

## Fast OpenGL Library

Generated by Doxygen 1.8.17



<b>1 WIP: FastOGLib - Fast OpenGL Library</b>	<b>1</b>
1.1 Instalation of Library	1
1.1.1 Linux/OSX	1
1.1.2 Windows	1
1.1.3 Developing the application	1
1.2 Contributors	1
<b>2 Namespace Index</b>	<b>3</b>
2.1 Namespace List	3
<b>3 Hierarchical Index</b>	<b>5</b>
3.1 Class Hierarchy	5
<b>4 Class Index</b>	<b>7</b>
4.1 Class List	7
<b>5 File Index</b>	<b>9</b>
5.1 File List	9
<b>6 Namespace Documentation</b>	<b>11</b>
6.1 AEG Namespace Reference	11
6.2 Global Namespace Reference	11
6.3 GPU Namespace Reference	11
6.4 INPUT Namespace Reference	11
6.4.1 Enumeration Type Documentation	12
6.4.1.1 KeyBoardKey	12
6.5 Lights Namespace Reference	13
6.6 OpenGLInstance Namespace Reference	13
6.6.1 Function Documentation	13
6.6.1.1 CreateAndCompileShaders()	13
6.6.1.2 CreateProgramAndLinkShaders()	14
6.6.1.3 InitWindow()	14
6.6.1.4 setClearColor()	14
6.7 OSDL Namespace Reference	14
6.8 Primitives Namespace Reference	14
6.8.1 Typedef Documentation	15
6.8.1.1 ColorC	15
6.8.1.2 ColorD	16
6.8.1.3 ColorF	16
6.8.1.4 ColorI	16
6.8.1.5 ColorL	16
6.8.1.6 ColorLL	16
6.8.1.7 ColorRGBAC	16
6.8.1.8 ColorRGBAD	16

6.8.1.9 ColorRGBAF	16
6.8.1.10 ColorRGBAI	17
6.8.1.11 ColorRGBAL	17
6.8.1.12 ColorRGBALL	17
6.8.1.13 ColorRGBAS	17
6.8.1.14 ColorRGBAU	17
6.8.1.15 ColorRGBAUL	17
6.8.1.16 ColorRGBAULL	17
6.8.1.17 ColorS	17
6.8.1.18 ColorU	18
6.8.1.19 ColorUL	18
6.8.1.20 ColorULL	18
6.8.1.21 PointC	18
6.8.1.22 PointD	18
6.8.1.23 PointF	18
6.8.1.24 PointI	18
6.8.1.25 PointL	18
6.8.1.26 PointLL	19
6.8.1.27 PointS	19
6.8.1.28 PointU	19
6.8.1.29 PointUL	19
6.8.1.30 PointULL	19
6.8.1.31 PrimitivePtr	19
6.8.1.32 Primitives	19
6.8.1.33 PrimitivesRef	19
6.8.2 Enumeration Type Documentation	19
6.8.2.1 ColorType	19
<b>7 Class Documentation</b>	<b>21</b>
7.1 INPUT::Adapter Class Reference	21
7.1.1 Constructor & Destructor Documentation	22
7.1.1.1 Adapter()	22
7.1.2 Member Function Documentation	23
7.1.2.1 isKeyHold()	23
7.1.2.2 isKeyPressed()	23
7.1.2.3 Notify()	23
7.1.3 Member Data Documentation	23
7.1.3.1 keyHold	23
7.1.3.2 keyPressed	23
7.2 INPUT::AdapterHandler Class Reference	24
7.2.1 Member Enumeration Documentation	26
7.2.1.1 KeyActionType	26

7.2.2 Constructor & Destructor Documentation	26
7.2.2.1 AdapterHandler() [1/2]	26
7.2.2.2 AdapterHandler() [2/2]	26
7.2.3 Member Function Documentation	26
7.2.3.1 getAction()	27
7.2.3.2 getInstance()	27
7.2.3.3 getKey()	27
7.2.3.4 onKeyUpdate()	27
7.2.3.5 operator=()	27
7.2.3.6 UpdateKey()	27
7.2.4 Member Data Documentation	27
7.2.4.1 Action	28
7.2.4.2 key	28
7.3 GPU::ArrayHandler Class Reference	28
7.3.1 Constructor & Destructor Documentation	29
7.3.1.1 ArrayHandler()	29
7.3.1.2 ~ArrayHandler()	29
7.3.2 Member Function Documentation	29
7.3.2.1 allocateBuffer()	29
7.3.2.2 areFreeBuffers()	30
7.3.2.3 areRepetitions()	30
7.3.2.4 getFirstAvailavleBuffer()	31
7.3.2.5 releaseBuffer()	31
7.3.3 Member Data Documentation	31
7.3.3.1 GPUSpace	32
7.4 Primitives::Color< T > Struct Template Reference	32
7.4.1 Member Function Documentation	33
7.4.1.1 operator[]()	33
7.4.2 Member Data Documentation	33
7.4.2.1 b	33
7.4.2.2 g	33
7.4.2.3 r	33
7.5 Primitives::ColorRGBA< T > Struct Template Reference	34
7.5.1 Member Function Documentation	34
7.5.1.1 operator[]()	34
7.5.2 Member Data Documentation	34
7.5.2.1 alpha	35
7.5.2.2 color	35
7.6 Controller Class Reference	35
7.6.1 Constructor & Destructor Documentation	36
7.6.1.1 Controller()	37
7.6.1.2 ~Controller()	37

7.7 Lights::DirectLight Class Reference	37
7.7.1 Constructor & Destructor Documentation	39
7.7.1.1 DirectLight()	39
7.7.1.2 ~DirectLight()	39
7.7.2 Member Function Documentation	39
7.7.2.1 getAlpha()	40
7.7.2.2 getB()	40
7.7.2.3 getConeAngle()	40
7.7.2.4 getConeLength()	40
7.7.2.5 getConeWidth()	40
7.7.2.6 getG()	40
7.7.2.7 getLumen()	40
7.7.2.8 getR()	40
7.7.2.9 getX()	41
7.7.2.10 getY()	41
7.7.2.11 Move()	41
7.7.2.12 setAlpha()	41
7.7.2.13 setColor()	41
7.7.2.14 setLumen()	41
7.7.2.15 setPosition()	41
7.7.3 Member Data Documentation	42
7.7.3.1 alpha	42
7.7.3.2 b	42
7.7.3.3 coneAngle	42
7.7.3.4 coneLength	42
7.7.3.5 coneWidth	42
7.7.3.6 g	42
7.7.3.7 lumen	42
7.7.3.8 r	43
7.7.3.9 x	43
7.7.3.10 y	43
7.8 OSDL::DualLink Class Reference	43
7.8.1 Constructor & Destructor Documentation	45
7.8.1.1 DualLink()	45
7.8.1.2 ~DualLink()	46
7.9 Lights::GlobalLight Class Reference	46
7.9.1 Constructor & Destructor Documentation	47
7.9.1.1 GlobalLight()	47
7.9.1.2 ~GlobalLight()	47
7.9.2 Member Function Documentation	47
7.9.2.1 getAlpha()	48
7.9.2.2 getB()	48

7.9.2.3 getG()	48
7.9.2.4 getLumen()	48
7.9.2.5 getR()	48
7.9.2.6 getX()	48
7.9.2.7 getY()	48
7.9.2.8 Move()	49
7.9.2.9 setAlpha()	49
7.9.2.10 setColor()	49
7.9.2.11 setLumen()	49
7.9.2.12 setPosition()	49
7.9.3 Member Data Documentation	49
7.9.3.1 alpha	49
7.9.3.2 b	50
7.9.3.3 g	50
7.9.3.4 lumen	50
7.9.3.5 r	50
7.9.3.6 x	50
7.9.3.7 y	50
7.10 GPU::GPU_Ref Class Reference	51
7.10.1 Constructor & Destructor Documentation	52
7.10.1.1 GPU_Ref()	52
7.10.2 Member Function Documentation	52
7.10.2.1 getEBO()	52
7.10.2.2 getVAO()	52
7.10.2.3 getVBO()	52
7.10.2.4 Release()	52
7.10.2.5 Take()	53
7.10.2.6 Taken()	53
7.10.3 Friends And Related Function Documentation	53
7.10.3.1 ArrayHandler	53
7.10.4 Member Data Documentation	53
7.10.4.1 EBO	53
7.10.4.2 taken	53
7.10.4.3 VAO	53
7.10.4.4 VBO	54
7.11 Lights::LightManager Class Reference	54
7.11.1 Constructor & Destructor Documentation	55
7.11.1.1 LightManager()	55
7.11.1.2 ~LightManager()	55
7.11.2 Member Function Documentation	55
7.11.2.1 addDirectLight() [1/2]	55
7.11.2.2 addDirectLight() [2/2]	55

7.11.2.3 addGlobalLight() [1/2]	56
7.11.2.4 addGlobalLight() [2/2]	56
7.11.2.5 addSphericalLight() [1/2]	56
7.11.2.6 addSphericalLight() [2/2]	56
7.11.2.7 draw()	56
7.11.2.8 drawDirectLights()	57
7.11.2.9 drawGlobalLights()	57
7.11.2.10 drawSphericalLights()	57
7.11.2.11 removeDirectLight()	57
7.11.2.12 removeGlobalLight()	57
7.11.2.13 removeSphericalLight()	57
7.11.3 Member Data Documentation	57
7.11.3.1 directLights	58
7.11.3.2 globalLights	58
7.11.3.3 sphericalLights	58
7.12 Primitives::Line Class Reference	58
7.12.1 Constructor & Destructor Documentation	61
7.12.1.1 Line() [1/2]	61
7.12.1.2 Line() [2/2]	61
7.12.2 Member Function Documentation	62
7.12.2.1 Display()	62
7.12.2.2 isNear()	62
7.12.2.3 isVisible()	62
7.12.2.4 operator=()	62
7.12.2.5 setTexture()	63
7.12.3 Member Data Documentation	63
7.12.3.1 rectangles	63
7.13 Model Class Reference	63
7.13.1 Constructor & Destructor Documentation	64
7.13.1.1 Model()	65
7.13.1.2 ~Model()	65
7.14 OSDL::Observer Class Reference	65
7.14.1 Constructor & Destructor Documentation	66
7.14.1.1 Observer()	66
7.14.1.2 ~Observer()	66
7.14.2 Member Function Documentation	66
7.14.2.1 Notify()	66
7.15 Primitives::Point2D< T > Struct Template Reference	67
7.15.1 Member Data Documentation	67
7.15.1.1 x	67
7.15.1.2 y	67
7.16 Primitive Class Reference	68



7.16.1 Member Enumeration Documentation	70
7.16.1.1 prim_type	70
7.16.2 Constructor & Destructor Documentation	70
7.16.2.1 Primitive() [1/3]	70
7.16.2.2 Primitive() [2/3]	70
7.16.2.3 Primitive() [3/3]	71
7.16.2.4 ~Primitive()	71
7.16.3 Member Function Documentation	71
7.16.3.1 cos()	71
7.16.3.2 Display()	71
7.16.3.3 isNear()	71
7.16.3.4 isVisible()	71
7.16.3.5 operator=()	72
7.16.3.6 sin()	72
7.16.4 Member Data Documentation	72
7.16.4.1 TYPE	72
7.17 Primitives::PrimitivesManager Class Reference	72
7.17.1 Constructor & Destructor Documentation	73
7.17.1.1 PrimitivesManager()	74
7.17.1.2 ~PrimitivesManager()	74
7.17.2 Member Function Documentation	74
7.17.2.1 addPrimitive()	74
7.17.2.2 DisplayAll()	74
7.17.2.3 getPrimitives()	74
7.17.2.4 getPrimitivesNear()	74
7.17.2.5 removeAllPrimitives()	74
7.17.2.6 removePrimitive() [1/2]	75
7.17.2.7 removePrimitive() [2/2]	75
7.17.3 Member Data Documentation	75
7.17.3.1 primitives	75
7.18 Primitives::Rectanagle Class Reference	75
7.18.1 Member Typedef Documentation	79
7.18.1.1 colors_array	79
7.18.1.2 verticies_array	79
7.18.2 Constructor & Destructor Documentation	79
7.18.2.1 Rectanagle() [1/2]	79
7.18.2.2 Rectanagle() [2/2]	79
7.18.2.3 ~Rectanagle()	79
7.18.3 Member Function Documentation	79
7.18.3.1 calculateMatrixes()	80
7.18.3.2 calculateVerticesArr()	80
7.18.3.3 Display()	80

7.18.3.4 getRotation()	80
7.18.3.5 getX()	80
7.18.3.6 getY()	80
7.18.3.7 isNear()	80
7.18.3.8 isVisible()	81
7.18.3.9 Move()	81
7.18.3.10 operator=()	81
7.18.3.11 rotate()	82
7.18.3.12 rotateVertices()	82
7.18.3.13 setAlpha()	82
7.18.3.14 setB()	82
7.18.3.15 setColor()	83
7.18.3.16 setG()	83
7.18.3.17 setHeight()	83
7.18.3.18 setPosition()	83
7.18.3.19 setR()	84
7.18.3.20 setRotation()	84
7.18.3.21 setTexture()	84
7.18.3.22 SetVertexColor()	85
7.18.3.23 setWidth()	85
7.18.4 Member Data Documentation	85
7.18.4.1 angle	85
7.18.4.2 color_arr	85
7.18.4.3 height	85
7.18.4.4 inc	85
7.18.4.5 verticies_arr	86
7.18.4.6 width	86
7.18.4.7 window	86
7.18.4.8 x	86
7.18.4.9 y	86
7.19 Lights::SphericalLight Class Reference	87
7.19.1 Constructor & Destructor Documentation	88
7.19.1.1 SphericalLight()	88
7.19.1.2 ~SphericalLight()	88
7.19.2 Member Function Documentation	88
7.19.2.1 getAlpha()	88
7.19.2.2 getB()	89
7.19.2.3 getG()	89
7.19.2.4 getLumen()	89
7.19.2.5 getR()	89
7.19.2.6 getX()	89
7.19.2.7 getY()	89

7.19.2.8 Move()	89
7.19.2.9 setAlpha()	90
7.19.2.10 setColor()	90
7.19.2.11 setLumen()	90
7.19.2.12 setPosition()	90
7.19.3 Member Data Documentation	90
7.19.3.1 alpha	90
7.19.3.2 b	90
7.19.3.3 g	91
7.19.3.4 lumen	91
7.19.3.5 r	91
7.19.3.6 x	91
7.19.3.7 y	91
7.20 Primitives::Square Class Reference	92
7.20.1 Constructor & Destructor Documentation	94
7.20.1.1 Square() [1/2]	94
7.20.1.2 Square() [2/2]	94
7.20.1.3 ~Square()	94
7.20.2 Member Function Documentation	94
7.20.2.1 operator=()	94
7.20.3 Member Data Documentation	94
7.20.3.1 size	95
7.21 OSDL::Subject Class Reference	95
7.21.1 Member Function Documentation	96
7.21.1.1 addObserver()	96
7.21.1.2 Notify()	97
7.21.1.3 NotifyObserver()	97
7.21.1.4 removeObserver()	97
7.21.2 Member Data Documentation	97
7.21.2.1 observers	97
7.22 AEG::Texture Class Reference	97
7.22.1 Constructor & Destructor Documentation	99
7.22.1.1 Texture()	99
7.22.2 Member Function Documentation	99
7.22.2.1 Clear()	99
7.22.2.2 getLocation()	99
7.22.2.3 getTextureID()	99
7.22.2.4 setTextureID()	99
7.22.2.5 setTextureLocation()	99
7.22.3 Friends And Related Function Documentation	100
7.22.3.1 Textures	100
7.22.4 Member Data Documentation	100

7.22.4.1 ID	100
7.22.4.2 location	100
7.23 AEG::Textures Class Reference	100
7.23.1 Constructor & Destructor Documentation	101
7.23.1.1 Textures()	102
7.23.1.2 ~Textures()	102
7.23.2 Member Function Documentation	102
7.23.2.1 allocateTextureSpace()	102
7.23.2.2 getTextureIndex()	102
7.23.2.3 loadTexture()	102
7.23.2.4 LoadTexture()	102
7.23.2.5 unloadTexture() [1/2]	103
7.23.2.6 unloadTexture() [2/2]	103
7.23.3 Member Data Documentation	103
7.23.3.1 TextureMap	103
7.24 VertexHandler Class Reference	103
7.24.1 Constructor & Destructor Documentation	106
7.24.1.1 VertexHandler() [1/2]	106
7.24.1.2 VertexHandler() [2/2]	106
7.24.1.3 ~VertexHandler()	106
7.24.2 Member Function Documentation	106
7.24.2.1 getEBO()	106
7.24.2.2 getTexture()	106
7.24.2.3 getVAO()	106
7.24.2.4 getVBO()	107
7.24.2.5 operator=()	107
7.24.2.6 setTextureData()	107
7.24.2.7 setVertexHandler()	107
7.24.3 Member Data Documentation	107
7.24.3.1 EBO	107
7.24.3.2 texture	107
7.24.3.3 VAO	108
7.24.3.4 VBO	108
7.25 View Class Reference	108
7.25.1 Constructor & Destructor Documentation	109
7.25.1.1 View()	109
7.25.1.2 ~View()	109
7.26 Global::WindowProperties Class Reference	110
7.26.1 Constructor & Destructor Documentation	111
7.26.1.1 WindowProperties() [1/2]	111
7.26.1.2 WindowProperties() [2/2]	111
7.26.1.3 ~WindowProperties()	111

7.26.2 Member Function Documentation	111
7.26.2.1 getAH()	112
7.26.2.2 getInstance()	112
7.26.2.3 getTextures()	112
7.26.2.4 getWindowHeight()	112
7.26.2.5 getWindowWidth()	112
7.26.2.6 operator=()	112
7.26.2.7 setWindowHeight()	112
7.26.2.8 setWindowWidth()	113
7.26.3 Member Data Documentation	113
7.26.3.1 windowHeight	113
7.26.3.2 windowWidth	113
<b>8 File Documentation</b>	<b>115</b>
8.1 Engine/Core/core/base/OpenGLStart.cpp File Reference	115
8.2 Engine/Core/core/base/OpenGLStart.hpp File Reference	116
8.3 Engine/Core/core/base/Point.hpp File Reference	117
8.4 Engine/Core/core/base/Primitive.cpp File Reference	118
8.5 Engine/Core/core/base/Primitive.hpp File Reference	119
8.5.1 Macro Definition Documentation	120
8.5.1.1 M_PI	120
8.6 Engine/Core/core/base/Texture.cpp File Reference	120
8.6.1 Macro Definition Documentation	121
8.6.1.1 STB_IMAGE_IMPLEMENTATION	121
8.7 Engine/Core/core/base/Texture.hpp File Reference	121
8.8 Engine/Core/core/base/vertexHandler.cpp File Reference	122
8.9 Engine/Core/core/base/vertexHandler.hpp File Reference	122
8.10 Engine/Core/core/InputAdapter.cpp File Reference	123
8.11 Engine/Core/core/InputAdapter.hpp File Reference	124
8.12 Engine/Core/core/OpenGLArrays/ArrayHandler.cpp File Reference	125
8.13 Engine/Core/core/OpenGLArrays/ArrayHandler.hpp File Reference	126
8.14 Engine/Core/core/primitives/Light/DirectLight.cpp File Reference	127
8.15 Engine/Core/core/primitives/Light/DirectLight.hpp File Reference	127
8.16 Engine/Core/core/primitives/Light/GlobalLight.cpp File Reference	128
8.17 Engine/Core/core/primitives/Light/GlobalLight.hpp File Reference	128
8.18 Engine/Core/core/primitives/Light/SphericalLight.cpp File Reference	129
8.19 Engine/Core/core/primitives/Light/SphericalLight.hpp File Reference	130
8.20 Engine/Core/core/primitives/Lights.cpp File Reference	130
8.21 Engine/Core/core/primitives/Lights.hpp File Reference	131
8.22 Engine/Core/core/primitives/Line.cpp File Reference	132
8.23 Engine/Core/core/primitives/Line.hpp File Reference	132
8.24 Engine/Core/core/primitives/PrimitivesManager.cpp File Reference	133

8.25 Engine/Core/core/primitives/PrimitivesManager.hpp File Reference . . . . .	134
8.26 Engine/Core/core/primitives/Rectangle.cpp File Reference . . . . .	135
8.27 Engine/Core/core/primitives/Rectangle.hpp File Reference . . . . .	135
8.28 Engine/Core/core/primitives/Square.cpp File Reference . . . . .	136
8.29 Engine/Core/core/primitives/Square.hpp File Reference . . . . .	137
8.30 Engine/Core/core/WindowProperties.cpp File Reference . . . . .	138
8.31 Engine/Core/core/WindowProperties.hpp File Reference . . . . .	138
8.32 Engine/Core/main.cpp File Reference . . . . .	139
8.32.1 Macro Definition Documentation . . . . .	140
8.32.1.1 STB_IMAGE_IMPLEMENTATION . . . . .	140
8.32.2 Function Documentation . . . . .	140
8.32.2.1 main() . . . . .	140
8.32.2.2 onKeyCallback() . . . . .	140
8.32.2.3 onKeyPress() . . . . .	141
8.32.2.4 readFile() . . . . .	141
8.32.3 Variable Documentation . . . . .	141
8.32.3.1 player . . . . .	141
8.33 Engine/Core/patterns/MVC/Controller.hpp File Reference . . . . .	141
8.34 Engine/Core/patterns/MVC/Model.hpp File Reference . . . . .	142
8.35 Engine/Core/patterns/MVC/View.hpp File Reference . . . . .	142
8.36 Engine/Core/patterns/ObserverSubject/DualLink.hpp File Reference . . . . .	143
8.37 Engine/Core/patterns/ObserverSubject/Observer.hpp File Reference . . . . .	145
8.38 Engine/Core/patterns/ObserverSubject/Subject.cpp File Reference . . . . .	146
8.39 Engine/Core/patterns/ObserverSubject/Subject.hpp File Reference . . . . .	146
8.40 README.md File Reference . . . . .	147
<b>Index</b>	<b>149</b>

# Chapter 1

## WIP: FastOGLib - Fast OpenGL Library

Fast and scalable library, able to create OpenGL programs on osx/linux/windows machines.

