Inhaltsverzeichnis

By Alexander Brinkmann

1 Package de.grogra.lignum.sky

Still missing:

- FirmamentWithMask
- MineSet

1.1 Firmament.class

Includes:

- Firmament/Firmament.cc
- Firmament/include/Firmament.h

ToDo:

- Further Testing
- add method getIncAz
- improve getSunPosition()
- Test getDirection(int n)
- Discuss formula in line 733
- Put R_Epsilon into an interface

2 Package de.grogra.lignum.jadt

Reflects namespace cxxadt

Still missing:
Compare c++adt/

2.1 ParametricCurve.class

Includes:

- c++adt/src/ParametricCurve.cc
- c++adt/include/ParametricCurve.h

ToDo:

- Allow Comments in input file

2.2 PointL.class

PointL reflects the Point class used in Lignum

Extends:
javax.vecmath.Point3d

Includes:

- c++adt/src/Point.cc
- c++adt/include/Point.h

New functions will be added whenever they are needed.

Already added:

- Method "**distance(PointL)**": calculates the distance between two points.
- Method "**mul(double, PositionVector)**": mulitplies a scalar with a PositionVector and sets the values of the PointL to the result.
- Method "**subtract(PositionVector,PositionVector)**" : Subtracts the second PosV from the first one and sets the calling PosV to the result.
- Method "**subtract(PointL,PointL)**" : Subtracts the second Point from the first one and sets the calling Point to the result.

2.3 PositionVector.class

Extends:
de.grogra.ray2.radiosity.Vector3d

Includes:

- c++adt/src/PositionVector.cc
- c++adt/include/PositionVector.h

New functions will be added whenever they are needed.

Already added:

- Method "**rotate(PointL, PositionVector,double)**" which reflects the according function in cLignum
- Method "**mul(double, PositionVector)**": mulitplies a scalar with a PositionVector and sets the values of the calling PositionVector to the result.
- Method "**mul(double)**": mulitplies a scalar with the calling PositionVector and sets the values of the calling PositionVector to the result.
- Method "**subtract(PositionVector,PositionVector)**" : Subtracts the second PosV from the first one and sets the calling PosV to the result.

2.4 Shape.class

Only an interface.

Includes:

- c++adt/include/Shape.h

2.5 Mathsym.class

Interface

- c++adt/include/mathsym.h

Status:

- Const R_EPSILON defined.
- New functions will be added whenever they are needed.

2.6 Ellipse.class

Includes:

- c++adt/src/Ellipse.cc
- c++adt/include/Ellipse.h

ToDo:

- Further Testing

3 Package de.grogra.lignum.stlLignum

Reflects the stl-Lignum folder and the Lignum namespace.

Still missing: Compare stl-lignum/

3.1 BeamShading.class

Includes:

- stl-lignum/TreeSegment/BeamShading.cc

Status:

- The method CylinderBeamShading has been added but needs testing. The testing will probably take some time.
- The method EllipseBeamShading is not yet added because the class BroadLeaf is still missing.

3.2 Petiole.class

Includes:

- stl-lignum/include/BroadLeaf.h (partly)
- stl-lignum/TreeSegment/BroadLeaf.cc

Status:

- Needs further testing.

3.3 BroadLeafAttributes

Includes:

- stl-lignum/include/BroadLeaf.h (partly)
- stl-lignum/include/BroadLeafI.h (partly)

Status:

- Needs further testing.
- Instead of templates like in cLignum the class is only available for Shape Ellipse at the moment. Using generics would be quite complicated. If only a few different shapes are needed then it would be much more convenient to create a single class for each shape.

3.4 BroadLeaf

Includes:

- stl-lignum/include/BroadLeaf.h (partly)
- stl-lignum/include/BroadLeafI.h (partly)

Status:

- Needs further testing.
- Instead of templates like in cLignum the class is only available for Shape Ellipse at the moment. Using generics would be quite complicated. If only a few different shapes are needed then it would be much more convenient to create a single class for each shape.

3.5 LGMAD (Enum)

Includes:

- stl-lignum/include/LGMSymbols.h (partly)

Status:

- Implements the LGMAD enum type

4 de.grogra.lignum.stlVoxelspace

4.1 VoxelObject

Includes:

- stl-voxelspace/include/VoxelObject.h (partly)

Status:

- Abstract Class
- Needs further testing

4.2 CfCylinder

Includes:

- stl-voxelspace/include/VoxelObject.h (partly)

Status:

- Needs further testing

4.3 HwEllipse

Includes:

- stl-voxelspace/include/VoxelObject.h (partly)

Status:

- Needs further testing