Raytracer 3D

Generated by Doxygen 1.8.12

Contents

Chapter 1

Class Index

1.1 Class List

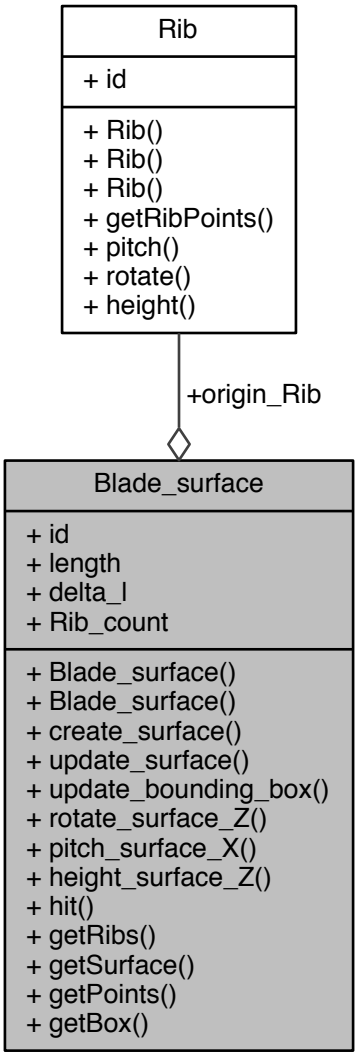
Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 2

Class Documentation

2.1 Blade_surface Class Reference

Collaboration diagram for Blade_surface:



Public Member Functions

- **Blade_surface** (const int id, const double length, const int Rib_count, Rib &origin_Rib)
- void **create_surface** ()
- void **update_surface** ()
- void **update_bounding_box** ()
- void **rotate_surface_Z** (double angle)
- void **pitch_surface_X** (const double angle)
- void **height_surface_Z** (const double height)
- bool **hit** (const Ray3D &ray, double &hitDistance, Vector3D &hitNormal, Point3D &hitPoint)

- std::vector< Rib > **getRibs** ()
- std::vector< Triangle > **getSurface** ()
- std::vector< Point3D > **getPoints** ()
- Bounding_box **getBox** ()

Public Attributes

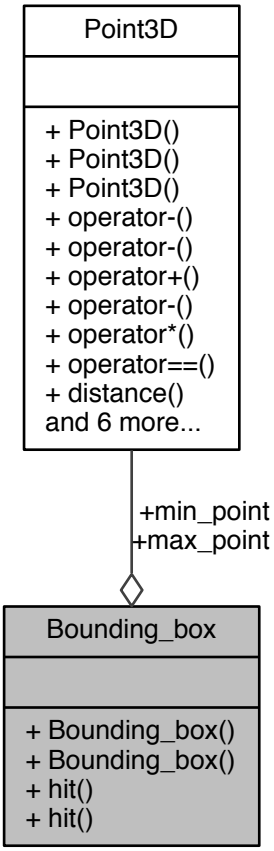
- int **id**
- double **length**
- double **delta_l**
- int **Rib_count**
- Rib **origin_Rib**

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

- Raytracer3D/Raytracer3D/Blade_surface.hpp
- Raytracer3D/Raytracer3D/Blade_surface.cpp

2.2 Bounding_box Class Reference

Collaboration diagram for Bounding_box:



Public Member Functions

- **Bounding_box** (std::vector< Point3D > &points)
- bool **hit** (const Ray3D &ray, Point3D &hit_point) const
- bool **hit** (const Ray3D &ray) const

Public Attributes

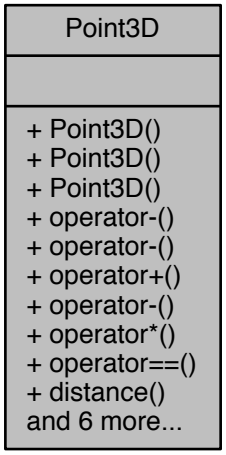
- Point3D **min_point**
- Point3D **max_point**

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

- Raytracer3D/Raytracer3D/Bounding_box.hpp
- Raytracer3D/Raytracer3D/Bounding_box.cpp

2.3 Point3D Class Reference

Collaboration diagram for Point3D:



Public Member Functions

- **Point3D** (const double x, const double y, const double z)
- **Point3D** (const double y, const double z)
- Point3D **operator-** () const
- Vector3D **operator-** (const Point3D &p) const
- Point3D **operator+** (const Vector3D &v) const
- Point3D **operator-** (const Vector3D &v) const