### 1.1 Instalation of Library

Actually does not provide plug in solution. Saying that so it's needed to download all source codes of Lib and dependent libs.

```
git stash
git pull
git submodule sync && git submodule update --init
```

#### 1.1.1 Linux/OSX

After checking out the desired version of library building is straight forward:

```
mkdir build
cd build
cmake ..
make
```

#### 1.1.2 Windows

Unfortunately on Windows the GNU compiler with make is required (MVC option is not tested - Feel free to test it, I will try to help my best). In order to build the app it's needed to select correct compiler:

```
mkdir build
git stash
git pull
git submodule sync && git submodule update --init
cd build
cmake .. -G "MinGW Makefiles"
mingw32-make.exe -j 10 -l 10
```

#### 1.1.3 Developing the application

Actually in order of development on this library the `main.cpp` file has to be modified. In future months I'll be adding more functionality, which can be monitored in issues, and after some time, I'll provide CMake style libraries.

### 1.2 Contributors

@mwawrzkow - Marcin Wawrzków - owner





## Chapter 2

# Namespace Index

### 2.1 Namespace List

Here is a list of all namespaces with brief descriptions:

<a href="#">AEG</a>	11
<a href="#">Global</a>	11
<a href="#">GPU</a>	11
<a href="#">INPUT</a>	11
<a href="#">Lights</a>	13
<a href="#">OpenGLInstance</a>	13
<a href="#">OSDL</a>	14
<a href="#">Primitives</a>	14



## Chapter 3

# Hierarchical Index

### 3.1 Class Hierarchy

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

GPU::ArrayHandler . . . . .	28
Primitives::Color< T > . . . . .	32
Primitives::ColorRGBA< T > . . . . .	34
Lights::DirectLight . . . . .	37
Lights::GlobalLight . . . . .	46
GPU::GPU_Ref . . . . .	51
Lights::LightManager . . . . .	54
OSDL::Observer . . . . .	65
INPUT::Adapter . . . . .	21
OSDL::DualLink . . . . .	43
Controller . . . . .	35
Model . . . . .	63
View . . . . .	108
Primitives::Point2D< T > . . . . .	67
Primitives::PrimitivesManager . . . . .	72
Lights::SphericalLight . . . . .	87
OSDL::Subject . . . . .	95
INPUT::AdapterHandler . . . . .	24
OSDL::DualLink . . . . .	43
AEG::Texture . . . . .	97
AEG::Textures . . . . .	100
VertexHandler . . . . .	103
Primitive . . . . .	68
Primitives::Line . . . . .	58
Primitives::Rectanagle . . . . .	75
Primitives::Square . . . . .	92
Global::WindowProperties . . . . .	110



## Chapter 4

# Class Index

### 4.1 Class List

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

INPUT::Adapter	21
INPUT::AdapterHandler	24
GPU::ArrayHandler	28
Primitives::Color< T >	32
Primitives::ColorRGBA< T >	34
Controller	35
Lights::DirectLight	37
OSDL::DualLink	43
Lights::GlobalLight	46
GPU::GPU_Ref	51
Lights::LightManager	54
Primitives::Line	58
Model	63
OSDL::Observer	65
Primitives::Point2D< T >	67
Primitive	68
Primitives::PrimitivesManager	72
Primitives::Rectangle	75
Lights::SphericalLight	87
Primitives::Square	92
OSDL::Subject	95
AEG::Texture	97
AEG::Textures	100
VertexHandler	103
View	108
Global::WindowProperties	110



## Chapter 5

# File Index

### 5.1 File List

Here is a list of all files with brief descriptions:

Engine/Core/main.cpp	139
Engine/Core/core/InputAdapter.cpp	123
Engine/Core/core/InputAdapter.hpp	124
Engine/Core/core/WindowProperties.cpp	138
Engine/Core/core/WindowProperties.hpp	138
Engine/Core/core/base/OpenGLStart.cpp	115
Engine/Core/core/base/OpenGLStart.hpp	116
Engine/Core/core/base/Point.hpp	117
Engine/Core/core/base/Primitive.cpp	118
Engine/Core/core/base/Primitive.hpp	119
Engine/Core/core/base/Texture.cpp	120
Engine/Core/core/base/Texture.hpp	121
Engine/Core/core/base/vertexHandler.cpp	122
Engine/Core/core/base/vertexHandler.hpp	122
Engine/Core/core/OpenGLArrays/ArrayHandler.cpp	125
Engine/Core/core/OpenGLArrays/ArrayHandler.hpp	126
Engine/Core/core/primitives/Lights.cpp	130
Engine/Core/core/primitives/Lights.hpp	131
Engine/Core/core/primitives/Line.cpp	132
Engine/Core/core/primitives/Line.hpp	132
Engine/Core/core/primitives/PrimitivesManager.cpp	133
Engine/Core/core/primitives/PrimitivesManager.hpp	134
Engine/Core/core/primitives/Rectanagle.cpp	135
Engine/Core/core/primitives/Rectanagle.hpp	135
Engine/Core/core/primitives/Square.cpp	136
Engine/Core/core/primitives/Square.hpp	137
Engine/Core/core/primitives/Light/DirectLight.cpp	127
Engine/Core/core/primitives/Light/DirectLight.hpp	127
Engine/Core/core/primitives/Light/GlobalLight.cpp	128
Engine/Core/core/primitives/Light/GlobalLight.hpp	128
Engine/Core/core/primitives/Light/SphericalLight.cpp	129
Engine/Core/core/primitives/Light/SphericalLight.hpp	130
Engine/Core/patterns/MVC/Controller.hpp	141
Engine/Core/patterns/MVC/Model.hpp	142
Engine/Core/patterns/MVC/View.hpp	142

Engine/Core/patterns/ObserverSubject/ <a href="#">DualLink.hpp</a> . . . . .	143
Engine/Core/patterns/ObserverSubject/ <a href="#">Observer.hpp</a> . . . . .	145
Engine/Core/patterns/ObserverSubject/ <a href="#">Subject.cpp</a> . . . . .	146
Engine/Core/patterns/ObserverSubject/ <a href="#">Subject.hpp</a> . . . . .	146



## Chapter 6

# Namespace Documentation

### 6.1 AEG Namespace Reference

#### Classes

- class [Texture](#)
- class [Textures](#)

### 6.2 Global Namespace Reference

#### Classes

- class [WindowProperties](#)

### 6.3 GPU Namespace Reference

#### Classes

- class [ArrayHandler](#)
- class [GPU\\_Ref](#)

### 6.4 INPUT Namespace Reference

#### Classes

- class [Adapter](#)
- class [AdapterHandler](#)

## Enumerations

- enum [KeyBoardKey](#) {  
[ERROR](#) = -1, [Key\\_1](#) = 0, [Key\\_2](#), [Key\\_3](#),  
[Key\\_4](#), [Key\\_5](#), [Key\\_6](#), [Key\\_7](#),  
[Key\\_8](#), [Key\\_9](#), [Key\\_0](#), [Q](#),  
[W](#), [E](#), [R](#), [T](#),  
[Y](#), [U](#), [I](#), [O](#),  
[P](#), [A](#), [S](#), [D](#),  
[F](#), [G](#), [H](#), [J](#),  
[K](#), [L](#), [Z](#), [X](#),  
[C](#), [V](#), [B](#), [N](#),  
[M](#), [Arrow\\_Up](#), [Arrow\\_Down](#), [Arrow\\_Left](#),  
[Arrow\\_Right](#) }

### 6.4.1 Enumeration Type Documentation

#### 6.4.1.1 KeyBoardKey

enum [INPUT::KeyBoardKey](#)

##### Enumerator

<a href="#">ERROR</a>	
<a href="#">Key_1</a>	
<a href="#">Key_2</a>	
<a href="#">Key_3</a>	
<a href="#">Key_4</a>	
<a href="#">Key_5</a>	
<a href="#">Key_6</a>	
<a href="#">Key_7</a>	
<a href="#">Key_8</a>	
<a href="#">Key_9</a>	
<a href="#">Key_0</a>	
<a href="#">Q</a>	
<a href="#">W</a>	
<a href="#">E</a>	
<a href="#">R</a>	
<a href="#">T</a>	
<a href="#">Y</a>	
<a href="#">U</a>	
<a href="#">I</a>	
<a href="#">O</a>	
<a href="#">P</a>	
<a href="#">A</a>	
<a href="#">S</a>	
<a href="#">D</a>	
<a href="#">F</a>	
<a href="#">G</a>	

## Enumerator

H	
J	
K	
L	
Z	
X	
C	
V	
B	
N	
M	
Arrow_Up	
Arrow_Down	
Arrow_Left	
Arrow_Right	

## 6.5 Lights Namespace Reference

### Classes

- class [DirectLight](#)
- class [GlobalLight](#)
- class [LightManager](#)
- class [SphericalLight](#)

## 6.6 OpenGLInstance Namespace Reference

### Functions

- GLFWwindow \* [InitWindow](#) (int width, int height, std::string name)
- void [setClearColor](#) (float R, float G, float B, float A)
- std::vector< GLuint > [CreateAndCompileShaders](#) (const char \*vertShaderCode, const char \*fragShaderCode↵  
Code)
- GLuint [CreateProgramAndLinkShaders](#) (const std::vector< GLuint > shaders)

### 6.6.1 Function Documentation

#### 6.6.1.1 CreateAndCompileShaders()

```
std::vector< GLuint > OpenGLInstance::CreateAndCompileShaders (
    const char * vertShaderCode,
    const char * fragShaderCode )
```

#### 6.6.1.2 CreateProgramAndLinkShaders()

```
GLuint OpenGLInstance::CreateProgramAndLinkShaders (
    const std::vector< GLuint > shaders )
```

#### 6.6.1.3 InitWindow()

```
GLFWwindow * OpenGLInstance::InitWindow (
    int width,
    int height,
    std::string name )
```

#### 6.6.1.4 setClearColor()

```
void OpenGLInstance::setClearColor (
    float R,
    float G,
    float B,
    float A )
```

## 6.7 OSDL Namespace Reference

### Classes

- class [DualLink](#)
- class [Observer](#)
- class [Subject](#)

## 6.8 Primitives Namespace Reference

### Classes

- struct [Color](#)
- struct [ColorRGBA](#)
- class [Line](#)
- struct [Point2D](#)
- class [PrimitivesManager](#)
- class [Rectanagle](#)
- class [Square](#)

## Typedefs

- typedef struct [Point2D](#)< float > [PointF](#)
- typedef [Point2D](#)< int > [PointI](#)
- typedef [Point2D](#)< double > [PointD](#)
- typedef [Point2D](#)< unsigned int > [PointU](#)
- typedef [Point2D](#)< unsigned long > [PointUL](#)
- typedef [Point2D](#)< unsigned long long > [PointULL](#)
- typedef [Point2D](#)< long > [PointL](#)
- typedef [Point2D](#)< long long > [PointLL](#)
- typedef [Point2D](#)< char > [PointC](#)
- typedef [Point2D](#)< short > [PointS](#)
- typedef struct [Color](#)< float > [ColorF](#)
- typedef [Color](#)< int > [ColorI](#)
- typedef [Color](#)< double > [ColorD](#)
- typedef [Color](#)< unsigned int > [ColorU](#)
- typedef [Color](#)< unsigned long > [ColorUL](#)
- typedef [Color](#)< unsigned long long > [ColorULL](#)
- typedef [Color](#)< long > [ColorL](#)
- typedef [Color](#)< long long > [ColorLL](#)
- typedef [Color](#)< char > [ColorC](#)
- typedef [Color](#)< short > [ColorS](#)
- typedef struct [ColorRGBA](#)< float > [ColorRGBAf](#)
- typedef [ColorRGBA](#)< int > [ColorRGBAI](#)
- typedef [ColorRGBA](#)< double > [ColorRGBAD](#)
- typedef [ColorRGBA](#)< unsigned int > [ColorRGBAUI](#)
- typedef [ColorRGBA](#)< unsigned long > [ColorRGBAUL](#)
- typedef [ColorRGBA](#)< unsigned long long > [ColorRGBAULL](#)
- typedef [ColorRGBA](#)< long > [ColorRGBAL](#)
- typedef [ColorRGBA](#)< long long > [ColorRGBALL](#)
- typedef [ColorRGBA](#)< char > [ColorRGBAC](#)
- typedef [ColorRGBA](#)< short > [ColorRGBAS](#)
- typedef [Primitive](#) \* [PrimitivePtr](#)
- typedef std::vector< [PrimitivePtr](#) > [Primitives](#)
- typedef [Primitives](#) & [PrimitivesRef](#)

## Enumerations

- enum [ColorType](#) {  
[RED](#), [GREEN](#), [BLUE](#), [ALPHA](#),  
[ERROR](#) }

### 6.8.1 Typedef Documentation

#### 6.8.1.1 ColorC

```
typedef Color<char> Primitives::ColorC
```

#### 6.8.1.2 ColorD

```
typedef Color<double> Primitives::ColorD
```

#### 6.8.1.3 ColorF

```
typedef struct Color< float > Primitives::ColorF
```

#### 6.8.1.4 ColorI

```
typedef Color<int> Primitives::ColorI
```

#### 6.8.1.5 ColorL

```
typedef Color<long> Primitives::ColorL
```

#### 6.8.1.6 ColorLL

```
typedef Color<long long> Primitives::ColorLL
```

#### 6.8.1.7 ColorRGBAC

```
typedef ColorRGBA<char> Primitives::ColorRGBAC
```

#### 6.8.1.8 ColorRGBAD

```
typedef ColorRGBA<double> Primitives::ColorRGBAD
```

#### 6.8.1.9 ColorRGBAf

```
typedef struct ColorRGBA< float > Primitives::ColorRGBAf
```

#### 6.8.1.10 ColorRGBA

```
typedef ColorRGBA<int> Primitives::ColorRGBA
```

#### 6.8.1.11 ColorRGBAL

```
typedef ColorRGBA<long> Primitives::ColorRGBAL
```

#### 6.8.1.12 ColorRGBALL

```
typedef ColorRGBA<long long> Primitives::ColorRGBALL
```

#### 6.8.1.13 ColorRGBAS

```
typedef ColorRGBA<short> Primitives::ColorRGBAS
```

#### 6.8.1.14 ColorRGBAU

```
typedef ColorRGBA<unsigned int> Primitives::ColorRGBAU
```

#### 6.8.1.15 ColorRGBAUL

```
typedef ColorRGBA<unsigned long> Primitives::ColorRGBAUL
```

#### 6.8.1.16 ColorRGBAULL

```
typedef ColorRGBA<unsigned long long> Primitives::ColorRGBAULL
```

#### 6.8.1.17 ColorS

```
typedef Color<short> Primitives::ColorS
```

#### 6.8.1.18 ColorU

```
typedef Color<unsigned int> Primitives::ColorU
```

#### 6.8.1.19 ColorUL

```
typedef Color<unsigned long> Primitives::ColorUL
```

#### 6.8.1.20 ColorULL

```
typedef Color<unsigned long long> Primitives::ColorULL
```

#### 6.8.1.21 PointC

```
typedef Point2D<char> Primitives::PointC
```

#### 6.8.1.22 PointD

```
typedef Point2D<double> Primitives::PointD
```

#### 6.8.1.23 PointF

```
typedef struct Point2D< float > Primitives::PointF
```

#### 6.8.1.24 PointI

```
typedef Point2D<int> Primitives::PointI
```

#### 6.8.1.25 PointL

```
typedef Point2D<long> Primitives::PointL
```



#### 6.8.1.26 PointLL

```
typedef Point2D<long long> Primitives::PointLL
```

#### 6.8.1.27 PointS

```
typedef Point2D<short> Primitives::PointS
```

#### 6.8.1.28 PointU

```
typedef Point2D<unsigned int> Primitives::PointU
```

#### 6.8.1.29 PointUL

```
typedef Point2D<unsigned long> Primitives::PointUL
```

#### 6.8.1.30 PointULL

```
typedef Point2D<unsigned long long> Primitives::PointULL
```

#### 6.8.1.31 PrimitivePtr

```
typedef Primitive* Primitives::PrimitivePtr
```

#### 6.8.1.32 Primitives

```
typedef std::vector<PrimitivePtr> Primitives::Primitives
```

#### 6.8.1.33 PrimitivesRef

```
typedef Primitives& Primitives::PrimitivesRef
```

### 6.8.2 Enumeration Type Documentation

#### 6.8.2.1 ColorType

```
enum Primitives::ColorType
```

**Enumerator**

RED	
GREEN	
BLUE	
ALPHA	
ERROR	

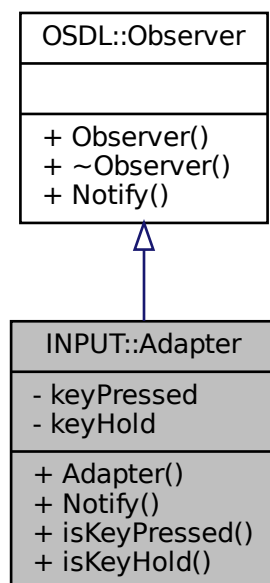
## Chapter 7

# Class Documentation

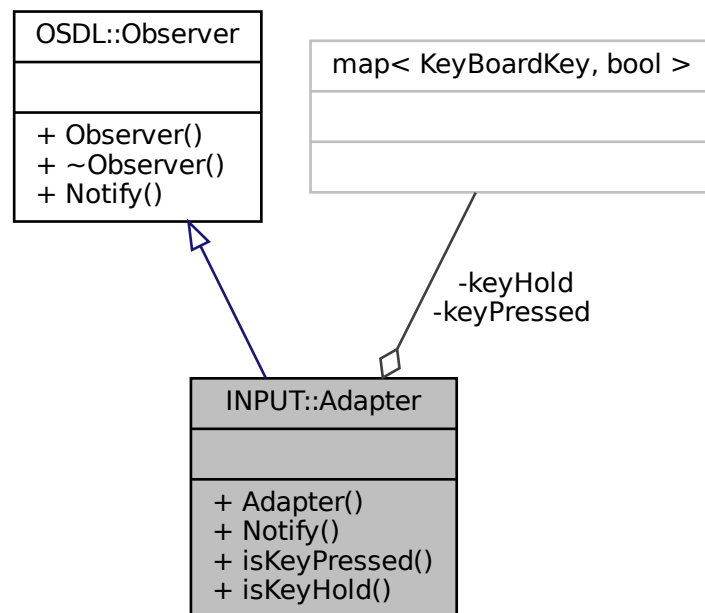
### 7.1 INPUT::Adapter Class Reference

```
#include <InputAdapter.hpp>
```

Inheritance diagram for INPUT::Adapter:



Collaboration diagram for INPUT::Adapter:



## Public Member Functions

- [Adapter](#) ()
- void [Notify](#) ([OSDL::Subject](#) \*)
- bool [isKeyPressed](#) ([KeyboardKey](#))
- bool [isKeyHold](#) ([KeyboardKey](#))

## Private Attributes

- std::map< [KeyboardKey](#), bool > [keyPressed](#)
- std::map< [KeyboardKey](#), bool > [keyHold](#)

## 7.1.1 Constructor & Destructor Documentation

### 7.1.1.1 Adapter()

```
INPUT::Adapter::Adapter ( )
```

## 7.1.2 Member Function Documentation

### 7.1.2.1 isKeyHold()

```
bool INPUT::Adapter::isKeyHold (
    KeyboardKey k )
```

### 7.1.2.2 isKeyPressed()

```
bool INPUT::Adapter::isKeyPressed (
    KeyboardKey k )
```

### 7.1.2.3 Notify()

```
void INPUT::Adapter::Notify (
    OSDL::Subject * ) [virtual]
```

Reimplemented from [OSDL::Observer](#).

