)
- src/element.cpp

4.4 Geometry Class Reference

```
#include </home/dani/Programs/beam_dump/src/geometry.h>
```

Inheritance diagram for Geometry:



Public Member Functions

- **Geometry** (std::string, int)
- virtual void **computeEnergy** (double, std::vector< [Particle](#) * > &)=0
- virtual void **computeIncrements** ()
- virtual void **setSections** (double)=0
- virtual void **computeGeometry** ()=0
- virtual void **computeNodalEnergy** ()
- virtual void **computeNodalPower** ()
- virtual void **computeNodalPower** (std::vector< double > &)
- void **drawGeometry** ()
- void **drawScalar** ()
- virtual void **outputEnergyFile** ()=0
- virtual void **outputPowerFile** ()=0
- virtual void **outputTable** ()=0
- std::string **getType** ()

Protected Attributes

- std::string **type**
- int **sections**
- int **sectors**
- double **slope**
- std::map< double, std::vector< double > > **energies**
- std::map< double, std::vector< double > > **powers**
- std::map< int, [Node](#) * > **nodes**
- std::vector< [Element](#) * > **elements**
- std::string **plotTitle**
- vtkPoints * **gridPoints**
- vtkFloatArray * **scalar**
- vtkUnstructuredGrid * **grid**
- vtkRenderWindow * **renWin**
- vtkInteractorStyleTrackballCamera * **style**
- vtkRenderWindowInteractor * **iren**
- vtkRenderer * **ren**
- vtkDataSetMapper * **aDataSetMapper**
- vtkActor * **anActor**
- vtkLookupTable * **table**

- vtkScalarBarActor * **barActor**
- vtkAxesActor * **axes**
- vtkOrientationMarkerWidget * **widget**

4.4.1 Detailed Description

Author:

Daniel Iglesias <daniel.iglesias@ciemat.es>

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

- [src/geometry.h](#)
- [src/geometry.cpp](#)

4.5 Node Class Reference

```
#include </home/dani/Programs/beam_dump/src/node.h>
```

Public Member Functions

- **Node** (double, double, double)
- void **setScalar** (double scalar_in)
- void **addScalar** (double scalar_in)
- double & **getScalar** ()
- double & **getX** ()
- double & **getY** ()
- double & **getZ** ()

4.5.1 Detailed Description

Author:

Daniel Iglesias <daniel.iglesias@ciemat.es>

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

- src/[node.h](#)
- src/node.cpp

4.6 Particle Class Reference

```
#include </home/dani/Programs/beam_dump/src/particle.h>
```

Public Member Functions

- **Particle** (double, double, double, double, double, double, double, double, double, double)
- double & **getX** ()
- double & **getY** ()
- double & **getEnergy** ()

4.6.1 Detailed Description

Author:

Daniel Iglesias <daniel.iglesias@ciemat.es>

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

- [src/particle.h](#)
- [src/particle.cpp](#)

4.7 Simulation Class Reference

```
#include </home/dani/Programs/beam_dump/src/simulation.h>
```

Public Member Functions

- void **read** (char *)
- void **compute** ()
- void **output** ()

4.7.1 Detailed Description

Author:

Daniel Iglesias <daniel.iglesias@ciemat.es>

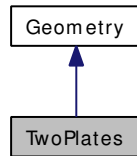
The documentation for this class was generated from the following files:

- src/[simulation.h](#)
- src/simulation.cpp

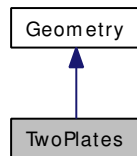
4.8 TwoPlates Class Reference

```
#include </home/dani/Programs/beam_dump/src/twoplates.h>
```

Inheritance diagram for TwoPlates:



Collaboration diagram for TwoPlates:



Public Member Functions

- **TwoPlates** (std::string, double, double, int, double, double)
- void **setSections** (double)
- void **computeEnergy** (double, std::vector< [Particle](#) * > &)
- void **computeGeometry** ()
- void **outputTable** ()
- void **outputEnergyFile** ()
- void **outputPowerFile** ()
- double **getLength** ()

4.8.1 Detailed Description

Author:

Daniel Iglesias <daniel.iglesias@ciemat.es>

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

- [src/twoplates.h](#)
- [src/twoplates.cpp](#)