- Point3D **operator**∗ (const double a) const
- bool **operator==** (const Point3D &p) const
- double **distance** (const Point3D &p) const
- double **x** () const
- double **y** () const
- double **z** () const
- void **rotate_Z** (const double angle)
- void **rotate_X** (const double angle)
- void **translate_Z** (const double height)

Friends

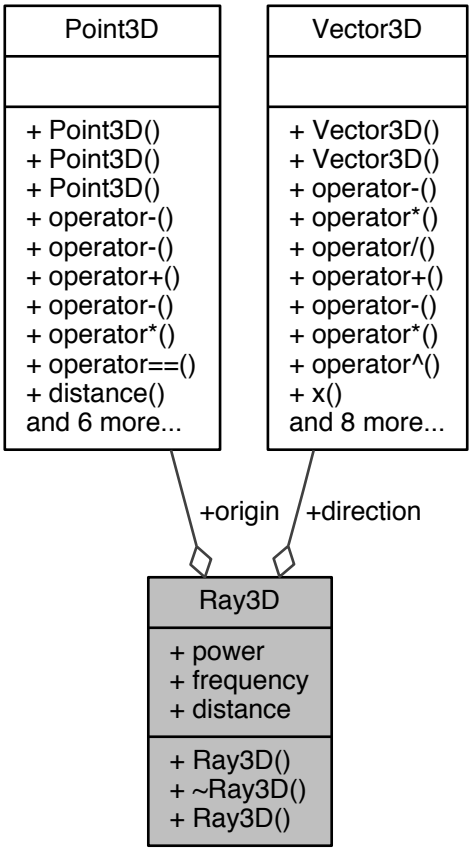
- void **swap** (Point3D &a, Point3D &b)

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

- Raytracer3D/Raytracer3D/Point3d.hpp
- Raytracer3D/Raytracer3D/Point3d.cpp

2.4 Ray3D Class Reference

Collaboration diagram for Ray3D:



Public Member Functions

- **Ray3D** (const Point3D &origin, const Vector3D &direction, const double power=1.0, const double frequency=0.0, const double distance=0)

Public Attributes

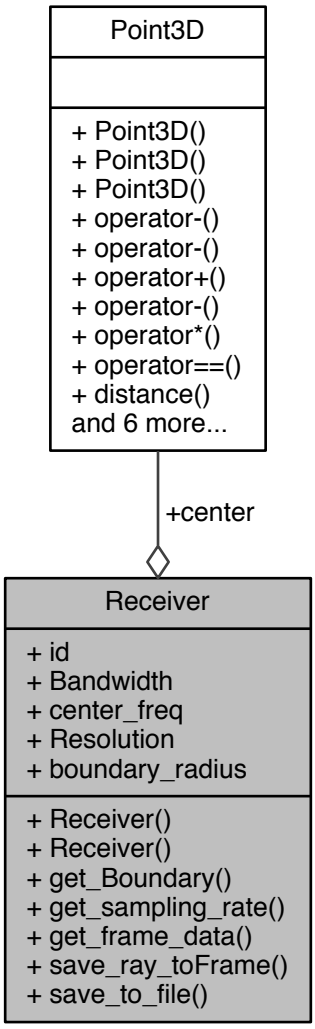
- Point3D **origin**
- Vector3D **direction**
- double **power**
- double **frequency**
- double **distance**

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

- Raytracer3D/Raytracer3D/Ray3d.hpp
- Raytracer3D/Raytracer3D/Ray3d.cpp

2.5 Receiver Class Reference

Collaboration diagram for Receiver:



Public Member Functions

- **Receiver** (const int id, const double Bandwidth, const double center_freq, const Point3D &center, const double boundary_radius, const std::string &savefile_name, const std::string &dopplerfile_name)
- Sphere **get_Boundary** ()
- double **get_sampling_rate** ()
- std::vector< Ray3D > **get_frame_data** ()
- void **save_ray_toFrame** (Ray3D &ray)
- void **save_to_file** ()

Public Attributes

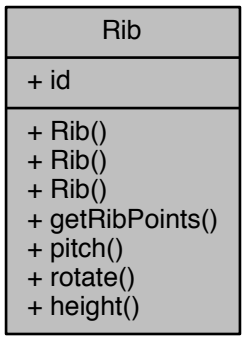
- int **id**
- double **Bandwidth**
- double **center_freq**
- double **Resolution**
- double **boundary_radius**
- Point3D **center**

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

- Raytracer3D/Raytracer3D/Receiver.hpp
- Raytracer3D/Raytracer3D/Receiver.cpp

2.6 Rib Class Reference

Collaboration diagram for Rib:



Public Member Functions

- **Rib** (int id, Rib &x, const double delta_l)
- **Rib** (int id, const std::string &filename)
- std::vector< Point3D > **getRibPoints** ()
- void **pitch** (double angle)
- void **rotate** (double angle)
- void **height** (const double height)

Public Attributes

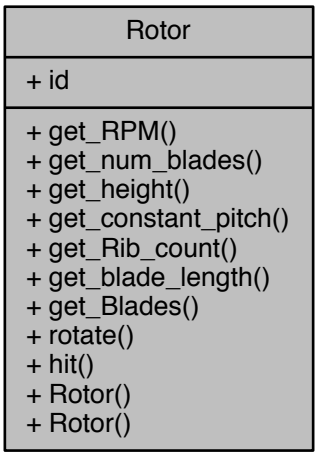
- int **id**

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

- Raytracer3D/Raytracer3D/Rib.hpp
- Raytracer3D/Raytracer3D/Rib.cpp

2.7 Rotor Class Reference

Collaboration diagram for Rotor:



Public Member Functions

- double **get_RPM** ()
- int **get_num_blades** ()
- double **get_height** ()
- double **get_constant_pitch** ()
- double **get_Rib_count** ()
- double **get_blade_length** ()
- std::vector< Blade_surface > **get_Blades** ()
- void **rotate** (const double angle)
- bool **hit** (const Ray3D &ray, double &hitDistance, Vector3D &hitNormal, Point3D &hitPoint)
- **Rotor** (const int id, const int num_blades, const double RPM, const double height, const double constant_↩ pitch, const double blade_length, const int Rib_count)

Public Attributes

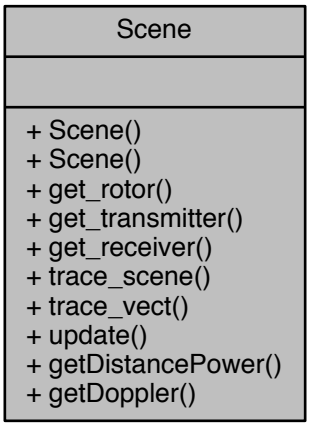
- int **id**

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

- Raytracer3D/Raytracer3D/Rotor.hpp
- Raytracer3D/Raytracer3D/Rotor.cpp

2.8 Scene Class Reference

Collaboration diagram for Scene:



Public Member Functions

- **Scene** (double rx_x, double rx_y, double rx_z, double Bandwidth, double rx_fc, double tx_x, double tx_y, double tx_fc, double tx_power, int num_blades, double RPM, double altitude, double pitch, double blade_↩ length, int num_ribs, const std::string &filename)
- Rotor **get_rotor** ()
- Transmitter **get_transmitter** ()
- Receiver **get_receiver** ()
- void **trace_scene** (int num_rays)
- void **trace_vect** (Ray3D &test_ray, double &hitDistance, Vector3D &hitNormal, Point3D &hitPoint)
- void **update** (double angle)
- double **getDistancePower** (const double frequency, const double power, const double distance) const
- double getDoppler (Ray3D &test_ray, Vector3D &hitNormal, Point3D &hitPoint, double RPM) const

2.8.1 Member Function Documentation

2.8.1.1 getDoppler()

```
double Scene::getDoppler (
            Ray3D & test_ray,
            Vector3D & hitNormal,
            Point3D & hitPoint,
            double RPM ) const  [inline]
```

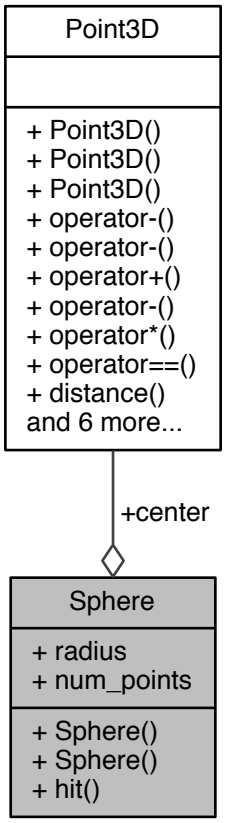
!!!! doppler does not get affected by the normal, need to check this. not sure if this is correct.