## 7.1.3 Member Data Documentation

### 7.1.3.1 keyHold

```
std::map<KeyboardKey, bool> INPUT::Adapter::keyHold [private]
```

### 7.1.3.2 keyPressed

```
std::map<KeyboardKey, bool> INPUT::Adapter::keyPressed [private]
```

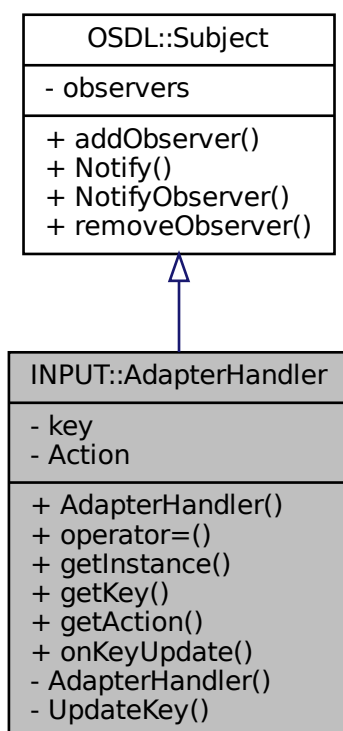
The documentation for this class was generated from the following files:

- [Engine/Core/core/InputAdapter.hpp](#)
- [Engine/Core/core/InputAdapter.cpp](#)

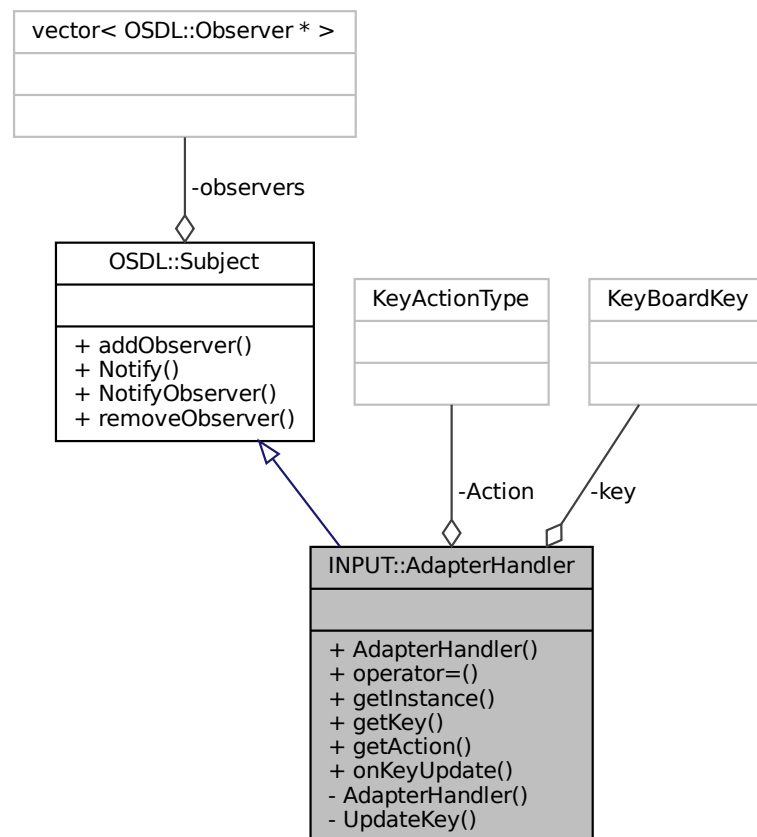
## 7.2 INPUT::AdapterHandler Class Reference

```
#include <InputAdapter.hpp>
```

Inheritance diagram for INPUT::AdapterHandler:



Collaboration diagram for INPUT::AdapterHandler:



## Public Types

- enum `KeyActionType` { `Hold`, `Press_down`, `Press_Up` }

## Public Member Functions

- `AdapterHandler (AdapterHandler &other)=delete`
- void `operator= (const AdapterHandler &)=delete`
- `AdapterHandler * getInstance ()`
- `KeyBoardKey getKey ()`
- `KeyActionType getAction ()`

## Static Public Member Functions

- static void `onKeyUpdate (GLFWwindow *window, int key, int status, int action, int mods)`

## Private Member Functions

- [AdapterHandler](#) ()
- void [UpdateKey](#) (int)

## Private Attributes

- [KeyBoardKey](#) key
- [KeyActionType](#) Action

## 7.2.1 Member Enumeration Documentation

### 7.2.1.1 KeyActionType

```
enum INPUT::AdapterHandler::KeyActionType
```

Enumerator

Hold	
Press_down	
Press_Up	

## 7.2.2 Constructor & Destructor Documentation

### 7.2.2.1 AdapterHandler() [1/2]

```
INPUT::AdapterHandler::AdapterHandler ( ) [private]
```

### 7.2.2.2 AdapterHandler() [2/2]

```
INPUT::AdapterHandler::AdapterHandler (
    AdapterHandler & other ) [delete]
```

## 7.2.3 Member Function Documentation



### 7.2.3.1 getAction()

`AdapterHandler::KeyActionType` INPUT::AdapterHandler::getAction ( )

### 7.2.3.2 getInstance()

`AdapterHandler *` INPUT::AdapterHandler::getInstance ( )

### 7.2.3.3 getKey()

`KeyBoardKey` INPUT::AdapterHandler::getKey ( )

### 7.2.3.4 onKeyUpdate()

```
void INPUT::AdapterHandler::onKeyUpdate (
    GLFWwindow * window,
    int key,
    int status,
    int action,
    int mods ) [static]
```

### 7.2.3.5 operator=()

```
void INPUT::AdapterHandler::operator= (
    const AdapterHandler & ) [delete]
```

### 7.2.3.6 UpdateKey()

```
void INPUT::AdapterHandler::UpdateKey (
    int Key ) [private]
```

## 7.2.4 Member Data Documentation

### 7.2.4.1 Action

`KeyActionType` INPUT::AdapterHandler::Action [private]

### 7.2.4.2 key

`KeyboardKey` INPUT::AdapterHandler::key [private]

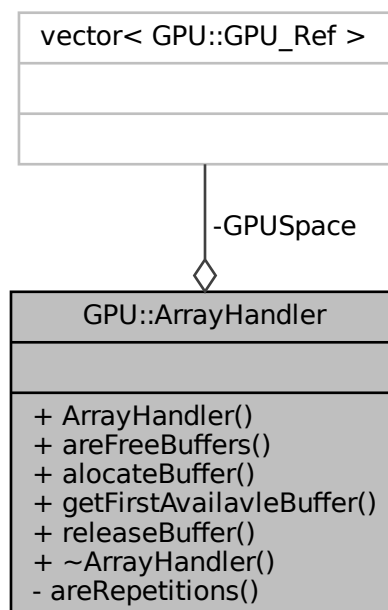
The documentation for this class was generated from the following files:

- Engine/Core/core/[InputAdapter.hpp](#)
- Engine/Core/core/[InputAdapter.cpp](#)

## 7.3 GPU::ArrayHandler Class Reference

```
#include <ArrayHandler.hpp>
```

Collaboration diagram for GPU::ArrayHandler:



## Public Member Functions

- [ArrayHandler](#) ()
- bool [areFreeBuffers](#) ()  
*Check if any buffers are available.*
- bool [allocateBuffer](#) (int)  
*Allocates GPU space for x buffers.*
- GPU\_Ref \* [getFirstAvailableBuffer](#) ()  
*Returns pointer to first Available Buffer.*
- void [releaseBuffer](#) (int)  
*Releases the buffer, if buffer is not taken does nothing.*
- [~ArrayHandler](#) ()

## Private Member Functions

- bool [areRepetitions](#) (unsigned int[], int)  
*check if are repetitions in Array*

## Private Attributes

- std::vector< [GPU\\_Ref](#) > [GPUSpace](#)

## 7.3.1 Constructor & Destructor Documentation

### 7.3.1.1 ArrayHandler()

```
GPU::ArrayHandler::ArrayHandler ( )
```

### 7.3.1.2 ~ArrayHandler()

```
GPU::ArrayHandler::~~ArrayHandler ( )
```

## 7.3.2 Member Function Documentation

### 7.3.2.1 allocateBuffer()

```
bool GPU::ArrayHandler::allocateBuffer (
    int size )
```

Allocates GPU space for x buffers.

**Parameters**

<i>int</i>	size amount of buffers
------------	------------------------

**Note****Return values**

<i>bool</i>	if buffers were created
-------------	-------------------------

**7.3.2.2 areFreeBuffers()**

```
bool GPU::ArrayHandler::areFreeBuffers ( )
```

Check if any buffers are available.

**Note****Return values**

<i>bool</i>	are any buffers free
-------------	----------------------

**7.3.2.3 areRepetitions()**

```
bool GPU::ArrayHandler::areRepetitions (
    unsigned int array[],
    int size ) [private]
```

check if are repetitions in Array

**Note****Parameters**

<i>int[]</i>	array
<i>int</i>	size of array

## Return values

<i>true</i>	If they're repetition
<i>false</i>	if there are no repetitions

**7.3.2.4 getFirstAvailavleBuffer()**

```
GPU_Ref * GPU::ArrayHandler::getFirstAvailavleBuffer ( )
```

Returns pointer to first Available Buffer.

## Note

## Return values

<i>GPU_Ref</i>	Pointer to buffer, in not buffers available return nullptr
----------------	--

**7.3.2.5 releaseBuffer()**

```
void GPU::ArrayHandler::releaseBuffer (
    int idx )
```

Releases the buffer, if buffer is not taken does nothing.

## Note

## Exceptions

------	--

## Return values

<i>None</i>	
-------------	--

**7.3.3 Member Data Documentation**

### 7.3.3.1 GPUSpace

```
std::vector<GPU_Ref> GPU::ArrayHandler::GPUSpace [private]
```

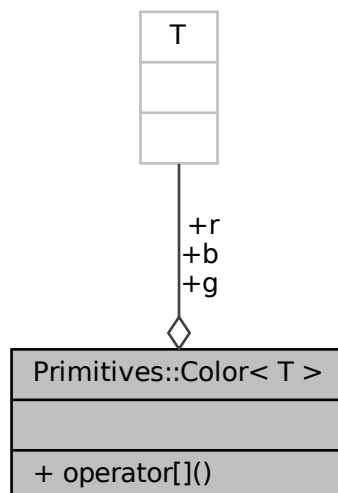
The documentation for this class was generated from the following files:

- Engine/Core/core/OpenGLArrays/[ArrayHandler.hpp](#)
- Engine/Core/core/OpenGLArrays/[ArrayHandler.cpp](#)

## 7.4 Primitives::Color< T > Struct Template Reference

```
#include <Point.hpp>
```

Collaboration diagram for Primitives::Color< T >:



### Public Member Functions

- `T & operator[]` ([ColorType](#) type)

### Public Attributes

- `T r = 0`
- `T g = 0`
- `T b = 0`

## 7.4.1 Member Function Documentation

### 7.4.1.1 operator[]()

```
template<typename T >
T& Primitives::Color< T >::operator[] (
    ColorType type ) [inline]
```

## 7.4.2 Member Data Documentation

### 7.4.2.1 b

```
template<typename T >
T Primitives::Color< T >::b = 0
```

### 7.4.2.2 g

```
template<typename T >
T Primitives::Color< T >::g = 0
```

### 7.4.2.3 r

```
template<typename T >
T Primitives::Color< T >::r = 0
```

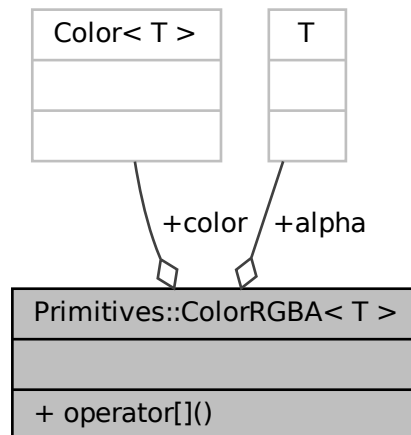
The documentation for this struct was generated from the following file:

- Engine/Core/core/base/[Point.hpp](#)

## 7.5 Primitives::ColorRGBA< T > Struct Template Reference

```
#include <Point.hpp>
```

Collaboration diagram for Primitives::ColorRGBA< T >:



### Public Member Functions

- `T & operator[]` (`ColorType` type)

### Public Attributes

- `Color< T > color`
- `T alpha = 0`

### 7.5.1 Member Function Documentation

#### 7.5.1.1 operator[]()

```
template<typename T >
T& Primitives::ColorRGBA< T >::operator[] (
    ColorType type ) [inline]
```

### 7.5.2 Member Data Documentation



## 7.5.2.1 alpha

```
template<typename T >
T Primitives::ColorRGBA< T >::alpha = 0
```

## 7.5.2.2 color

```
template<typename T >
Color<T> Primitives::ColorRGBA< T >::color
```

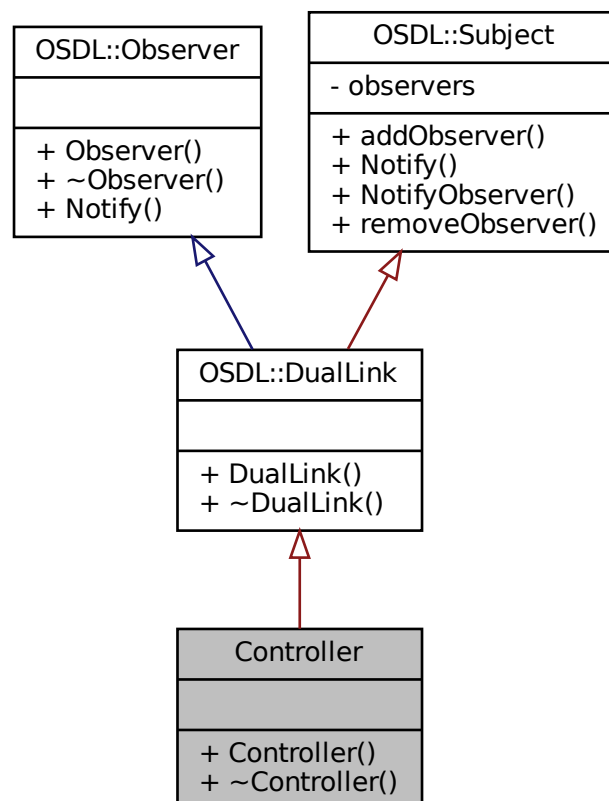
The documentation for this struct was generated from the following file:

- Engine/Core/core/base/[Point.hpp](#)

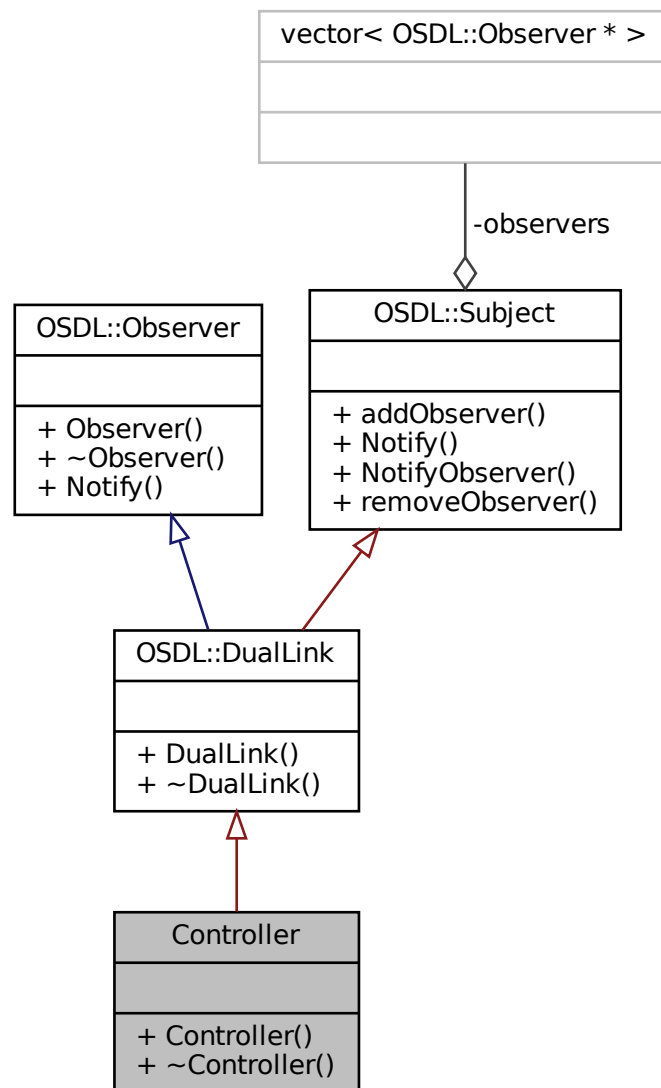
## 7.6 Controller Class Reference

```
#include <Controller.hpp>
```

Inheritance diagram for Controller:



Collaboration diagram for Controller:



## Public Member Functions

- [Controller\(\)](#)
- [~Controller\(\)](#)

## Additional Inherited Members

### 7.6.1 Constructor & Destructor Documentation

#### 7.6.1.1 Controller()

```
Controller::Controller ( )
```

#### 7.6.1.2 ~Controller()

```
Controller::~~Controller ( )
```

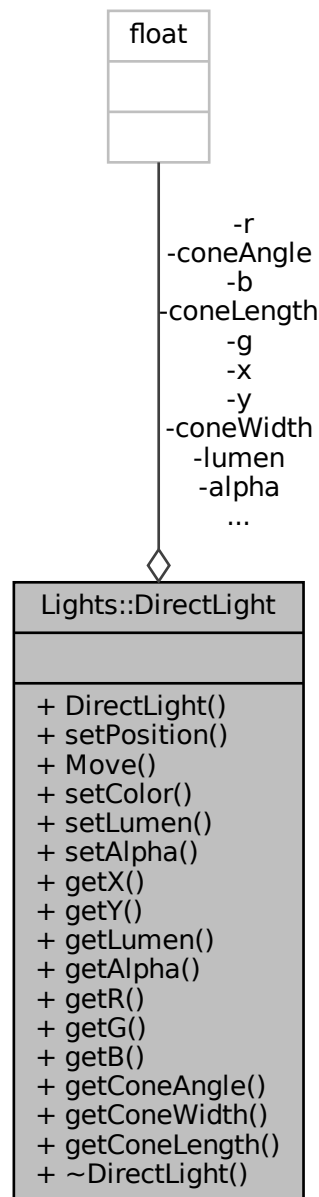
The documentation for this class was generated from the following file:

- Engine/Core/patterns/MVC/[Controller.hpp](#)

## 7.7 Lights::DirectLight Class Reference

```
#include <DirectLight.hpp>
```

Collaboration diagram for Lights::DirectLight:



## Public Member Functions

- [DirectLight](#) (float, float, float, float, float, float, float, float, float)
- void [setPosition](#) (float, float)
- void [Move](#) (float, float)
- void [setColor](#) (float, float, float)
- void [setLumen](#) (float)
- void [setAlpha](#) (float)
- float [getX](#) ()

- float [getY](#) ()
- float [getLumen](#) ()
- float [getAlpha](#) ()
- float [getR](#) ()
- float [getG](#) ()
- float [getB](#) ()
- float [getConeAngle](#) ()
- float [getConeWidth](#) ()
- float [getConeLength](#) ()
- [~DirectLight](#) ()

## Private Attributes

- float [x](#)
- float [y](#)
- float [lumen](#)
- float [alpha](#) = 1.0f
- float [r](#)
- float [g](#)
- float [b](#)
- float [coneAngle](#)
- float [coneWidth](#)
- float [coneLength](#)

## 7.7.1 Constructor & Destructor Documentation

### 7.7.1.1 DirectLight()

```
Lights::DirectLight::DirectLight (
    float x,
    float y,
    float lumen,
    float r,
    float g,
    float b,
    float coneAngle,
    float coneWidth,
    float coneLength )
```

### 7.7.1.2 ~DirectLight()

```
Lights::DirectLight::~~DirectLight ( )
```

## 7.7.2 Member Function Documentation

#### 7.7.2.1 getAlpha()

```
float Lights::DirectLight::getAlpha ( )
```

#### 7.7.2.2 getB()

```
float Lights::DirectLight::getB ( )
```

#### 7.7.2.3 getConeAngle()

```
float Lights::DirectLight::getConeAngle ( )
```

#### 7.7.2.4 getConeLength()

```
float Lights::DirectLight::getConeLength ( )
```

#### 7.7.2.5 getConeWidth()

```
float Lights::DirectLight::getConeWidth ( )
```

#### 7.7.2.6 getG()

```
float Lights::DirectLight::getG ( )
```

#### 7.7.2.7 getLumen()

```
float Lights::DirectLight::getLumen ( )
```

#### 7.7.2.8 getR()

```
float Lights::DirectLight::getR ( )
```

#### 7.7.2.9 getX()

```
float Lights::DirectLight::getX ( )
```

#### 7.7.2.10 getY()

```
float Lights::DirectLight::getY ( )
```

#### 7.7.2.11 Move()

```
void Lights::DirectLight::Move (
    float x,
    float y )
```

#### 7.7.2.12 setAlpha()

```
void Lights::DirectLight::setAlpha (
    float alpha )
```

#### 7.7.2.13 setColor()

```
void Lights::DirectLight::setColor (
    float r,
    float g,
    float b )
```

#### 7.7.2.14 setLumen()

```
void Lights::DirectLight::setLumen (
    float lumen )
```

#### 7.7.2.15 setPosition()

```
void Lights::DirectLight::setPosition (
    float x,
    float y )
```

### 7.7.3 Member Data Documentation

#### 7.7.3.1 alpha

```
float Lights::DirectLight::alpha = 1.0f [private]
```

#### 7.7.3.2 b

```
float Lights::DirectLight::b [private]
```

#### 7.7.3.3 coneAngle

```
float Lights::DirectLight::coneAngle [private]
```

#### 7.7.3.4 coneLength

```
float Lights::DirectLight::coneLength [private]
```

#### 7.7.3.5 coneWidth

```
float Lights::DirectLight::coneWidth [private]
```

#### 7.7.3.6 g

```
float Lights::DirectLight::g [private]
```

#### 7.7.3.7 lumen

```
float Lights::DirectLight::lumen [private]
```



#### 7.7.3.8 r

```
float Lights::DirectLight::r [private]
```

#### 7.7.3.9 x

```
float Lights::DirectLight::x [private]
```

#### 7.7.3.10 y

```
float Lights::DirectLight::y [private]
```

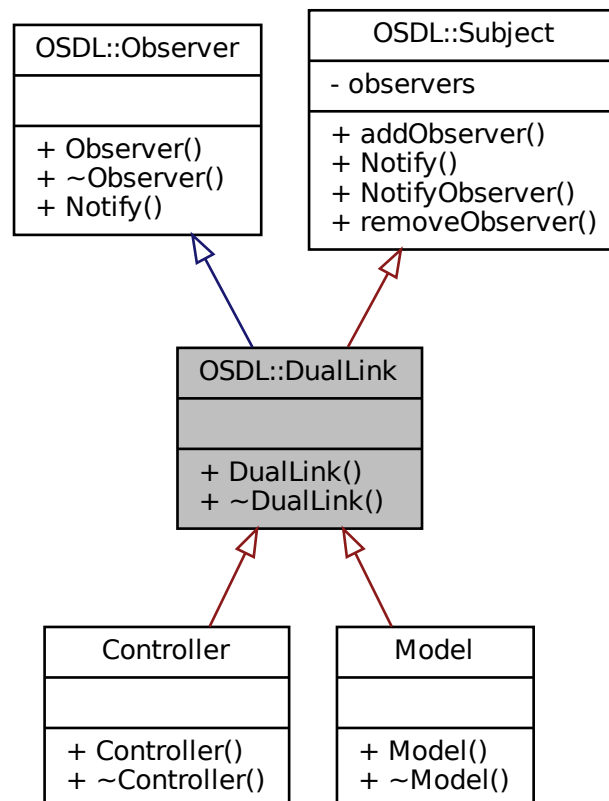
The documentation for this class was generated from the following files:

- Engine/Core/core/primitives/Light/[DirectLight.hpp](#)
- Engine/Core/core/primitives/Light/[DirectLight.cpp](#)

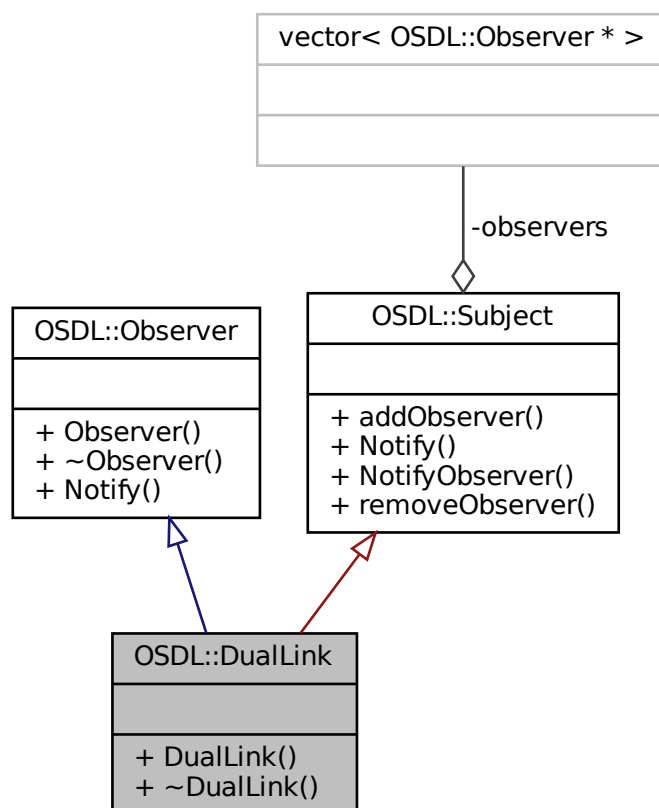
## 7.8 OSDL::DualLink Class Reference

```
#include <DualLink.hpp>
```