Chapter 5

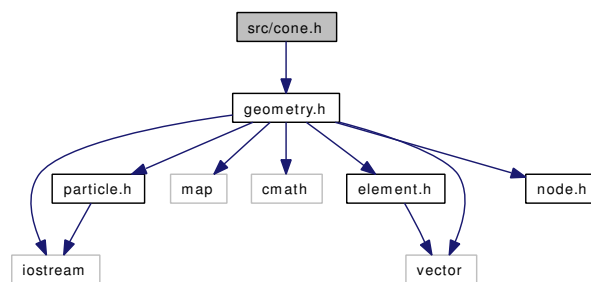
beam_dump File Documentation

5.1 src/cone.h File Reference

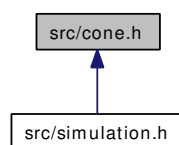
Pure conical geometry.

```
#include "geometry.h"
```

Include dependency graph for cone.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [Cone](#)

5.1.1 Detailed Description

Pure conical geometry.

[Geometry](#) with origin in `s=0` defined by `s-length` and initial diameter (constant slope).

Author:

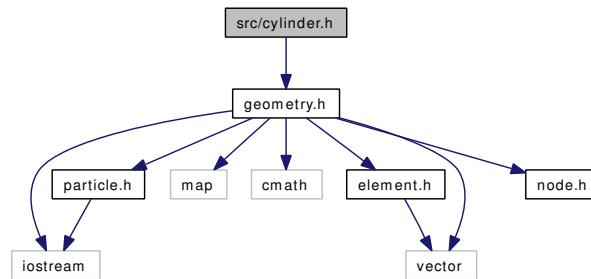
Daniel Iglesias <daniel.iglesias@ciemat.es>

5.2 src/cylinder.h File Reference

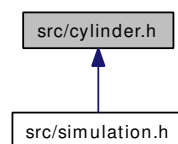
Pure cylindrical geometry.

```
#include "geometry.h"
```

Include dependency graph for cylinder.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [Cylinder](#)

5.2.1 Detailed Description

Pure cylindrical geometry.

[Geometry](#) with origin in `s=0` defined by `s`-length and diameter.

Author:

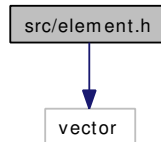
Daniel Iglesias <daniel.iglesias@ciemat.es>

5.3 src/element.h File Reference

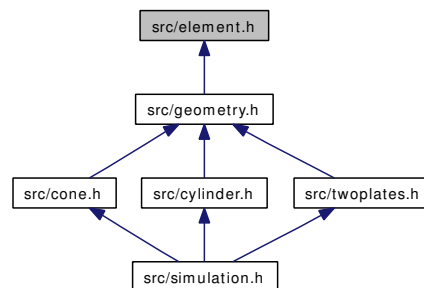
Facet planar geometry for graphical representation purposes.

```
#include <vector>
```

Include dependency graph for element.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [Element](#)

5.3.1 Detailed Description

Facet planar geometry for graphical representation purposes.

Uses the [Node](#) class for the defining the vertices. Usually, the dimensions are proportional to the number of sections (longitudinal) and sectors (transversal divisions).

Author:

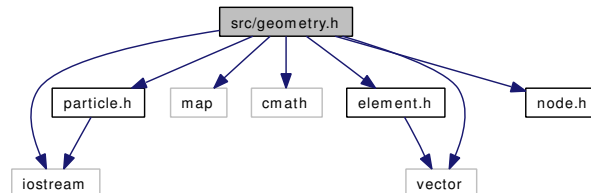
Daniel Iglesias <daniel.iglesias@ciemat.es>

5.4 src/geometry.h File Reference

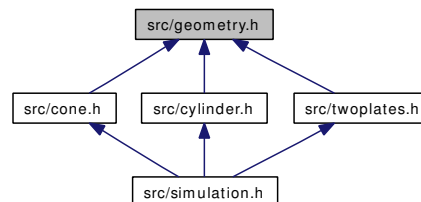
Base class for the different geometries.

```
#include <iostream>
#include <vector>
#include <map>
#include <cmath>
#include "particle.h"
#include "element.h"
#include "node.h"
```

Include dependency graph for geometry.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [Geometry](#)

5.4.1 Detailed Description

Base class for the different geometries.

Abstract class, cannot be instantiated.

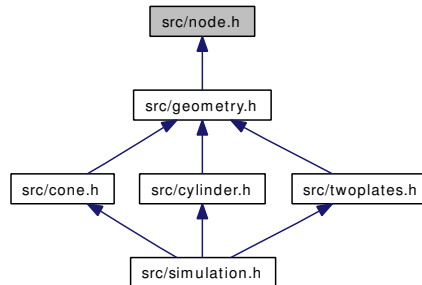
Author:

Daniel Iglesias <daniel.iglesias@ciemat.es>

5.5 src/node.h File Reference

[Node](#) class for graphical representation purposes.