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

- Raytracer3D/Raytracer3D/Scene.hpp
- Raytracer3D/Raytracer3D/Scene.cpp

2.9 Sphere Class Reference

Collaboration diagram for Sphere:



Public Member Functions

- **Sphere** (const double radius, const Point3D &center)
- bool **hit** (const Ray3D &ray, double &hitDistance, Vector3D &hitNormal, Point3D &hitPoint) const

Public Attributes

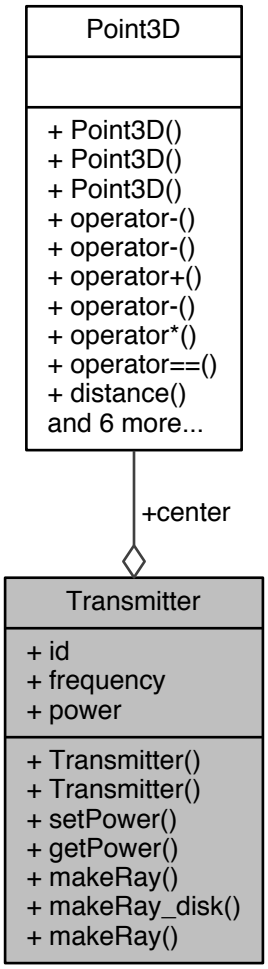
- double **radius**
- Point3D **center**
- int **num_points**

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

- Raytracer3D/Raytracer3D/Sphere.hpp
- Raytracer3D/Raytracer3D/Sphere.cpp

2.10 Transmitter Class Reference

Collaboration diagram for Transmitter:



Public Member Functions

- **Transmitter** (const int id, const double frequency, const double power, const Point3D &center, const double l)
- void **setPower** (const double power)
- double **getPower** () const
- Ray3D **makeRay** ()
- Ray3D **makeRay_disk** (const double height)
- Ray3D **makeRay** (const Vector3D &rayDirection)

Public Attributes

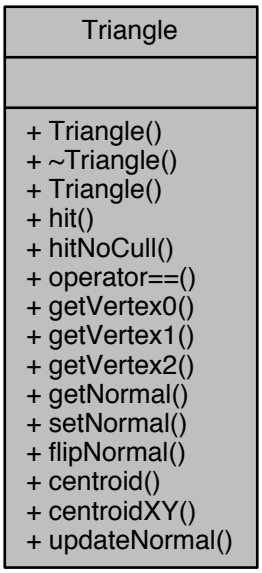
- int **id**
- double **frequency**
- Point3D **center**
- double **power**

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

- Raytracer3D/Raytracer3D/Transmitter.hpp
- Raytracer3D/Raytracer3D/Transmitter.cpp

2.11 Triangle Class Reference

Collaboration diagram for Triangle:



Public Member Functions

- **Triangle** (Point3D &v0, Point3D &v1, Point3D &v2)
- bool **hit** (const Ray3D &ray, double &hitDistance, Vector3D &hitNormal, Point3D &hitPoint) const
- bool **hitNoCull** (const Ray3D &ray, double &hitDistance, Vector3D &hitNormal, Point3D &hitPoint) const
- bool **operator==** (Triangle &Tri)
- Point3D **getVertex0** ()
- Point3D **getVertex1** ()
- Point3D **getVertex2** ()
- Vector3D **getNormal** () const

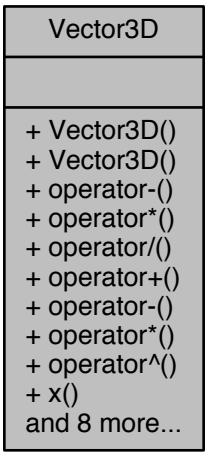
- void **setNormal** (const Vector3D &normal)
- void **flipNormal** ()
- Point3D **centroid** () const
- Point3D **centroidXY** () const
- void **updateNormal** ()

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

- Raytracer3D/Raytracer3D/Triangle.hpp
- Raytracer3D/Raytracer3D/Triangle.cpp

2.12 Vector3D Class Reference

Collaboration diagram for Vector3D:



Public Member Functions

- **Vector3D** (const double x, const double y, const double z)
- Vector3D **operator-** (void) const
- Vector3D **operator**∗ (const double a) const
- Vector3D **operator**/ (const double a) const
- Vector3D **operator+** (const Vector3D &v) const
- Vector3D **operator-** (const Vector3D &v) const
- double **operator**∗ (const Vector3D &b) const
- Vector3D **operator**^ (const Vector3D &v) const
- double **x** () const
- double **y** () const
- double **z** () const
- double **dotProduct** (const Vector3D &v) const

- Vector3D **crossProduct** (const Vector3D &v) const
- double **length** () const
- Vector3D **normalized** () const
- Vector3D **rotatedAboutZ** (const double angle) const
- bool **isNormal** () const

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

- Raytracer3D/Raytracer3D/Vector3d.hpp
- Raytracer3D/Raytracer3D/Vector3d.cpp