Inheritance diagram for OSDL::DualLink:



Collaboration diagram for OSDL::DualLink:



## Public Member Functions

- [DualLink\(\)](#)
- [~DualLink\(\)](#)

## Additional Inherited Members

### 7.8.1 Constructor & Destructor Documentation

#### 7.8.1.1 DualLink()

```
OSDL::DualLink::DualLink ( )
```

### 7.8.1.2 ~DualLink()

```
OSDL::DualLink::~~DualLink ( )
```

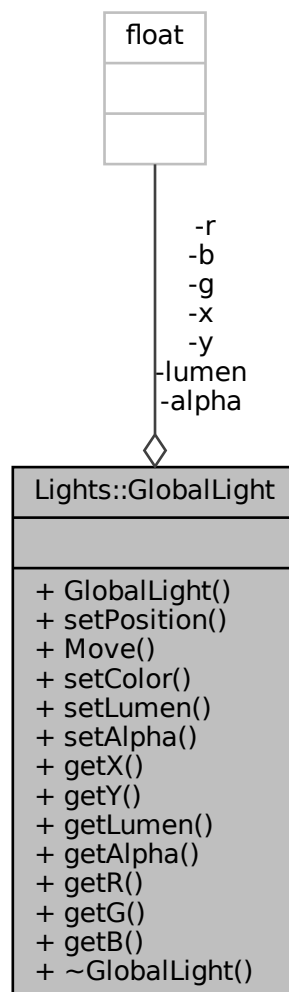
The documentation for this class was generated from the following file:

- [Engine/Core/patterns/ObserverSubject/DualLink.hpp](#)

## 7.9 Lights::GlobalLight Class Reference

```
#include <GlobalLight.hpp>
```

Collaboration diagram for Lights::GlobalLight:



## Public Member Functions

- [GlobalLight](#) (float, float, float, float, float, float)
- void [setPosition](#) (float, float)
- void [Move](#) (float, float)
- void [setColor](#) (float, float, float)
- void [setLumen](#) (float)
- void [setAlpha](#) (float)
- float [getX](#) ()
- float [getY](#) ()
- float [getLumen](#) ()
- float [getAlpha](#) ()
- float [getR](#) ()
- float [getG](#) ()
- float [getB](#) ()
- [~GlobalLight](#) ()

## Private Attributes

- float [x](#)
- float [y](#)
- float [lumen](#)
- float [alpha](#) = 1.0f
- float [r](#)
- float [g](#)
- float [b](#)

## 7.9.1 Constructor & Destructor Documentation

### 7.9.1.1 GlobalLight()

```
Lights::GlobalLight::GlobalLight (
    float x,
    float y,
    float lumen,
    float r,
    float g,
    float b )
```

### 7.9.1.2 ~GlobalLight()

```
Lights::GlobalLight::~~GlobalLight ( )
```

## 7.9.2 Member Function Documentation

#### 7.9.2.1 getAlpha()

```
float Lights::GlobalLight::getAlpha ( )
```

#### 7.9.2.2 getB()

```
float Lights::GlobalLight::getB ( )
```

#### 7.9.2.3 getG()

```
float Lights::GlobalLight::getG ( )
```

#### 7.9.2.4 getLumen()

```
float Lights::GlobalLight::getLumen ( )
```

#### 7.9.2.5 getR()

```
float Lights::GlobalLight::getR ( )
```

#### 7.9.2.6 getX()

```
float Lights::GlobalLight::getX ( )
```

#### 7.9.2.7 getY()

```
float Lights::GlobalLight::getY ( )
```

#### 7.9.2.8 Move()

```
void Lights::GlobalLight::Move (
    float x,
    float y )
```

#### 7.9.2.9 setAlpha()

```
void Lights::GlobalLight::setAlpha (
    float alpha )
```

#### 7.9.2.10 setColor()

```
void Lights::GlobalLight::setColor (
    float r,
    float g,
    float b )
```

#### 7.9.2.11 setLumen()

```
void Lights::GlobalLight::setLumen (
    float lumen )
```

#### 7.9.2.12 setPosition()

```
void Lights::GlobalLight::setPosition (
    float x,
    float y )
```

### 7.9.3 Member Data Documentation

#### 7.9.3.1 alpha

```
float Lights::GlobalLight::alpha = 1.0f [private]
```

### 7.9.3.2 b

```
float Lights::GlobalLight::b [private]
```

### 7.9.3.3 g

```
float Lights::GlobalLight::g [private]
```

### 7.9.3.4 lumen

```
float Lights::GlobalLight::lumen [private]
```

### 7.9.3.5 r

```
float Lights::GlobalLight::r [private]
```

### 7.9.3.6 x

```
float Lights::GlobalLight::x [private]
```

### 7.9.3.7 y

```
float Lights::GlobalLight::y [private]
```

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

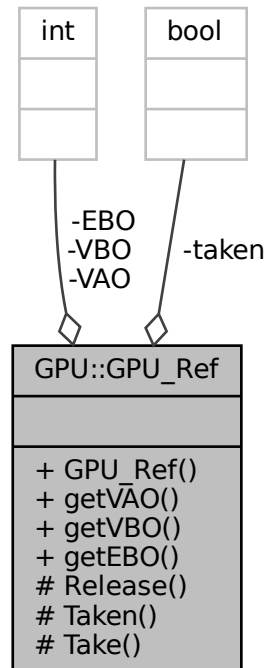
- Engine/Core/core/primitives/Light/[GlobalLight.hpp](#)
- Engine/Core/core/primitives/Light/[GlobalLight.cpp](#)



## 7.10 GPU::GPU\_Ref Class Reference

```
#include <ArrayHandler.hpp>
```

Collaboration diagram for GPU::GPU\_Ref:



### Public Member Functions

- GPU\_Ref (int, int, int)
- const unsigned int & getVAO ()
- const unsigned int & getVBO ()
- const unsigned int & getEBO ()

### Protected Member Functions

- void Release ()
- bool & Taken ()
- GPU\_Ref \* Take ()

### Private Attributes

- unsigned int VAO
- unsigned int VBO
- unsigned int EBO
- bool taken

## Friends

- class [ArrayHandler](#)

## 7.10.1 Constructor & Destructor Documentation

### 7.10.1.1 GPU\_Ref()

```
GPU::GPU_Ref::GPU_Ref (
    int  VAO,
    int  VBO,
    int  EBO )
```

## 7.10.2 Member Function Documentation

### 7.10.2.1 getEBO()

```
const unsigned int & GPU::GPU_Ref::getEBO ( )
```

### 7.10.2.2 getVAO()

```
const unsigned int & GPU::GPU_Ref::getVAO ( )
```

### 7.10.2.3 getVBO()

```
const unsigned int & GPU::GPU_Ref::getVBO ( )
```

### 7.10.2.4 Release()

```
void GPU::GPU_Ref::Release ( ) [protected]
```

### 7.10.2.5 Take()

```
GPU_Ref * GPU::GPU_Ref::Take ( ) [protected]
```

### 7.10.2.6 Taken()

```
bool & GPU::GPU_Ref::Taken ( ) [protected]
```

## 7.10.3 Friends And Related Function Documentation

### 7.10.3.1 ArrayHandler

```
friend class ArrayHandler [friend]
```

## 7.10.4 Member Data Documentation

### 7.10.4.1 EBO

```
unsigned int GPU::GPU_Ref::EBO [private]
```

### 7.10.4.2 taken

```
bool GPU::GPU_Ref::taken [private]
```

### 7.10.4.3 VAO

```
unsigned int GPU::GPU_Ref::VAO [private]
```

#### 7.10.4.4 VBO

```
unsigned int GPU::GPU_Ref::VBO [private]
```

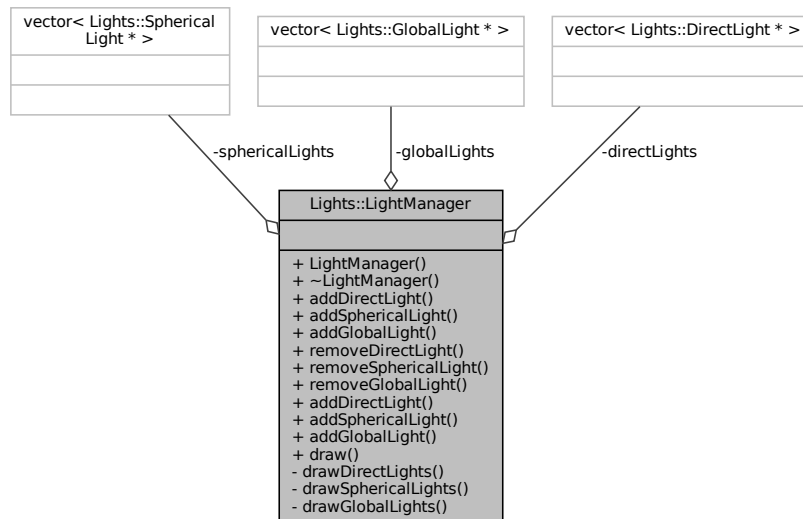
The documentation for this class was generated from the following files:

- Engine/Core/core/OpenGLArrays/[ArrayHandler.hpp](#)
- Engine/Core/core/OpenGLArrays/[ArrayHandler.cpp](#)

## 7.11 Lights::LightManager Class Reference

```
#include <Lights.hpp>
```

Collaboration diagram for Lights::LightManager:



### Public Member Functions

- [LightManager](#) ()
- [~LightManager](#) ()
- void [addDirectLight](#) ([DirectLight](#) \*)
- void [addSphericalLight](#) ([SphericalLight](#) \*)
- void [addGlobalLight](#) ([GlobalLight](#) \*)
- void [removeDirectLight](#) ([DirectLight](#) \*)
- void [removeSphericalLight](#) ([SphericalLight](#) \*)
- void [removeGlobalLight](#) ([GlobalLight](#) \*)
- void [addDirectLight](#) (float, float, float, float, float, float, float, float)
- void [addSphericalLight](#) (float, float, float, float, float, float, float)
- void [addGlobalLight](#) (float, float, float, float, float, float, float)
- void [draw](#) ([Primitives::PrimitivesManager](#) primitives)

## Private Member Functions

- void [drawDirectLights](#) ([Primitives::PrimitivesManager](#) primitives)
- void [drawSphericalLights](#) ([Primitives::PrimitivesManager](#) primitives)
- void [drawGlobalLights](#) ([Primitives::PrimitivesManager](#) primitives)

## Private Attributes

- std::vector< [DirectLight](#) \* > [directLights](#)
- std::vector< [SphericalLight](#) \* > [sphericalLights](#)
- std::vector< [GlobalLight](#) \* > [globalLights](#)

## 7.11.1 Constructor & Destructor Documentation

### 7.11.1.1 LightManager()

```
Lights::LightManager::LightManager ( )
```

### 7.11.1.2 ~LightManager()

```
Lights::LightManager::~~LightManager ( )
```

## 7.11.2 Member Function Documentation

### 7.11.2.1 addDirectLight() [1/2]

```
void Lights::LightManager::addDirectLight (
    DirectLight * directLight )
```

### 7.11.2.2 addDirectLight() [2/2]

```
void Lights::LightManager::addDirectLight (
    float x,
    float y,
    float z,
    float r,
    float g,
    float b,
    float a,
    float intensity,
    float angle )
```

### 7.11.2.3 addGlobalLight() [1/2]

```
void Lights::LightManager::addGlobalLight (
    float ,
    float ,
    float ,
    float ,
    float ,
    float ,
    float ,
    float )
```

### 7.11.2.4 addGlobalLight() [2/2]

```
void Lights::LightManager::addGlobalLight (
    GlobalLight * globalLight )
```

### 7.11.2.5 addSphericalLight() [1/2]

```
void Lights::LightManager::addSphericalLight (
    float x,
    float y,
    float z,
    float r,
    float g,
    float b,
    float a )
```

### 7.11.2.6 addSphericalLight() [2/2]

```
void Lights::LightManager::addSphericalLight (
    SphericalLight * sphericalLight )
```

### 7.11.2.7 draw()

```
void Lights::LightManager::draw (
    Primitives::PrimitivesManager primitives )
```

### 7.11.2.8 drawDirectLights()

```
void Lights::LightManager::drawDirectLights (
    Primitives::PrimitivesManager primitives ) [private]
```

### 7.11.2.9 drawGlobalLights()

```
void Lights::LightManager::drawGlobalLights (
    Primitives::PrimitivesManager primitives ) [private]
```

### 7.11.2.10 drawSphericalLights()

```
void Lights::LightManager::drawSphericalLights (
    Primitives::PrimitivesManager primitives ) [private]
```

### 7.11.2.11 removeDirectLight()

```
void Lights::LightManager::removeDirectLight (
    DirectLight * directLight )
```

### 7.11.2.12 removeGlobalLight()

```
void Lights::LightManager::removeGlobalLight (
    GlobalLight * globalLight )
```

### 7.11.2.13 removeSphericalLight()

```
void Lights::LightManager::removeSphericalLight (
    SphericalLight * sphericalLight )
```

## 7.11.3 Member Data Documentation

### 7.11.3.1 directLights

```
std::vector<DirectLight*> Lights::LightManager::directLights [private]
```

### 7.11.3.2 globalLights

```
std::vector<GlobalLight*> Lights::LightManager::globalLights [private]
```

### 7.11.3.3 sphericalLights

```
std::vector<SphericalLight*> Lights::LightManager::sphericalLights [private]
```

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

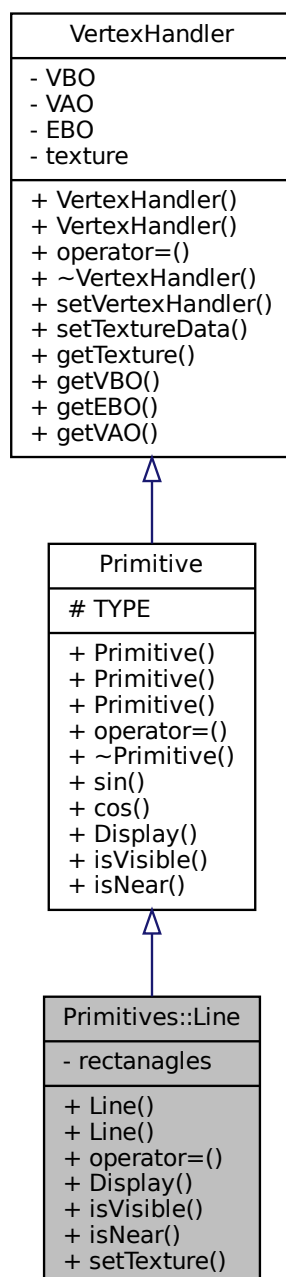
- Engine/Core/core/primitives/[Lights.hpp](#)
- Engine/Core/core/primitives/[Lights.cpp](#)

## 7.12 Primitives::Line Class Reference

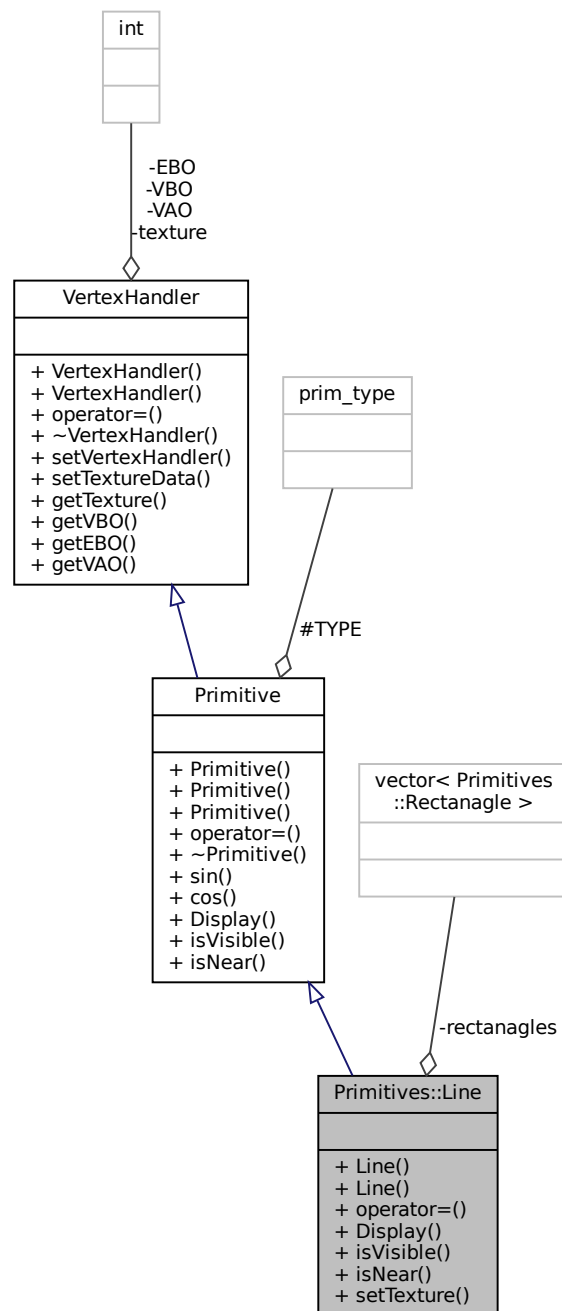
```
#include <Line.hpp>
```



Inheritance diagram for Primitives::Line:



Collaboration diagram for Primitives::Line:



## Public Member Functions

- [Line](#) (int x0, int y0, int x1, int y1, int r, int g, int b, float alpha)  
Construct a new [Line](#) object.
- [Line](#) (const [Line](#) &)
- [Line](#) operator= (const [Line](#) &)
- void [Display](#) () override

Display the [Line](#).

- bool [isVisible](#) () override  
*is the [Line](#) visible*
- bool [isNear](#) (float x, float y, float radius) override
- void [setTexture](#) (std::string)

## Private Attributes

- std::vector< [Rectangle](#) > [rectangles](#)

## Additional Inherited Members

### 7.12.1 Constructor & Destructor Documentation

#### 7.12.1.1 [Line\(\)](#) [1/2]

```
Primitives::Line::Line (
    int x0,
    int y0,
    int x1,
    int y1,
    int r,
    int g,
    int b,
    float alpha )
```

Construct a new [Line](#) object.

#### Parameters

<i>x0</i>	- x coordinate of first point
<i>y0</i>	- y coordinate of first point
<i>x1</i>	- x coordinate of second point
<i>y1</i>	- y coordinate of second point
<i>w</i>	- width of line
<i>r</i>	- red color component
<i>g</i>	- green color component
<i>b</i>	- blue color component
<i>alpha</i>	- alpha value of line

#### 7.12.1.2 [Line\(\)](#) [2/2]

```
Primitives::Line::Line (
    const Line & l )
```

## 7.12.2 Member Function Documentation

### 7.12.2.1 Display()

```
void Primitives::Line::Display ( ) [override], [virtual]
```

Display the [Line](#).

Implements [Primitive](#).

### 7.12.2.2 isNear()

```
bool Primitives::Line::isNear (
    float x,
    float y,
    float radius ) [override], [virtual]
```

Implements [Primitive](#).

### 7.12.2.3 isVisible()

```
bool Primitives::Line::isVisible ( ) [override], [virtual]
```

is the [Line](#) visible

#### Returns

true

false

Implements [Primitive](#).

