graph-viewer-vr

Generated by Doxygen 1.8.15

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

Chapter 2

Class Index

2.1 Class List

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

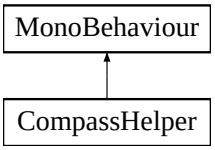
Chapter 3

Class Documentation

3.1 CompassHelper Class Reference

Displays compass below player.

Inheritance diagram for CompassHelper:



Public Attributes

- GameObject **player**

3.1.1 Detailed Description

Displays compass below player.

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

- CompassHelper.cs

3.2 Edge Class Reference

Public Member Functions

- **Edge** (string sourceId, string destinationId, float weight)

Public Attributes

- string **sourceId**
- string **destinationId**
- float **weight**
- GameObject **sourceNode**
- GameObject **destinationNode**

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

- Edge.cs

3.3 Graph Class Reference

Holds list of nodes and edges.

Public Member Functions

- **Graph** (List< Node > nodes, List< Edge > edges)

Public Attributes

- List< Edge > edges = new List<Edge>()
  List consisting of all edges.
- List< Node > nodes = new List<Node>()
  List consisting of all nodes.

3.3.1 Detailed Description

Holds list of nodes and edges.

3.3.2 Member Data Documentation

3.3.2.1 edges

```
List<Edge> Graph.edges = new List<Edge>()
```

List consisting of all edges.

3.3.2.2 nodes

```
List<Node> Graph.nodes = new List<Node>()
```

List consisting of all nodes.

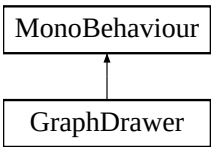
The documentation for this class was generated from the following file:

- Graph.cs

3.4 GraphDrawer Class Reference

Is attached to the GraphViewer component in the hierarchy and manages displaying the graph.

Inheritance diagram for GraphDrawer:



Public Member Functions

- void DrawNodes ()

  Iterates through all nodes and plots them into 3D world space.
- void DrawEdges ()

  Iterates through all edges and connects source and target nodes.

Public Attributes

- GameObject **ball**
- GameObject **lineGenerator**
- GameObject **nodeText**
- GameObject **player**
- string inputFile

  This variable can be changed within the Unity GUI. Keep in mind, that the extension has to be.xml and the extension.graphml is not supported.
- float CoordinateScale = 10f

  This constant is multiplied with the coordinates of the nodes.
- float LineScale

  This variable is calculated using the amount of nodes and scales the width of the edges.

3.4.1 Detailed Description

Is attached to the GraphViewer component in the hierarchy and manages displaying the graph.

3.4.2 Member Function Documentation

3.4.2.1 DrawEdges()

```
void GraphDrawer.DrawEdges ( )  [inline]
```

Iterates through all edges and connects source and target nodes.

3.4.2.2 DrawNodes()

```
void GraphDrawer.DrawNodes ( )  [inline]
```

Iterates through all nodes and plots them into 3D world space.

3.4.3 Member Data Documentation

3.4.3.1 CoordinateScale

```
float GraphDrawer.CoordinateScale = 10f
```

This constant is multiplied with the coordinates of the nodes.

3.4.3.2 inputFile

```
string GraphDrawer.inputFile
```

This variable can be changed within the Unity GUI. Keep in mind, that the extension has to be.xml and the extension.graphml is not supported.

3.4.3.3 LineScale

```
float GraphDrawer.LineScale
```

This variable is calculated using the amount of nodes and scales the width of the edges.

See definition in GraphDrawer.DrawEdges

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

- GraphDrawer.cs

3.5 Node Class Reference

Public Member Functions

- **Node** (string id, float size, float[ ] rgb, float[ ] xyz)

Public Attributes

- string id

  This id gets displayed as node label.
- float **size**
- float [ ] rgb

  Red, green and blue values in range [0, 1].
- float [ ] xyz

  Node coordinates in 3D world space. Are scaled to avoid
- TextMesh label

  Unity supported text renderer to display node id.

3.5.1 Member Data Documentation

3.5.1.1 id

```
string Node.id
```

This id gets displayed as node label.

See Node.label

3.5.1.2 label

```
TextMesh Node.label
```

Unity supported text renderer to display node id.

3.5.1.3 rgb

```
float [] Node.rgb
```

Red, green and blue values in range [0, 1].