This graph shows which files directly or indirectly include this file:



Classes

- class [Node](#)

5.5.1 Detailed Description

[Node](#) class for graphical representation purposes.

Simple point definition and manipulation. It has a scalar property for storing the power density.

Author:

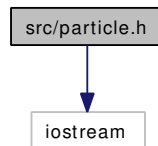
Daniel Iglesias <daniel.iglesias@ciemat.es>

5.6 src/particle.h File Reference

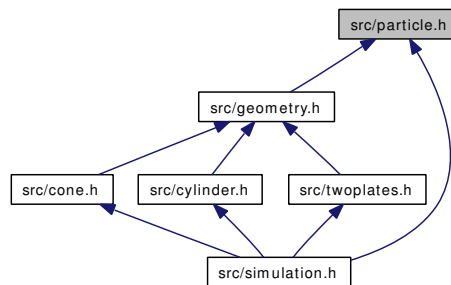
Represents each of the finite charged particles.

```
#include <iostream>
```

Include dependency graph for particle.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [Particle](#)

5.6.1 Detailed Description

Represents each of the finite charged particles.

Very basic class with stored the x-y position in the plane section and the energy of the particle.

Author:

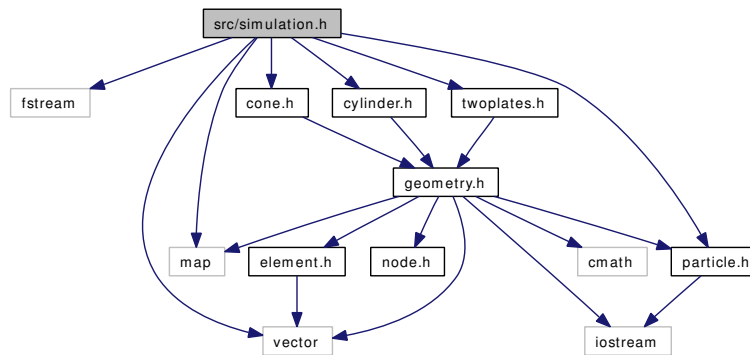
Daniel Iglesias <daniel.iglesias@ciemat.es>

5.7 src/simulation.h File Reference

Procedures for the program workflow.

```
#include <fstream>
#include <vector>
#include <map>
#include "cone.h"
#include "cylinder.h"
#include "twoplates.h"
#include "particle.h"
```

Include dependency graph for simulation.h:



Classes

- class [Simulation](#)

5.7.1 Detailed Description

Procedures for the program workflow.

Reads files, creates all of the objects, launches the computation and creates the output.

Author:

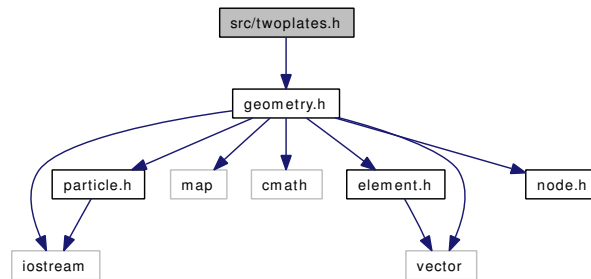
Daniel Iglesias <daniel.iglesias@ciemat.es>

5.8 src/twoplates.h File Reference

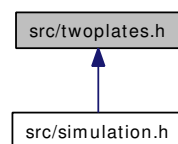
Two symmetrical plates geometry.

```
#include <geometry.h>
```

Include dependency graph for twoplates.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [TwoPlates](#)

5.8.1 Detailed Description

Two symmetrical plates geometry.

[Geometry](#) with origin in $s=0$ defined by s -length and initial plate separation (constant slope).

Author:

Daniel Iglesias <daniel.iglesias@ciemat.es>

Index

Cone, [7](#)

Cylinder, [9](#)

Element, [10](#)

Geometry, [11](#)

Node, [13](#)

Particle, [14](#)

Simulation, [15](#)

src/cone.h, [17](#)

src/cylinder.h, [19](#)

src/element.h, [20](#)

src/geometry.h, [21](#)

src/node.h, [22](#)

src/particle.h, [23](#)

src/simulation.h, [24](#)

src/twoplates.h, [25](#)

TwoPlates, [16](#)