### 7.12.2.4 operator=()

```
Line Primitives::Line::operator= (
    const Line & l )
```

### 7.12.2.5 setTexture()

```
void Primitives::Line::setTexture (
    std::string texture )
```

## 7.12.3 Member Data Documentation

### 7.12.3.1 rectangles

```
std::vector<Rectangle> Primitives::Line::rectangles [private]
```

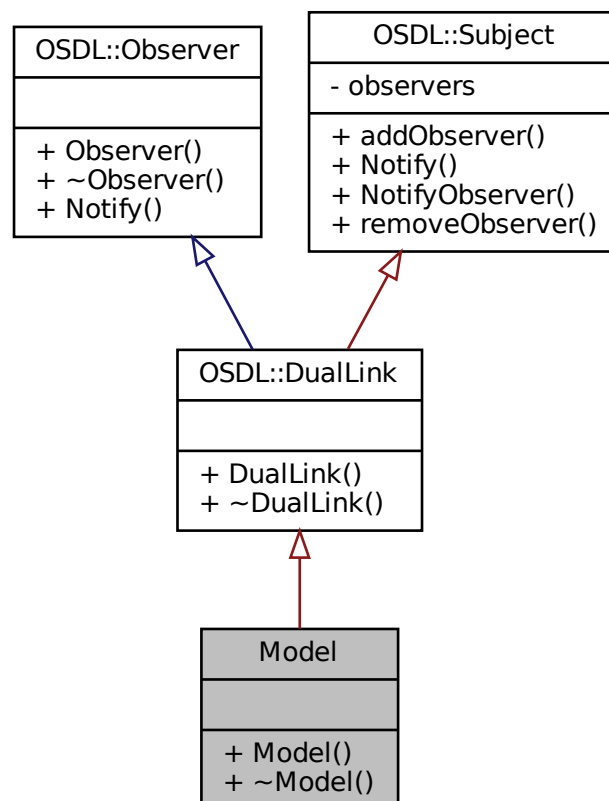
The documentation for this class was generated from the following files:

- Engine/Core/core/primitives/[Line.hpp](#)
- Engine/Core/core/primitives/[Line.cpp](#)

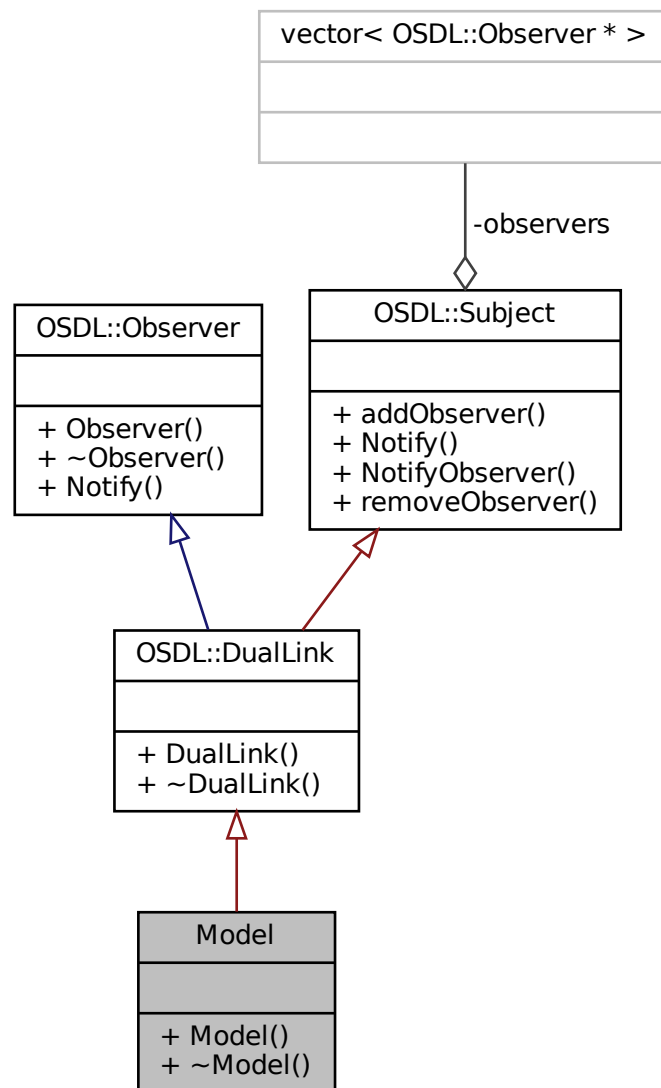
## 7.13 Model Class Reference

```
#include <Model.hpp>
```

Inheritance diagram for Model:



Collaboration diagram for Model:



## Public Member Functions

- [Model](#) ()
- [~Model](#) ()

## Additional Inherited Members

### 7.13.1 Constructor & Destructor Documentation

## 7.13.1.1 Model()

```
Model::Model ( )
```

## 7.13.1.2 ~Model()

```
Model::~Model ( )
```

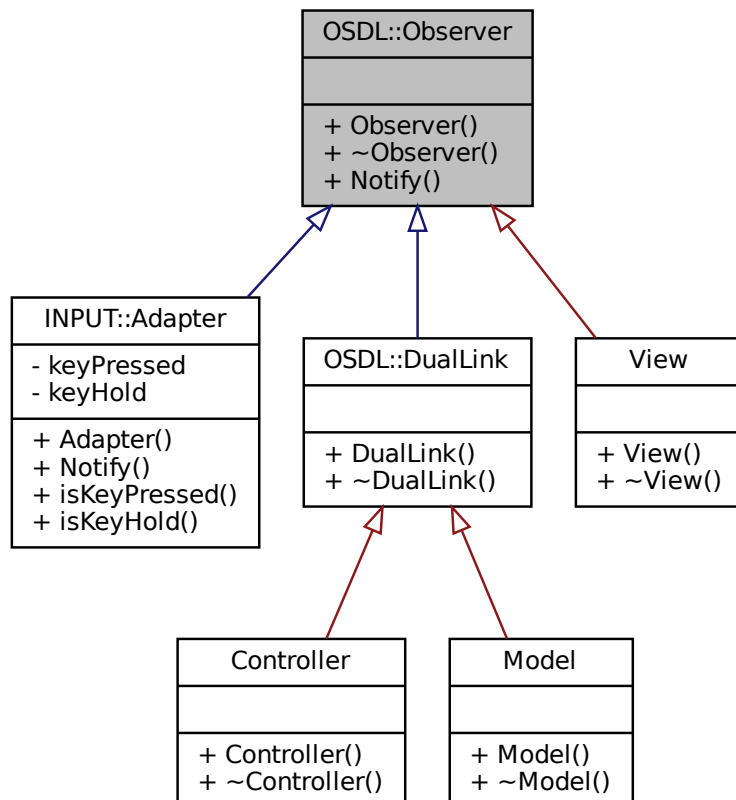
The documentation for this class was generated from the following file:

- Engine/Core/patterns/MVC/[Model.hpp](#)

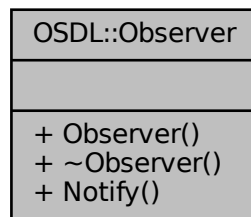
## 7.14 OSDL::Observer Class Reference

```
#include <Observer.hpp>
```

Inheritance diagram for OSDL::Observer:



Collaboration diagram for OSDL::Observer:



## Public Member Functions

- [Observer\(\)](#)
- [~Observer\(\)](#)
- virtual void [Notify](#) ([Subject](#) \*)

### 7.14.1 Constructor & Destructor Documentation

#### 7.14.1.1 Observer()

```
OSDL::Observer::Observer ( )
```

#### 7.14.1.2 ~Observer()

```
OSDL::Observer::~~Observer ( )
```

### 7.14.2 Member Function Documentation

#### 7.14.2.1 Notify()

```
virtual void OSDL::Observer::Notify (  
    Subject * ) [virtual]
```

Reimplemented in [INPUT::Adapter](#).

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

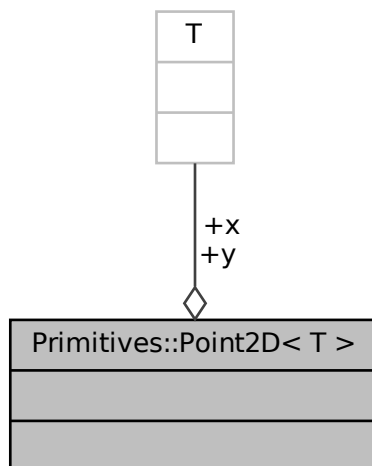
- Engine/Core/patterns/ObserverSubject/[Observer.hpp](#)



## 7.15 Primitives::Point2D< T > Struct Template Reference

```
#include <Point.hpp>
```

Collaboration diagram for Primitives::Point2D< T >:



### Public Attributes

- T `x` = 0
- T `y` = 0

### 7.15.1 Member Data Documentation

#### 7.15.1.1 `x`

```
template<typename T >
T Primitives::Point2D< T >::x = 0
```

#### 7.15.1.2 `y`

```
template<typename T >
T Primitives::Point2D< T >::y = 0
```

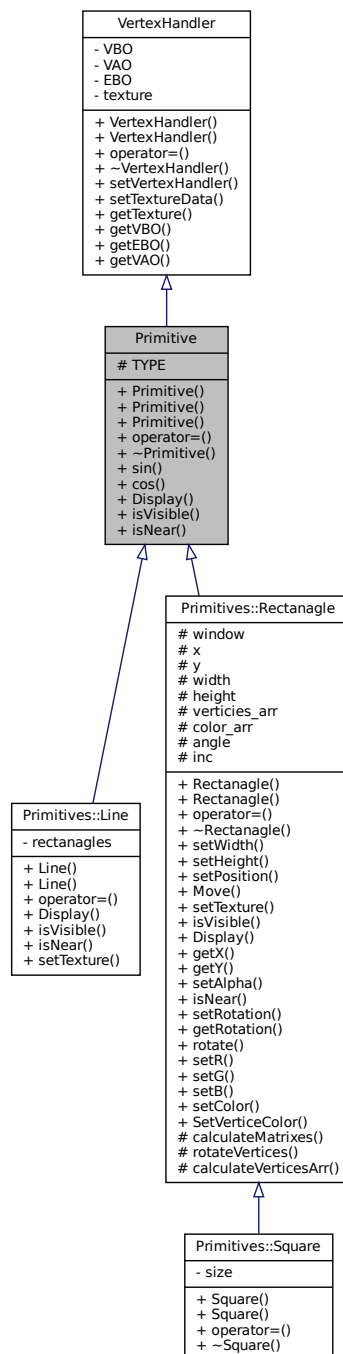
The documentation for this struct was generated from the following file:

- Engine/Core/core/base/[Point.hpp](#)

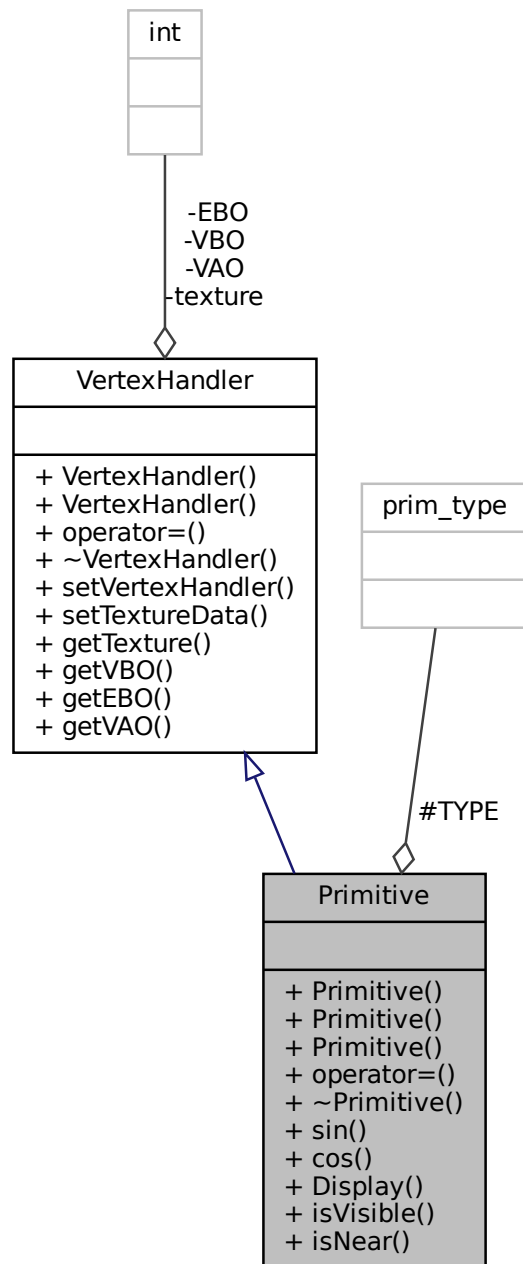
## 7.16 Primitive Class Reference

```
#include <Primitive.hpp>
```

Inheritance diagram for Primitive:



Collaboration diagram for Primitive:



## Public Member Functions

- [Primitive](#) ([prim\\_type](#))
- [Primitive](#) ([prim\\_type](#) type, bool isDefaultPrimitive)
- [Primitive](#) (const [Primitive](#) &p)
- [Primitive](#) & [operator=](#) (const [Primitive](#) &p)
- virtual [~Primitive](#) ()

- float [sin](#) (float r)
- float [cos](#) (float r)
- virtual void [Display](#) ()=0
- virtual bool [isVisible](#) ()=0
- virtual bool [isNear](#) (float, float, float)=0

## Protected Types

- enum [prim\\_type](#) { [RECTANAGLE](#), [TRIANGLE](#), [SQUARE](#), [LINE](#) }

## Protected Attributes

- enum [Primitive::prim\\_type](#) TYPE

## 7.16.1 Member Enumeration Documentation

### 7.16.1.1 [prim\\_type](#)

enum [Primitive::prim\\_type](#) [protected]

#### Enumerator

<a href="#">RECTANAGLE</a>	
<a href="#">TRIANGLE</a>	
<a href="#">SQUARE</a>	
<a href="#">LINE</a>	

## 7.16.2 Constructor & Destructor Documentation

### 7.16.2.1 [Primitive\(\)](#) [1/3]

```
Primitive::Primitive (
    prim\_type type )
```

### 7.16.2.2 [Primitive\(\)](#) [2/3]

```
Primitive::Primitive (
    prim\_type type,
    bool isDefaultPrimitive )
```

### 7.16.2.3 Primitive() [3/3]

```
Primitive::Primitive (
    const Primitive & p )
```

### 7.16.2.4 ~Primitive()

```
Primitive::~~Primitive ( ) [virtual]
```

## 7.16.3 Member Function Documentation

### 7.16.3.1 cos()

```
float Primitive::cos (
    float r )
```

### 7.16.3.2 Display()

```
virtual void Primitive::Display ( ) [pure virtual]
```

Implemented in [Primitives::Rectangle](#), and [Primitives::Line](#).

### 7.16.3.3 isNear()

```
virtual bool Primitive::isNear (
    float ,
    float ,
    float ) [pure virtual]
```

Implemented in [Primitives::Rectangle](#), and [Primitives::Line](#).

### 7.16.3.4 isVisible()

```
virtual bool Primitive::isVisible ( ) [pure virtual]
```

Implemented in [Primitives::Rectangle](#), and [Primitives::Line](#).

### 7.16.3.5 operator=()

```
Primitive & Primitive::operator= (
    const Primitive & p )
```

### 7.16.3.6 sin()

```
float Primitive::sin (
    float r )
```

## 7.16.4 Member Data Documentation

### 7.16.4.1 TYPE

```
enum Primitive::prim_type Primitive::TYPE [protected]
```

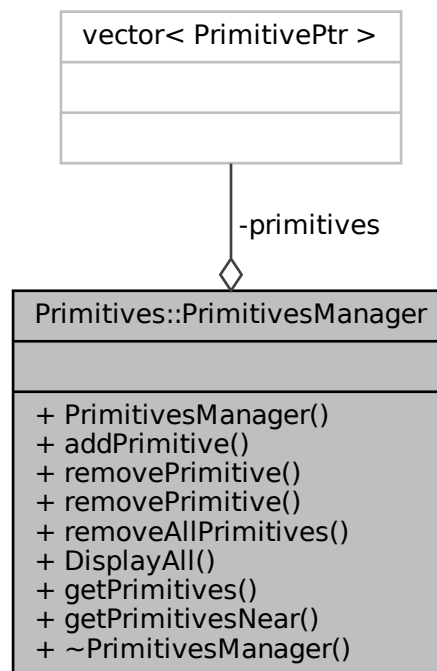
The documentation for this class was generated from the following files:

- Engine/Core/core/base/[Primitive.hpp](#)
- Engine/Core/core/base/[Primitive.cpp](#)

## 7.17 Primitives::PrimitivesManager Class Reference

```
#include <PrimitivesManager.hpp>
```

Collaboration diagram for Primitives::PrimitivesManager:



## Public Member Functions

- [PrimitivesManager](#) ()
- void [addPrimitive](#) ([Primitive](#) \*)
- void [removePrimitive](#) ([Primitive](#) \*)
- void [removePrimitive](#) (int)
- void [removeAllPrimitives](#) ()
- void [DisplayAll](#) ()
- [PrimitivesRef](#) [getPrimitives](#) ()
- [Primitives](#) [getPrimitivesNear](#) (float, float, float)
- [~PrimitivesManager](#) ()

## Private Attributes

- [Primitives](#) [primitives](#)

### 7.17.1 Constructor & Destructor Documentation

### 7.17.1.1 PrimitivesManager()

```
Primitives::PrimitivesManager::PrimitivesManager ( )
```

### 7.17.1.2 ~PrimitivesManager()

```
Primitives::PrimitivesManager::~~PrimitivesManager ( )
```

## 7.17.2 Member Function Documentation

### 7.17.2.1 addPrimitive()

```
void Primitives::PrimitivesManager::addPrimitive (
    Primitive * primitive )
```

### 7.17.2.2 DisplayAll()

```
void Primitives::PrimitivesManager::DisplayAll ( )
```

### 7.17.2.3 getPrimitives()

```
PrimitivesRef Primitives::PrimitivesManager::getPrimitives ( )
```

### 7.17.2.4 getPrimitivesNear()

```
Primitives Primitives::PrimitivesManager::getPrimitivesNear (
    float x,
    float y,
    float radius )
```

### 7.17.2.5 removeAllPrimitives()

```
void Primitives::PrimitivesManager::removeAllPrimitives ( )
```



#### 7.17.2.6 removePrimitive() [1/2]

```
void Primitives::PrimitivesManager::removePrimitive (
    int index )
```

#### 7.17.2.7 removePrimitive() [2/2]

```
void Primitives::PrimitivesManager::removePrimitive (
    Primitive * primitive )
```

### 7.17.3 Member Data Documentation

#### 7.17.3.1 primitives

```
Primitives Primitives::PrimitivesManager::primitives [private]
```

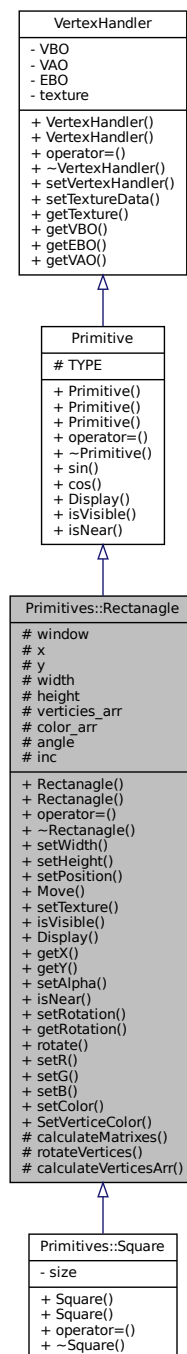
The documentation for this class was generated from the following files:

- Engine/Core/core/primitives/[PrimitivesManager.hpp](#)
- Engine/Core/core/primitives/[PrimitivesManager.cpp](#)

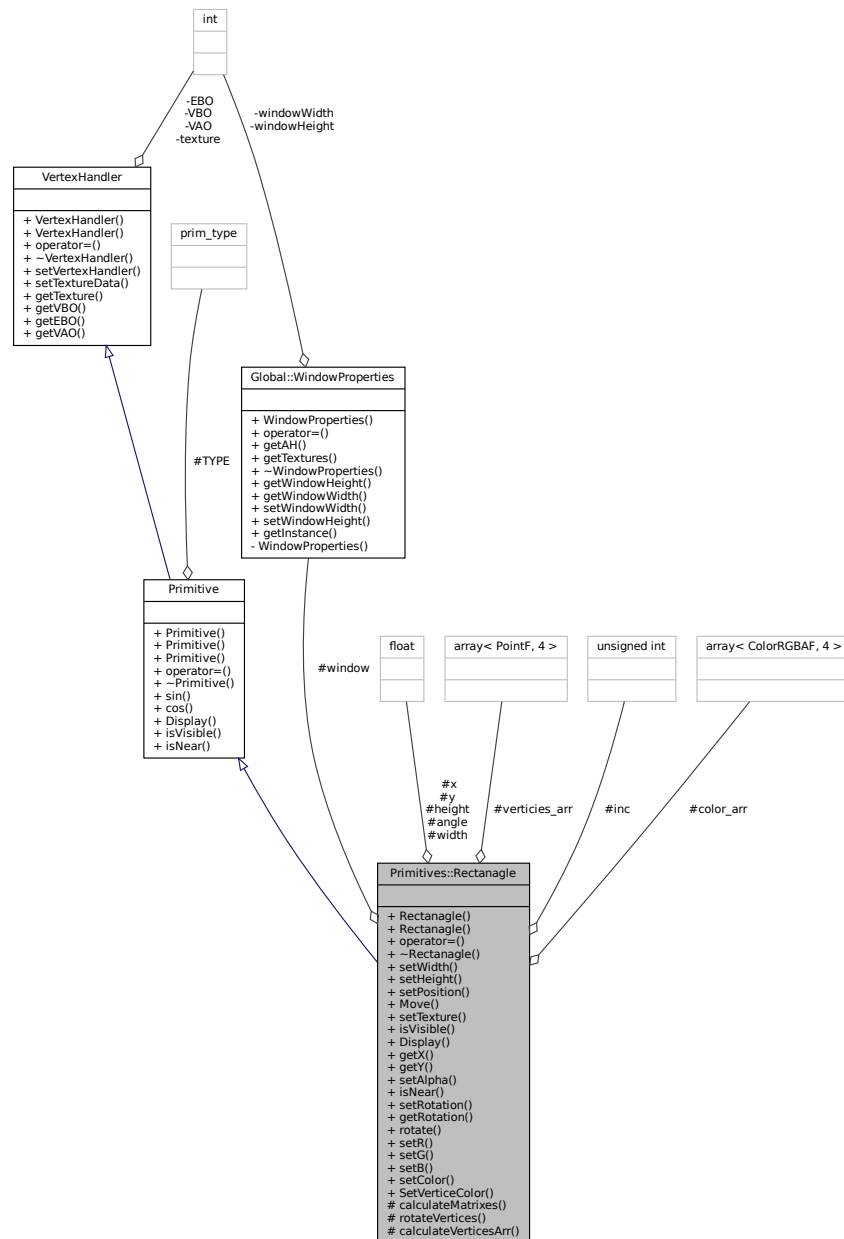
## 7.18 Primitives::Rectanagle Class Reference

```
#include <Rectanagle.hpp>
```

Inheritance diagram for Primitives::Rectangle:



Collaboration diagram for Primitives::Rectanagle:



## Public Member Functions

- [Rectanagle](#) (float, float, float, float, float=1.0f)
- [Rectanagle](#) (const [Rectanagle](#) &s)
- [Rectanagle](#) & `operator=` (const [Rectanagle](#) &s)
- virtual `~Rectanagle` ()
- void [setWidth](#) (float)  
Set the Width value.
- void [setHeight](#) (float)  
Set the Height value.

- void [setPosition](#) (float, float)  
*Set the Position object.*
- void [Move](#) (float, float)  
*move the square relative to the current position*
- void [setTexture](#) (std::string)  
*Set the Texture object.*
- bool [isVisible](#) () override  
*is Object on the screen*
- void [Display](#) () override  
*Draw the square.*
- float [getX](#) ()
- float [getY](#) ()
- void [setAlpha](#) (float)  
*Set the Alpha value.*
- bool [isNear](#) (float, float, float) override  
*is the Object int the radius*
- void [setRotation](#) (float)  
*Set the rotation of the square.*
- float [getRotation](#) ()
- void [rotate](#) (float)  
*rotate the square relative to the current rotation*
- void [setR](#) (float)  
*Set R color value.*
- void [setG](#) (float)  
*Set G color value.*
- void [setB](#) (float)
- void [setColor](#) (float, float, float, float=1.0f)  
*Set the Color value.*
- void [SetVertexColor](#) (int, float, [ColorType](#)=ERROR)

## Protected Types

- typedef std::array< [PointF](#), 4 > [verticies\\_array](#)
- typedef std::array< [ColorRGBAf](#), 4 > [colors\\_array](#)

## Protected Member Functions

- void [calculateMatrixes](#) ()
- [verticies\\_array](#) [rotateVertices](#) ()  
*Rotate the vertices of the square.*
- void [calculateVerticesArr](#) ()

## Protected Attributes

- [Global::WindowProperties](#) \* [window](#)
- float [x](#)
- float [y](#)
- float [width](#)
- float [height](#)
- [verticies\\_array](#) [verticies\\_arr](#) = {}
- [colors\\_array](#) [color\\_arr](#) = {}
- float [angle](#) = 0.0f
- unsigned int [inc](#) [6] = {0, 1, 2, 0, 2, 3}

## 7.18.1 Member Typedef Documentation

### 7.18.1.1 colors\_array

```
typedef std::array<ColorRGBAf, 4> Primitives::Rectanagle::colors_array [protected]
```

### 7.18.1.2 verticies\_array

```
typedef std::array<PointF, 4> Primitives::Rectanagle::verticies_array [protected]
```

## 7.18.2 Constructor & Destructor Documentation

### 7.18.2.1 Rectanagle() [1/2]

```
Primitives::Rectanagle::Rectanagle (
    float x,
    float y,
    float width,
    float height,
    float alpha = 1.0f )
```

### 7.18.2.2 Rectanagle() [2/2]

```
Primitives::Rectanagle::Rectanagle (
    const Rectanagle & s )
```

### 7.18.2.3 ~Rectanagle()

```
Primitives::Rectanagle::~~Rectanagle ( ) [virtual]
```

## 7.18.3 Member Function Documentation

#### 7.18.3.1 calculateMatrixes()

```
void Primitives::Rectanagle::calculateMatrixes ( ) [protected]
```

#### 7.18.3.2 calculateVerticesArr()

```
void Primitives::Rectanagle::calculateVerticesArr ( ) [protected]
```

#### 7.18.3.3 Display()

```
void Primitives::Rectanagle::Display ( ) [override], [virtual]
```

Draw the square.

Implements [Primitive](#).

#### 7.18.3.4 getRotation()

```
float Primitives::Rectanagle::getRotation ( )
```

#### 7.18.3.5 getX()

```
float Primitives::Rectanagle::getX ( )
```

#### 7.18.3.6 getY()

```
float Primitives::Rectanagle::getY ( )
```

#### 7.18.3.7 isNear()

```
bool Primitives::Rectanagle::isNear (
    float x,
    float y,
    float radius ) [override], [virtual]
```

is the Object int the radius

## Parameters

<i>float</i>	x - x coordinate of the center of the circle
<i>float</i>	y - y coordinate of the center of the circle
<i>float</i>	radius - radius of the circle

## Returns

true - if object is in the radius  
false - if object is not in the radius

Implements [Primitive](#).

**7.18.3.8 isVisible()**

```
bool Primitives::Rectanagle::isVisible ( ) [override], [virtual]
```

is Object on the screen

## Returns

true - if object is on the screen  
false - if object is not on the screen

Implements [Primitive](#).

**7.18.3.9 Move()**

```
void Primitives::Rectanagle::Move (
    float x,
    float y )
```

move the square relative to the current position

## Parameters

<i>float</i>	x - offset in x direction
<i>float</i>	y - offset in y direction

**7.18.3.10 operator=()**

```
Rectanagle & Primitives::Rectanagle::operator= (
    const Rectanagle & s )
```

### 7.18.3.11 rotate()

```
void Primitives::Rectangle::rotate (
    float angle )
```

rotate the square relative to the current rotation

#### Parameters

<i>float</i>	angle - angle in radians
--------------	--------------------------

### 7.18.3.12 rotateVertices()

```
Rectangle::vertices\_array Primitives::Rectangle::rotateVertices ( ) [protected]
```

Rotate the vertices of the square.

#### Parameters

<i>vertices</i>	- vertices of the square
<i>rot</i>	- rotation matrix

### 7.18.3.13 setAlpha()

```
void Primitives::Rectangle::setAlpha (
    float alpha )
```

Set the Alpha value.

#### Parameters

<i>float</i>	alpha - alpha value
--------------	---------------------

### 7.18.3.14 setB()

```
void Primitives::Rectangle::setB (
    float b )
```



### 7.18.3.15 setColor()

```
void Primitives::Rectanagle::setColor (
    float r,
    float g,
    float b,
    float alpha = 1.0f )
```

Set the [Color](#) value.

#### Parameters

<i>float</i>	r - red value
<i>float</i>	g - green value
<i>float</i>	b - blue value
<i>float</i>	alpha - alpha value

### 7.18.3.16 setG()

```
void Primitives::Rectanagle::setG (
    float g )
```

Set G color value.

#### Parameters

<i>g</i>	- green value
----------	---------------

### 7.18.3.17 setHeight()

```
void Primitives::Rectanagle::setHeight (
    float height )
```

Set the Height value.

### 7.18.3.18 setPosition()

```
void Primitives::Rectanagle::setPosition (
    float x,
    float y )
```

Set the Position object.

## Parameters

<i>float</i>	x - x coordinate of the center of the square
<i>float</i>	y - y coordinate of the center of the square

**7.18.3.19 setR()**

```
void Primitives::Rectanagle::setR (
    float r )
```

Set R color value.

## Parameters

<i>r</i>	- red value
----------	-------------

**7.18.3.20 setRotation()**

```
void Primitives::Rectanagle::setRotation (
    float angle )
```

Set the rotation of the square.

## Parameters

<i>float</i>	angle - angle in radians of the square
--------------	--

**7.18.3.21 setTexture()**

```
void Primitives::Rectanagle::setTexture (
    std::string data )
```

Set the Texture object.

## Parameters

<i>std::string</i>	data - path to the texture
--------------------	----------------------------

### 7.18.3.22 SetVerticeColor()

```
void Primitives::Rectanagle::SetVerticeColor (
    int index,
    float value,
    ColorType colorType = ERROR )
```

### 7.18.3.23 setWidth()

```
void Primitives::Rectanagle::setWidth (
    float width )
```

Set the Width value.

## 7.18.4 Member Data Documentation

### 7.18.4.1 angle

```
float Primitives::Rectanagle::angle = 0.0f [protected]
```

### 7.18.4.2 color\_arr

```
colors_array Primitives::Rectanagle::color_arr = {} [protected]
```

### 7.18.4.3 height

```
float Primitives::Rectanagle::height [protected]
```

### 7.18.4.4 inc

```
unsigned int Primitives::Rectanagle::inc[6] = {0, 1, 2, 0, 2, 3} [protected]
```

#### 7.18.4.5 vertices\_arr

```
vertices_array Primitives::Rectanagle::vertices_arr = {} [protected]
```

#### 7.18.4.6 width

```
float Primitives::Rectanagle::width [protected]
```

#### 7.18.4.7 window

```
Global::WindowProperties* Primitives::Rectanagle::window [protected]
```

#### 7.18.4.8 x

```
float Primitives::Rectanagle::x [protected]
```

#### 7.18.4.9 y

```
float Primitives::Rectanagle::y [protected]
```

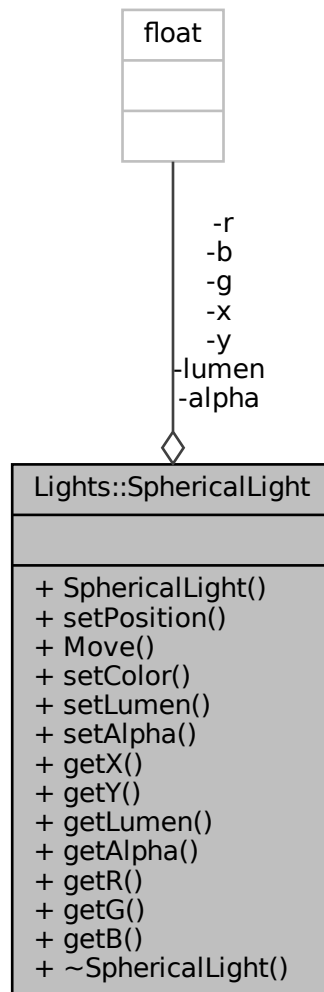
The documentation for this class was generated from the following files:

- Engine/Core/core/primitives/[Rectanagle.hpp](#)
- Engine/Core/core/primitives/[Rectanagle.cpp](#)

## 7.19 Lights::SphericalLight Class Reference

```
#include <SphericalLight.hpp>
```

Collaboration diagram for Lights::SphericalLight:



### Public Member Functions

- [SphericalLight](#) (float `x`, float `y`, float `lumen`, float `r`, float `g`, float `b`)
- void [setPosition](#) (float, float)
- void [Move](#) (float, float)
- void [setColor](#) (float, float, float)
- void [setLumen](#) (float)
- void [setAlpha](#) (float)
- float [getX](#) ()
- float [getY](#) ()

- float [getLumen](#) ()
- float [getAlpha](#) ()
- float [getR](#) ()
- float [getG](#) ()
- float [getB](#) ()
- [~SphericalLight](#) ()

## Private Attributes

- float [x](#)
- float [y](#)
- float [lumen](#)
- float [alpha](#) = 1.0f
- float [r](#)
- float [g](#)
- float [b](#)

## 7.19.1 Constructor & Destructor Documentation

### 7.19.1.1 SphericalLight()

```
Lights::SphericalLight::SphericalLight (
    float x,
    float y,
    float lumen,
    float r,
    float g,
    float b )
```

### 7.19.1.2 ~SphericalLight()

```
Lights::SphericalLight::~~SphericalLight ( )
```

## 7.19.2 Member Function Documentation

### 7.19.2.1 getAlpha()

```
float Lights::SphericalLight::getAlpha ( )
```

### 7.19.2.2 getB()

```
float Lights::SphericalLight::getB ( )
```

### 7.19.2.3 getG()

```
float Lights::SphericalLight::getG ( )
```

### 7.19.2.4 getLumen()

```
float Lights::SphericalLight::getLumen ( )
```

### 7.19.2.5 getR()

```
float Lights::SphericalLight::getR ( )
```

### 7.19.2.6 getX()

```
float Lights::SphericalLight::getX ( )
```

### 7.19.2.7 getY()

```
float Lights::SphericalLight::getY ( )
```

### 7.19.2.8 Move()

```
void Lights::SphericalLight::Move (
    float x,
    float y )
```

#### 7.19.2.9 setAlpha()

```
void Lights::SphericalLight::setAlpha (
    float alpha )
```

#### 7.19.2.10 setColor()

```
void Lights::SphericalLight::setColor (
    float r,
    float g,
    float b )
```

#### 7.19.2.11 setLumen()

```
void Lights::SphericalLight::setLumen (
    float lumen )
```

#### 7.19.2.12 setPosition()

```
void Lights::SphericalLight::setPosition (
    float x,
    float y )
```

### 7.19.3 Member Data Documentation

#### 7.19.3.1 alpha

```
float Lights::SphericalLight::alpha = 1.0f [private]
```

#### 7.19.3.2 b

```
float Lights::SphericalLight::b [private]
```



### 7.19.3.3 g

```
float Lights::SphericalLight::g [private]
```

### 7.19.3.4 lumen

```
float Lights::SphericalLight::lumen [private]
```

### 7.19.3.5 r

```
float Lights::SphericalLight::r [private]
```

### 7.19.3.6 x

```
float Lights::SphericalLight::x [private]
```

### 7.19.3.7 y

```
float Lights::SphericalLight::y [private]
```

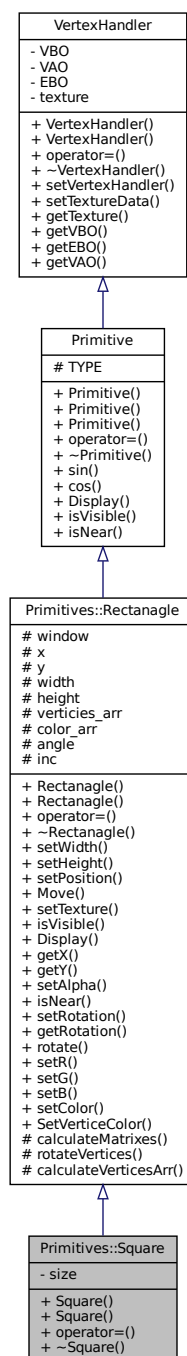
The documentation for this class was generated from the following files:

- Engine/Core/core/primitives/Light/[SphericalLight.hpp](#)
- Engine/Core/core/primitives/Light/[SphericalLight.cpp](#)

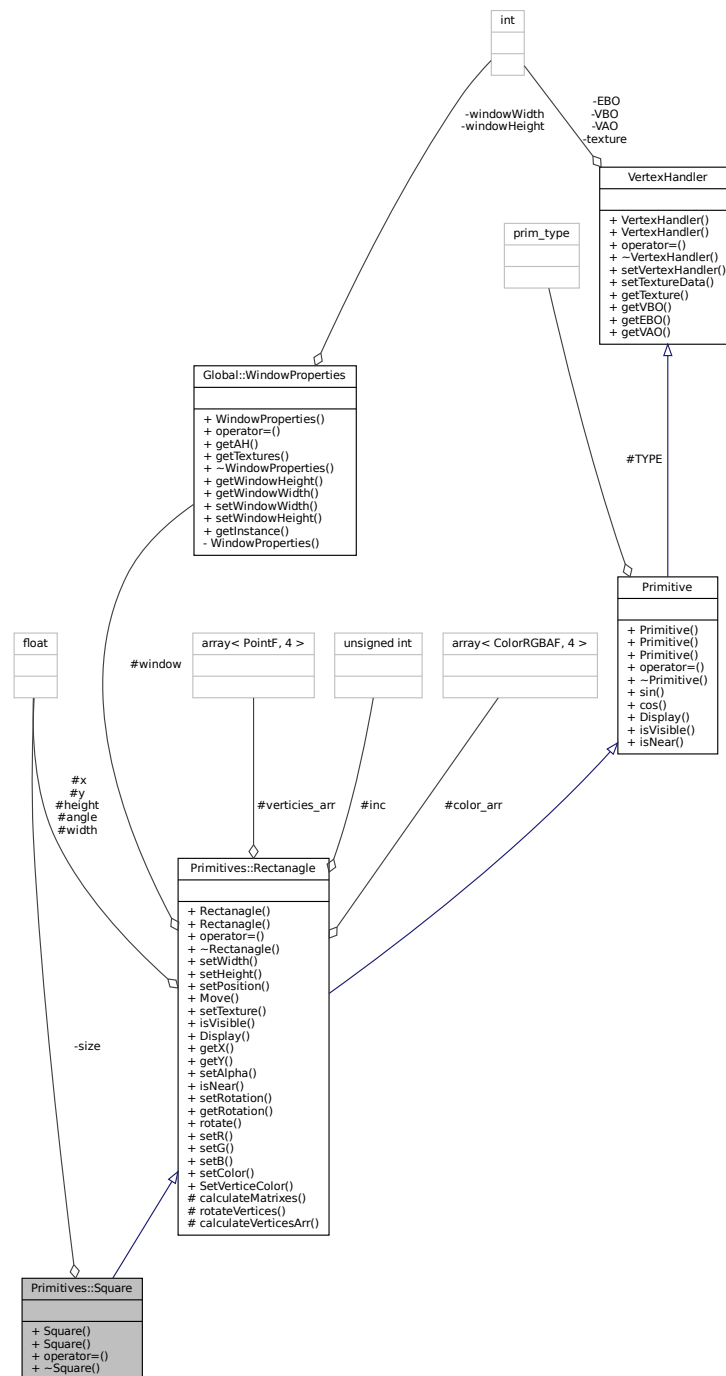
## 7.20 Primitives::Square Class Reference

```
#include <Square.hpp>
```

Inheritance diagram for Primitives::Square:



Collaboration diagram for Primitives::Square:



## Public Member Functions

- [Square](#) (float, float, float, float=1.0f)
- [Square](#) (const [Square](#) &s)
- [Square](#) & `operator=` (const [Square](#) &s)
- virtual `~Square` ()

## Private Attributes

- float `size` = 0

## Additional Inherited Members

### 7.20.1 Constructor & Destructor Documentation

#### 7.20.1.1 `Square()` [1/2]

```
Primitives::Square::Square (
    float x,
    float y,
    float a,
    float alpha = 1.0f )
```

#### 7.20.1.2 `Square()` [2/2]

```
Primitives::Square::Square (
    const Square & s )
```

#### 7.20.1.3 `~Square()`

```
Primitives::Square::~~Square ( ) [virtual]
```

### 7.20.2 Member Function Documentation

#### 7.20.2.1 `operator=()`

```
Square & Primitives::Square::operator= (
    const Square & s )
```

### 7.20.3 Member Data Documentation

## 7.20.3.1 size

```
float Primitives::Square::size = 0 [private]
```

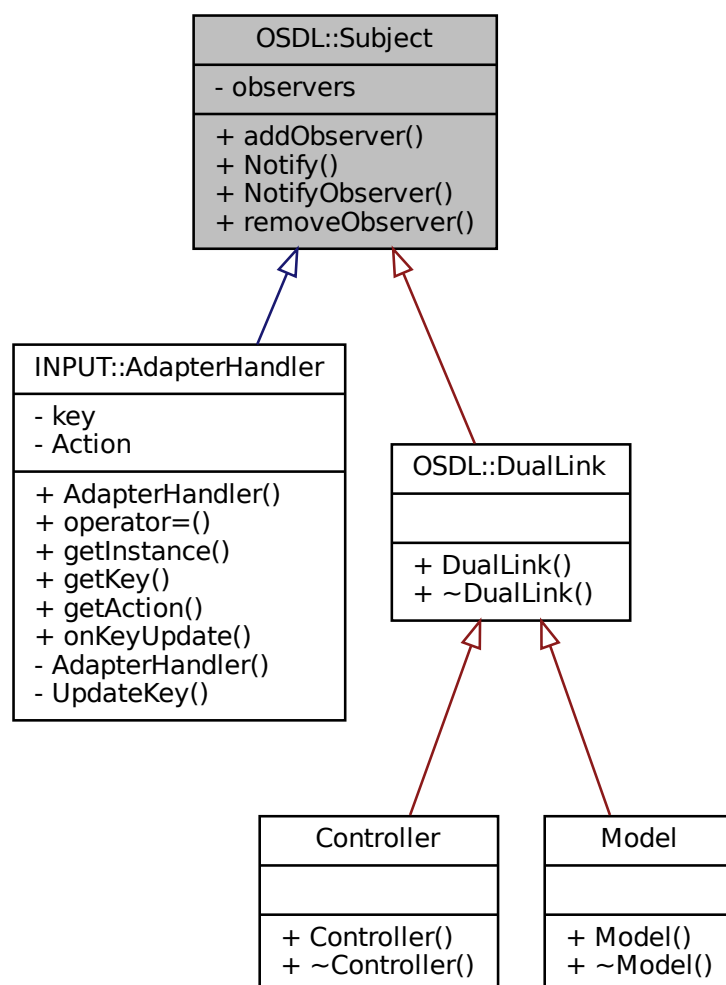
The documentation for this class was generated from the following files:

- [Engine/Core/core/primitives/Square.hpp](#)
- [Engine/Core/core/primitives/Square.cpp](#)

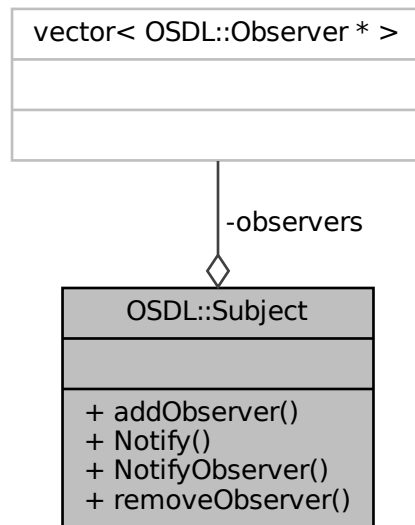
## 7.21 OSDL::Subject Class Reference

```
#include <Subject.hpp>
```

Inheritance diagram for OSDL::Subject:



Collaboration diagram for OSDL::Subject:



## Public Member Functions

- void `addObserver` (`Observer` &)
- void `Notify` ()
- void `NotifyObserver` (`Observer` &)
- void `removeObserver` (`Observer` &)

## Private Attributes

- `std::vector< Observer * > observers`

## 7.21.1 Member Function Documentation

### 7.21.1.1 addObserver()

```
void OSDL::Subject::addObserver (  
    Observer & observer )
```

### 7.21.1.2 Notify()

```
void OSDL::Subject::Notify ( )
```

### 7.21.1.3 NotifyObserver()

```
void OSDL::Subject::NotifyObserver (
    Observer & o )
```

### 7.21.1.4 removeObserver()

```
void OSDL::Subject::removeObserver (
    Observer & o )
```

## 7.21.2 Member Data Documentation

### 7.21.2.1 observers

```
std::vector<Observer*> OSDL::Subject::observers [private]
```

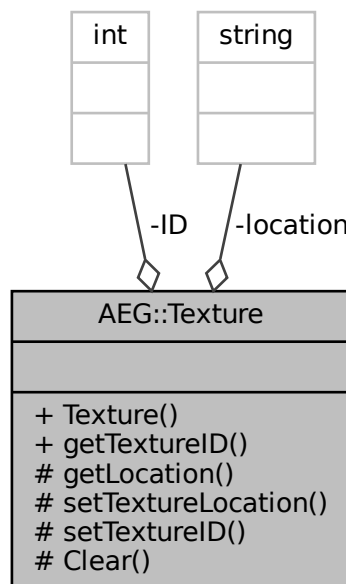
The documentation for this class was generated from the following files:

- Engine/Core/patterns/ObserverSubject/[Subject.hpp](#)
- Engine/Core/patterns/ObserverSubject/[Subject.cpp](#)

## 7.22 AEG::Texture Class Reference

```
#include <Texture.hpp>
```

Collaboration diagram for AEG::Texture:



## Public Member Functions

- [Texture](#) (std::string, unsigned int)
- const unsigned int & [getTextureID](#) ()

## Protected Member Functions

- const std::string & [getLocation](#) ()
- void [setTextureLocation](#) (std::string)
- void [setTextureID](#) (unsigned int)
- void [Clear](#) ()

## Private Attributes

- unsigned int [ID](#)
- std::string [location](#)

## Friends

- class [Textures](#)



## 7.22.1 Constructor & Destructor Documentation

### 7.22.1.1 Texture()

```
AEG::Texture::Texture (
    std::string location,
    unsigned int ID )
```

## 7.22.2 Member Function Documentation

### 7.22.2.1 Clear()

```
void AEG::Texture::Clear ( ) [protected]
```

### 7.22.2.2 getLocation()

```
const std::string & AEG::Texture::getLocation ( ) [protected]
```

### 7.22.2.3 getTextureID()

```
const unsigned int & AEG::Texture::getTextureID ( )
```

### 7.22.2.4 setTextureID()

```
void AEG::Texture::setTextureID (
    unsigned int ID ) [protected]
```

### 7.22.2.5 setTextureLocation()

```
void AEG::Texture::setTextureLocation (
    std::string location ) [protected]
```

### 7.22.3 Friends And Related Function Documentation

#### 7.22.3.1 Textures

```
friend class Textures [friend]
```

### 7.22.4 Member Data Documentation

#### 7.22.4.1 ID

```
unsigned int AEG::Texture::ID [private]
```

#### 7.22.4.2 location

```
std::string AEG::Texture::location [private]
```

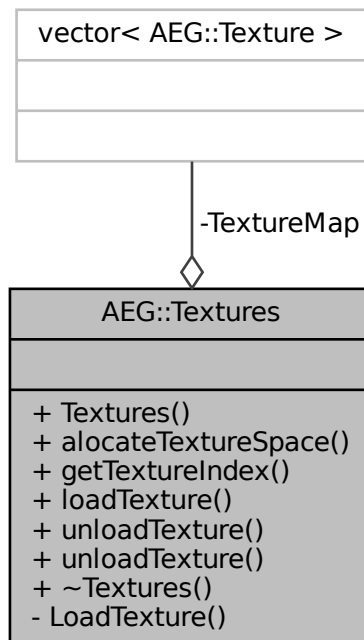
The documentation for this class was generated from the following files:

- Engine/Core/core/base/[Texture.hpp](#)
- Engine/Core/core/base/[Texture.cpp](#)

## 7.23 AEG::Textures Class Reference

```
#include <Texture.hpp>
```

Collaboration diagram for AEG::Textures:



## Public Member Functions

- [Textures](#) ()
- void [allocateTextureSpace](#) (unsigned int)
- const unsigned int & [getTextureIndex](#) (const std::string &)
- void [loadTexture](#) (const std::string &)
- void [unloadTexture](#) (const std::string &)
- void [unloadTexture](#) (unsigned int)
- [~Textures](#) ()

## Private Member Functions

- void [LoadTexture](#) (int, const std::string &)

## Private Attributes

- std::vector< [Texture](#) > [TextureMap](#)

### 7.23.1 Constructor & Destructor Documentation

### 7.23.1.1 Textures()

```
AEG::Textures::Textures ( )
```

### 7.23.1.2 ~Textures()

```
AEG::Textures::~~Textures ( )
```

## 7.23.2 Member Function Documentation

### 7.23.2.1 allocateTextureSpace()

```
void AEG::Textures::allocateTextureSpace (
    unsigned int size )
```

### 7.23.2.2 getTextureIndex()

```
const unsigned int & AEG::Textures::getTextureIndex (
    const std::string & link )
```

### 7.23.2.3 loadTexture()

```
void AEG::Textures::loadTexture (
    const std::string & link )
```

### 7.23.2.4 LoadTexture()

```
void AEG::Textures::LoadTexture (
    int ID,
    const std::string & link ) [private]
```

### 7.23.2.5 unloadTexture() [1/2]

```
void AEG::Textures::unloadTexture (
    const std::string & link )
```

### 7.23.2.6 unloadTexture() [2/2]

```
void AEG::Textures::unloadTexture (
    unsigned int ID )
```

## 7.23.3 Member Data Documentation

### 7.23.3.1 TextureMap

```
std::vector<Texture> AEG::Textures::TextureMap [private]
```

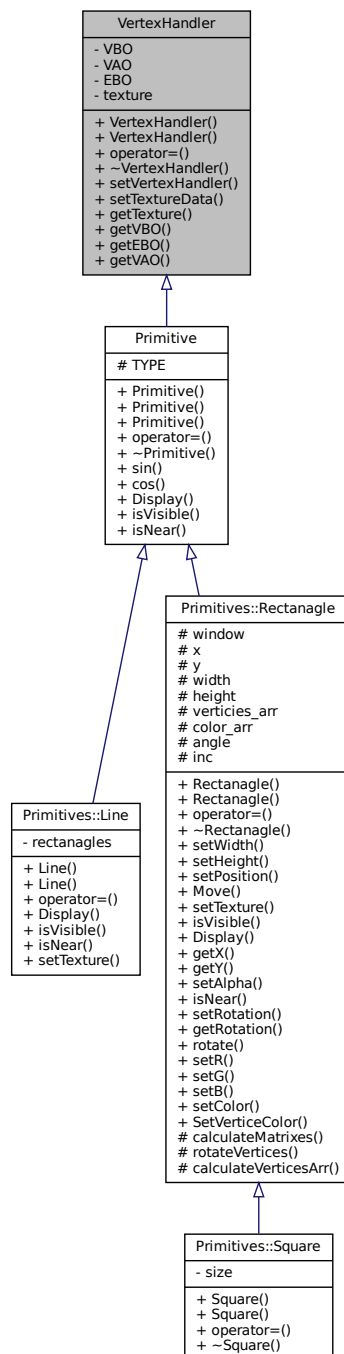
The documentation for this class was generated from the following files:

- Engine/Core/core/base/[Texture.hpp](#)
- Engine/Core/core/base/[Texture.cpp](#)

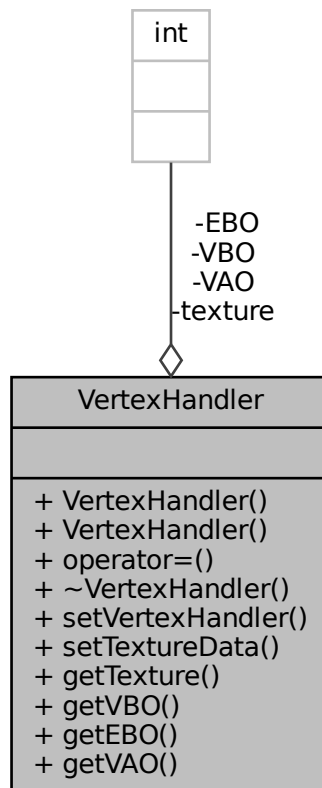
## 7.24 VertexHandler Class Reference

```
#include <vertexHandler.hpp>
```

Inheritance diagram for VertexHandler:



Collaboration diagram for VertexHandler:



## Public Member Functions

- [VertexHandler](#) (bool isDefaultPrimitive)
- [VertexHandler](#) (const [VertexHandler](#) &other)
- [VertexHandler](#) & [operator=](#) (const [VertexHandler](#) &other)
- [~VertexHandler](#) ()
- void [setVertexHandler](#) (float vertices[], int VerticesSize, unsigned int indices[], int indicesSize)
- void [setTextureData](#) (const std::string)
- int [getTexture](#) ()
- int [getVBO](#) ()
- int [getEBO](#) ()
- int [getVAO](#) ()

## Private Attributes

- unsigned int [VBO](#)
- unsigned int [VAO](#)
- unsigned int [EBO](#)
- unsigned int [texture](#)

## 7.24.1 Constructor & Destructor Documentation

### 7.24.1.1 VertexHandler() [1/2]

```
VertexHandler::VertexHandler (
    bool isDefaultPrimitive )
```

### 7.24.1.2 VertexHandler() [2/2]

```
VertexHandler::VertexHandler (
    const VertexHandler & other )
```

### 7.24.1.3 ~VertexHandler()

```
VertexHandler::~VertexHandler ( )
```

## 7.24.2 Member Function Documentation

### 7.24.2.1 getEBO()

```
int VertexHandler::getEBO ( )
```

### 7.24.2.2 getTexture()

```
int VertexHandler::getTexture ( )
```

### 7.24.2.3 getVAO()

```
int VertexHandler::getVAO ( )
```



#### 7.24.2.4 getVBO()

```
int VertexHandler::getVBO ( )
```

#### 7.24.2.5 operator=()

```
VertexHandler & VertexHandler::operator= (
    const VertexHandler & other )
```

#### 7.24.2.6 setTextureData()

```
void VertexHandler::setTextureData (
    const std::string data )
```

#### 7.24.2.7 setVertexHandler()

```
void VertexHandler::setVertexHandler (
    float vertices[],
    int VerticesSize,
    unsigned int indices[],
    int indicesSize )
```

### 7.24.3 Member Data Documentation

#### 7.24.3.1 EBO

```
unsigned int VertexHandler::EBO [private]
```

#### 7.24.3.2 texture

```
unsigned int VertexHandler::texture [private]
```

### 7.24.3.3 VAO

```
unsigned int VertexHandler::VAO [private]
```

### 7.24.3.4 VBO

```
unsigned int VertexHandler::VBO [private]
```

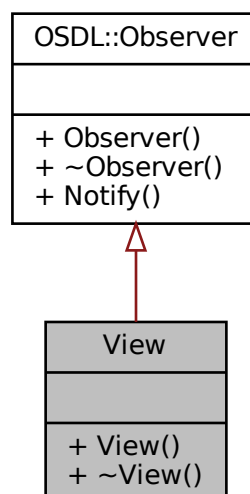
The documentation for this class was generated from the following files:

- Engine/Core/core/base/[vertexHandler.hpp](#)
- Engine/Core/core/base/[vertexHandler.cpp](#)

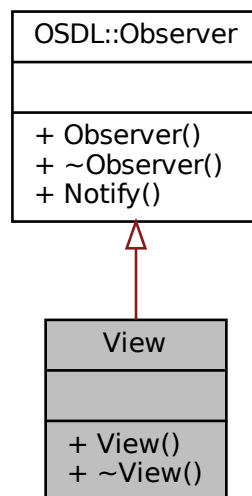
## 7.25 View Class Reference

```
#include <View.hpp>
```

Inheritance diagram for View:



Collaboration diagram for View:



## Public Member Functions

- [View](#) ()
- [~View](#) ()

## Additional Inherited Members

### 7.25.1 Constructor & Destructor Documentation

#### 7.25.1.1 View()

```
View::View ( )
```

#### 7.25.1.2 ~View()

```
View::~~View ( )
```

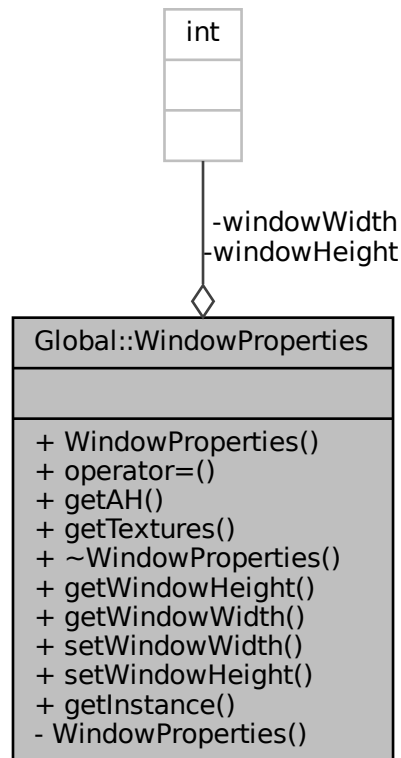
The documentation for this class was generated from the following file:

- [Engine/Core/patterns/MVC/View.hpp](#)

## 7.26 Global::WindowProperties Class Reference

```
#include <WindowProperties.hpp>
```

Collaboration diagram for Global::WindowProperties:



### Public Member Functions

- [WindowProperties](#) ([WindowProperties](#) &other)=delete  
*ArrayHandler for All [GPU](#) bindings.*
- void [operator=](#) (const [WindowProperties](#) &)=delete
- [GPU::ArrayHandler](#) \* [getAH](#) ()
- [AEG::Textures](#) \* [getTextures](#) ()
- [~WindowProperties](#) ()
- const int & [getWindowHeight](#) ()
- const int & [getWindowWidth](#) ()
- void [setWindowWidth](#) (int)
- void [setWindowHeight](#) (int)

### Static Public Member Functions

- static [WindowProperties](#) & [getInstance](#) ()

## Private Member Functions

- [WindowProperties](#) ()

## Private Attributes

- int [windowHeight](#)
- int [windowWidth](#)

## 7.26.1 Constructor & Destructor Documentation

### 7.26.1.1 WindowProperties() [1/2]

```
Global::WindowProperties::WindowProperties ( ) [private]
```

### 7.26.1.2 WindowProperties() [2/2]

```
Global::WindowProperties::WindowProperties (
    WindowProperties & other ) [delete]
```

ArrayHandler for All [GPU](#) bindings.

#### Note

#### Return values

<i>None</i>	
-------------	--

### 7.26.1.3 ~WindowProperties()

```
Global::WindowProperties::~~WindowProperties ( )
```

## 7.26.2 Member Function Documentation

### 7.26.2.1 getAH()

```
GPU::ArrayHandler * Global::WindowProperties::getAH ( )
```

### 7.26.2.2 getInstance()

```
WindowProperties & Global::WindowProperties::getInstance ( ) [static]
```

### 7.26.2.3 getTextures()

```
AEGL::Textures * Global::WindowProperties::getTextures ( )
```

### 7.26.2.4 getWindowHeight()

```
const int & Global::WindowProperties::getWindowHeight ( )
```

### 7.26.2.5 getWindowWidth()

```
const int & Global::WindowProperties::getWindowWidth ( )
```

### 7.26.2.6 operator=()

```
void Global::WindowProperties::operator= (
    const WindowProperties & ) [delete]
```

### 7.26.2.7 setWindowHeight()

```
void Global::WindowProperties::setWindowHeight (
    int height )
```

### 7.26.2.8 setWindowWidth()

```
void Global::WindowProperties::setWindowWidth (
    int width )
```

## 7.26.3 Member Data Documentation

### 7.26.3.1 windowHeight

```
int Global::WindowProperties::windowHeight [private]
```

### 7.26.3.2 windowWidth

```
int Global::WindowProperties::windowWidth [private]
```

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

- Engine/Core/core/[WindowProperties.hpp](#)
- Engine/Core/core/[WindowProperties.cpp](#)



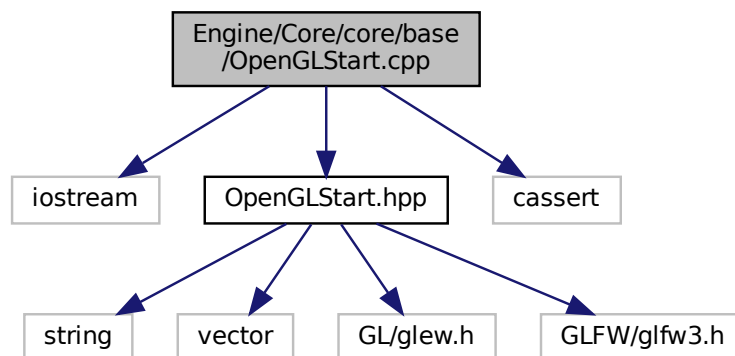


## Chapter 8

# File Documentation

### 8.1 Engine/Core/core/base/OpenGLStart.cpp File Reference

```
#include <iostream>
#include "OpenGLStart.hpp"
#include <cassert>
Include dependency graph for OpenGLStart.cpp:
```



### Namespaces

- [OpenGLInstance](#)

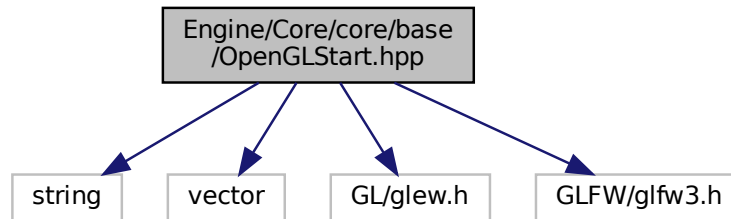
### Functions

- GLFWwindow \* [OpenGLInstance::InitWindow](#) (int width, int height, std::string name)
- void [OpenGLInstance::setClearColor](#) (float R, float G, float B, float A)
- std::vector< GLuint > [OpenGLInstance::CreateAndCompileShaders](#) (const char \*vertShaderCode, const char \*fragShaderCode)
- GLuint [OpenGLInstance::CreateProgramAndLinkShaders](#) (const std::vector< GLuint > shaders)

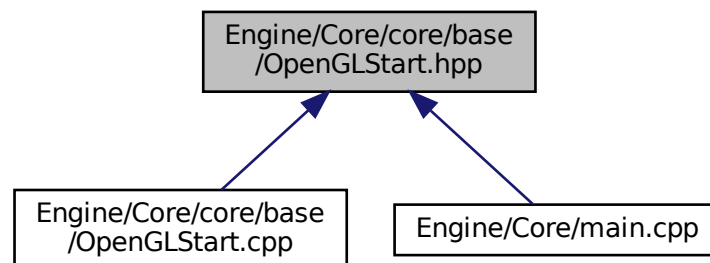
## 8.2 Engine/Core/core/base/OpenGLStart.hpp File Reference

```
#include <string>
#include <vector>
#include <GL/glew.h>
#include <GLFW/glfw3.h>
```

Include dependency graph for OpenGLStart.hpp:



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



### Namespaces

- [OpenGLInstance](#)

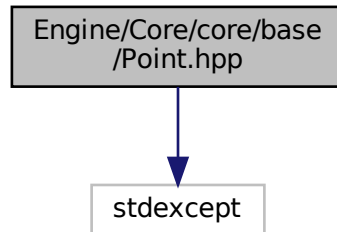
### Functions

- GLFWwindow \* [OpenGLInstance::InitWindow](#) (int width, int height, std::string name)
- void [OpenGLInstance::setClearColor](#) (float R, float G, float B, float A)
- std::vector< GLuint > [OpenGLInstance::CreateAndCompileShaders](#) (const char \*vertShaderCode, const char \*fragShaderCode)
- GLuint [OpenGLInstance::CreateProgramAndLinkShaders](#) (const std::vector< GLuint > shaders)

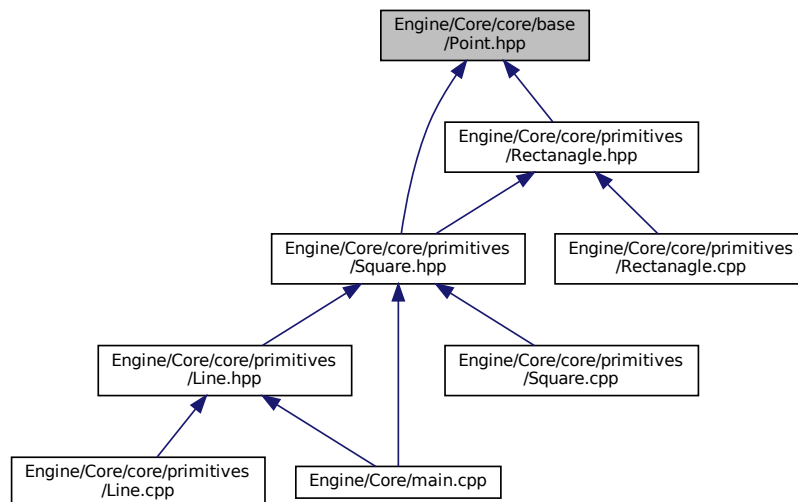
## 8.3 Engine/Core/core/base/Point.hpp File Reference

```
#include <stdexcept>
```

Include dependency graph for Point.hpp:



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



### Classes

- struct [Primitives::Point2D< T >](#)
- struct [Primitives::Color< T >](#)
- struct [Primitives::ColorRGBA< T >](#)

### Namespaces

- [Primitives](#)

## Typedefs

- typedef struct Point2D< float > [Primitives::PointF](#)
- typedef Point2D< int > [Primitives::PointI](#)
- typedef Point2D< double > [Primitives::PointD](#)
- typedef Point2D< unsigned int > [Primitives::PointU](#)
- typedef Point2D< unsigned long > [Primitives::PointUL](#)
- typedef Point2D< unsigned long long > [Primitives::PointULL](#)
- typedef Point2D< long > [Primitives::PointL](#)
- typedef Point2D< long long > [Primitives::PointLL](#)
- typedef Point2D< char > [Primitives::PointC](#)
- typedef Point2D< short > [Primitives::PointS](#)
- typedef struct Color< float > [Primitives::ColorF](#)
- typedef Color< int > [Primitives::ColorI](#)
- typedef Color< double > [Primitives::ColorD](#)
- typedef Color< unsigned int > [Primitives::ColorU](#)
- typedef Color< unsigned long > [Primitives::ColorUL](#)
- typedef Color< unsigned long long > [Primitives::ColorULL](#)
- typedef Color< long > [Primitives::ColorL](#)
- typedef Color< long long > [Primitives::ColorLL](#)
- typedef Color< char > [Primitives::ColorC](#)
- typedef Color< short > [Primitives::ColorS](#)
- typedef struct ColorRGBA< float > [Primitives::ColorRGBAf](#)
- typedef ColorRGBA< int > [Primitives::ColorRGBAI](#)
- typedef ColorRGBA< double > [Primitives::ColorRGBAD](#)
- typedef ColorRGBA< unsigned int > [Primitives::ColorRGBAUI](#)
- typedef ColorRGBA< unsigned long > [Primitives::ColorRGBAU](#)
- typedef ColorRGBA< unsigned long long > [Primitives::ColorRGBAU](#)
- typedef ColorRGBA< long > [Primitives::ColorRGBAL](#)
- typedef ColorRGBA< long long > [Primitives::ColorRGBALL](#)
- typedef ColorRGBA< char > [Primitives::ColorRGBAC](#)
- typedef ColorRGBA< short > [Primitives::ColorRGBAS](#)

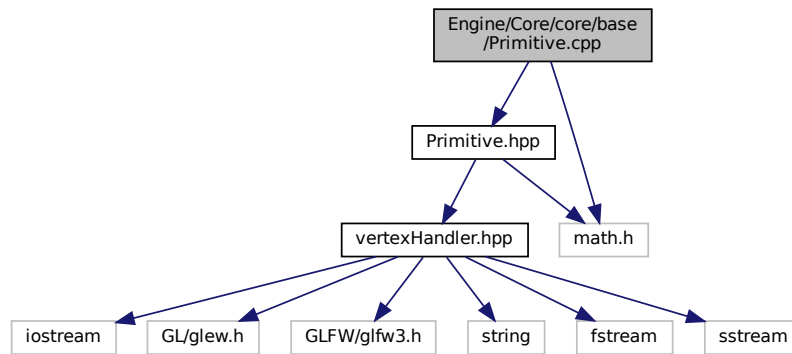
## Enumerations

- enum [Primitives::ColorType](#) {  
[Primitives::RED](#), [Primitives::GREEN](#), [Primitives::BLUE](#), [Primitives::ALPHA](#),  
[Primitives::ERROR](#) }

## 8.4 Engine/Core/core/base/Primitive.cpp File Reference

```
#include "Primitive.hpp"
#include <math.h>
```

Include dependency graph for Primitive.cpp:

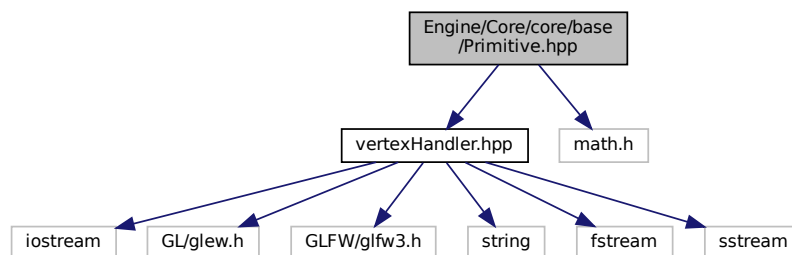


## 8.5 Engine/Core/core/base/Primitive.hpp File Reference

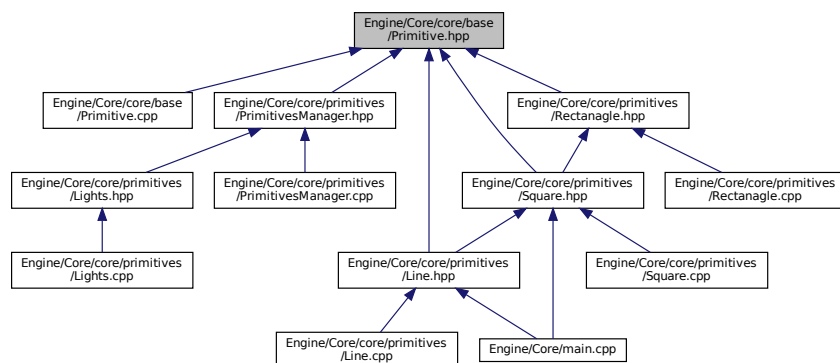
```
#include "vertexHandler.hpp"
```

```
#include <math.h>
```

Include dependency graph for Primitive.hpp:



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



## Classes

- class [Primitive](#)

## Macros

- `#define` [M\\_PI](#) 3.14159265358979323846

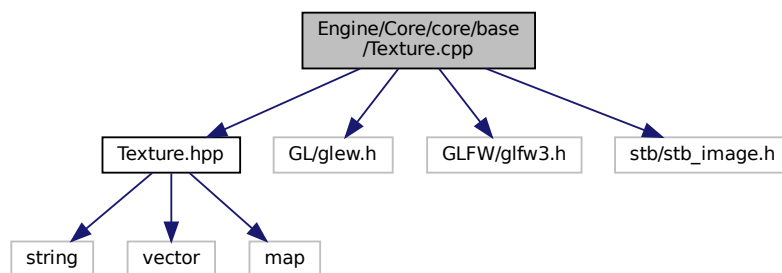
### 8.5.1 Macro Definition Documentation

#### 8.5.1.1 M\_PI

```
#define M_PI 3.14159265358979323846
```

## 8.6 Engine/Core/core/base/Texture.cpp File Reference

```
#include "Texture.hpp"  
#include <GL/glew.h>  
#include <GLFW/glfw3.h>  
#include "stb/stb_image.h"  
Include dependency graph for Texture.cpp:
```



## Namespaces

- [AEG](#)

## Macros

- `#define` [STB\\_IMAGE\\_IMPLEMENTATION](#)

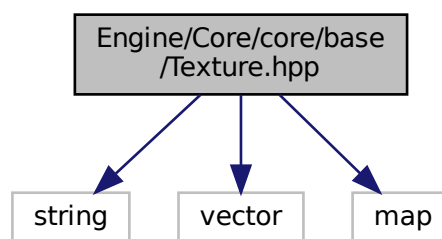
## 8.6.1 Macro Definition Documentation

### 8.6.1.1 STB\_IMAGE\_IMPLEMENTATION

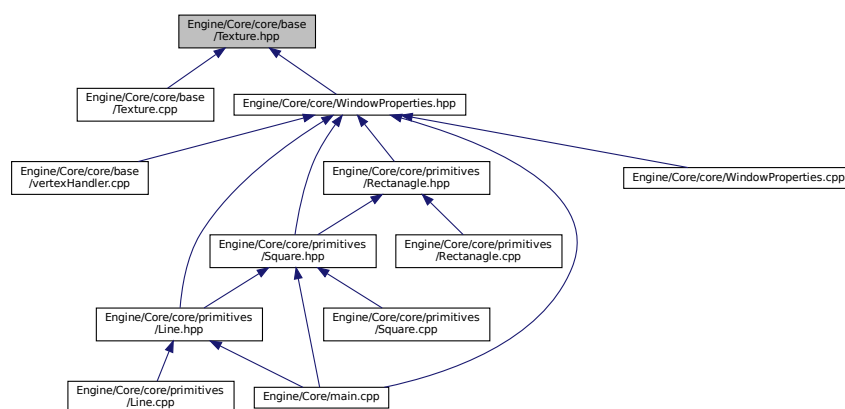
```
#define STB_IMAGE_IMPLEMENTATION
```

## 8.7 Engine/Core/core/base/Texture.hpp File Reference

```
#include <string>
#include <vector>
#include <map>
Include dependency graph for Texture.hpp:
```



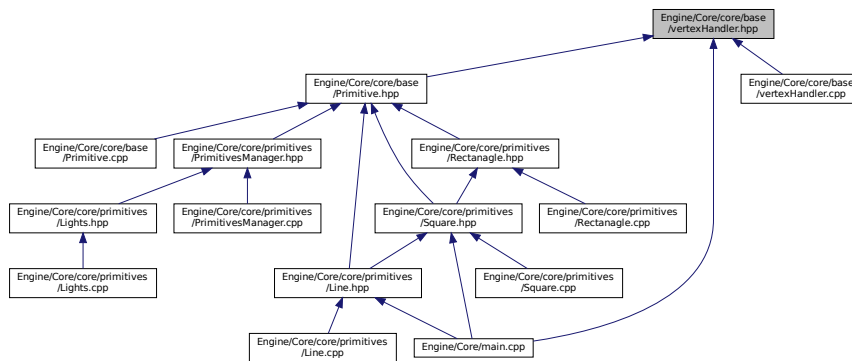
This graph shows which files directly or indirectly include this file:







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



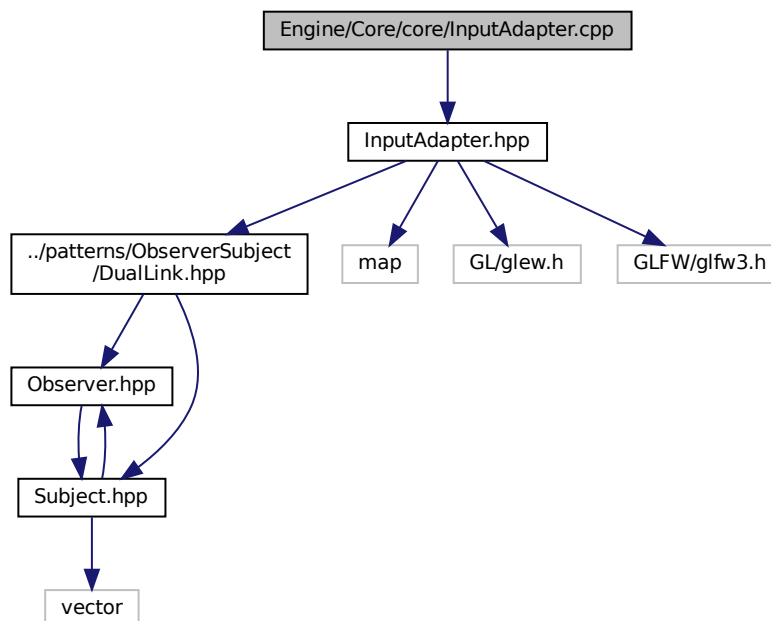
## Classes

- class **VertexHandler**

## 8.10 Engine/Core/core/InputAdapter.cpp File Reference

```
#include "InputAdapter.hpp"
```

Include dependency graph for InputAdapter.cpp:



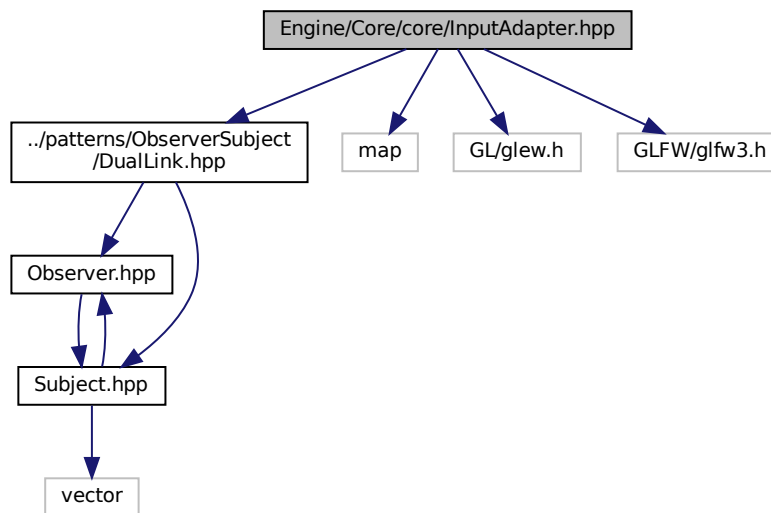
## Namespaces

- [INPUT](#)

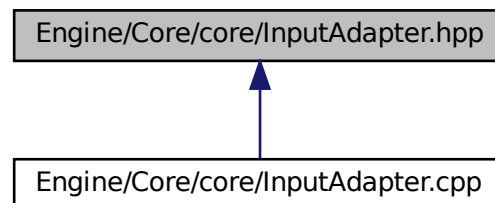
## 8.11 Engine/Core/core/InputAdapter.hpp File Reference

```
#include "../patterns/ObserverSubject/DualLink.hpp"
#include <map>
#include <GL/glew.h>
#include <GLFW/glfw3.h>
```

Include dependency graph for InputAdapter.hpp:



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



## Classes

- class [INPUT::Adapter](#)
- class [INPUT::AdapterHandler](#)

## Namespaces

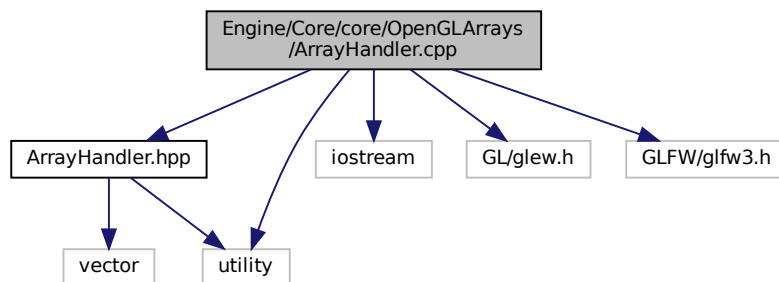
- [INPUT](#)

## Enumerations

- enum [INPUT::KeyBoardKey](#) {  
[INPUT::ERROR](#) = -1, [INPUT::Key\\_1](#) = 0, [INPUT::Key\\_2](#), [INPUT::Key\\_3](#),  
[INPUT::Key\\_4](#), [INPUT::Key\\_5](#), [INPUT::Key\\_6](#), [INPUT::Key\\_7](#),  
[INPUT::Key\\_8](#), [INPUT::Key\\_9](#), [INPUT::Key\\_0](#), [INPUT::Q](#),  
[INPUT::W](#), [INPUT::E](#), [INPUT::R](#), [INPUT::T](#),  
[INPUT::Y](#), [INPUT::U](#), [INPUT::I](#), [INPUT::O](#),  
[INPUT::P](#), [INPUT::A](#), [INPUT::S](#), [INPUT::D](#),  
[INPUT::F](#), [INPUT::G](#), [INPUT::H](#), [INPUT::J](#),  
[INPUT::K](#), [INPUT::L](#), [INPUT::Z](#), [INPUT::X](#),  
[INPUT::C](#), [INPUT::V](#), [INPUT::B](#), [INPUT::N](#),  
[INPUT::M](#), [INPUT::Arrow\\_Up](#), [INPUT::Arrow\\_Down](#), [INPUT::Arrow\\_Left](#),  
[INPUT::Arrow\\_Right](#) }

## 8.12 Engine/Core/core/OpenGLArrays/ArrayHandler.cpp File Reference

```
#include "ArrayHandler.hpp"
#include <iostream>
#include <GL/glew.h>
#include <GLFW/glfw3.h>
#include <utility>
Include dependency graph for ArrayHandler.cpp:
```

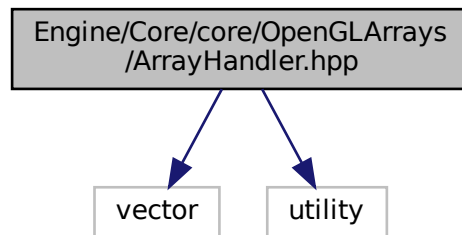


## Namespaces

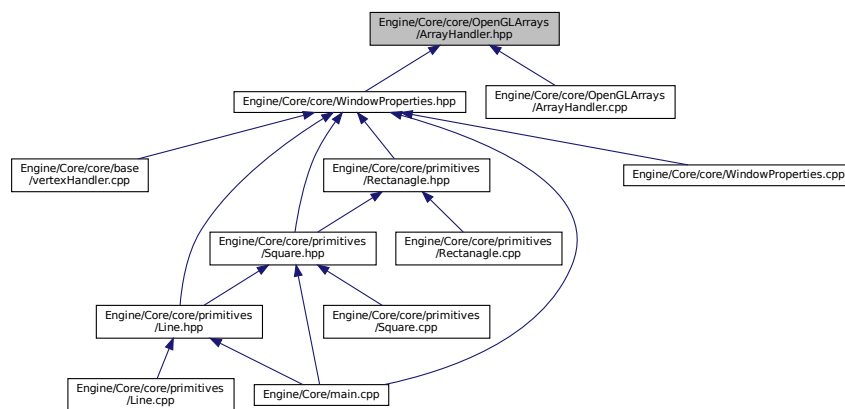
- [GPU](#)

## 8.13 Engine/Core/core/OpenGLArrays/ArrayHandler.hpp File Reference

```
#include <vector>
#include <utility>
Include dependency graph for ArrayHandler.hpp:
```



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



### Classes

- class [GPU::GPU\\_Ref](#)
- class [GPU::ArrayHandler](#)

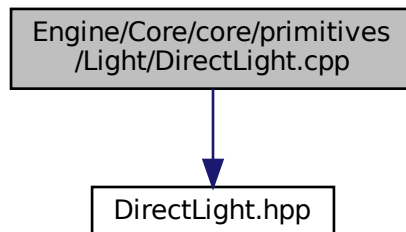
### Namespaces

- [GPU](#)

## 8.14 Engine/Core/core/primitives/Light/DirectLight.cpp File Reference

```
#include "DirectLight.hpp"
```

Include dependency graph for DirectLight.cpp:

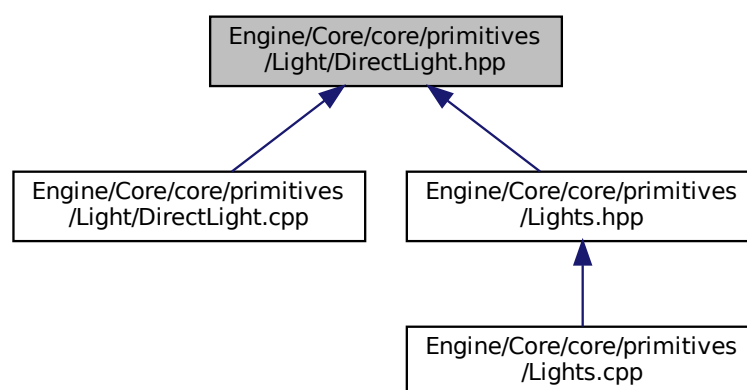


### Namespaces

- [Lights](#)

## 8.15 Engine/Core/core/primitives/Light/DirectLight.hpp File Reference

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



### Classes

- class [Lights::DirectLight](#)

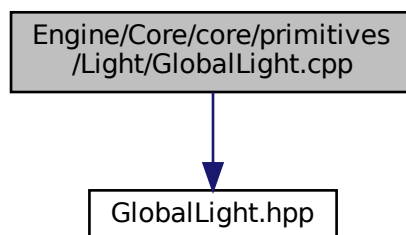
## Namespaces

- [Lights](#)

## 8.16 Engine/Core/core/primitives/Light/GlobalLight.cpp File Reference

```
#include "GlobalLight.hpp"
```

Include dependency graph for GlobalLight.cpp:

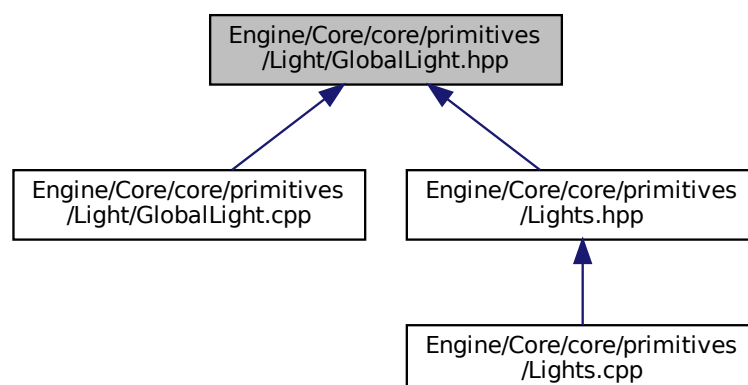


## Namespaces

- [Lights](#)

## 8.17 Engine/Core/core/primitives/Light/GlobalLight.hpp File Reference

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



## Classes

- class [Lights::GlobalLight](#)

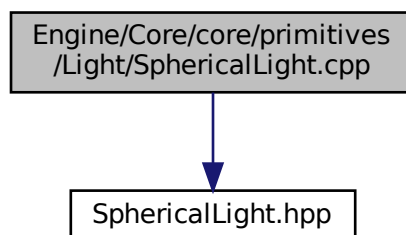
## Namespaces

- [Lights](#)

## 8.18 Engine/Core/core/primitives/Light/SphericalLight.cpp File Reference

```
#include "SphericalLight.hpp"
```

Include dependency graph for SphericalLight.cpp:

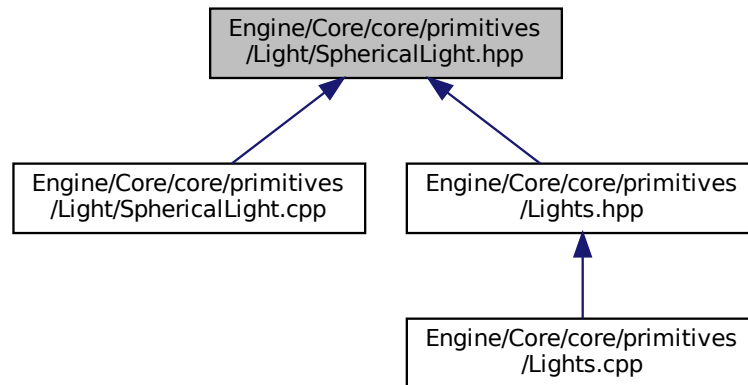


## Namespaces

- [Lights](#)

## 8.19 Engine/Core/core/primitives/Light/SphericalLight.hpp File Reference

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



### Classes

- class [Lights::SphericalLight](#)

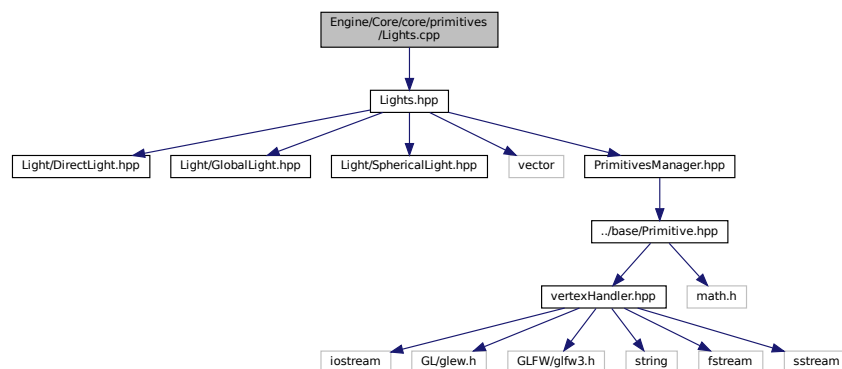
### Namespaces

- [Lights](#)

## 8.20 Engine/Core/core/primitives/Lights.cpp File Reference

```
#include "Lights.hpp"
```

Include dependency graph for `Lights.cpp`:





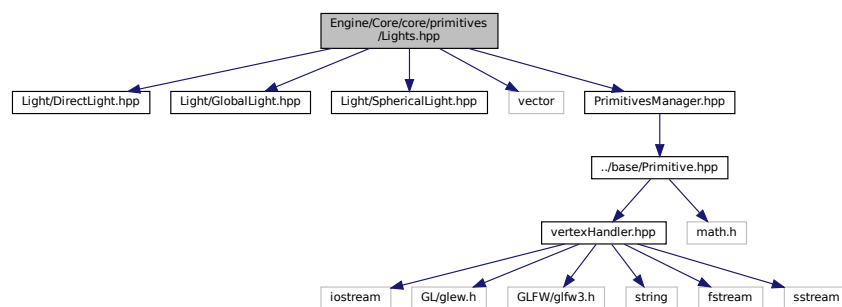
## Namespaces

- [Lights](#)

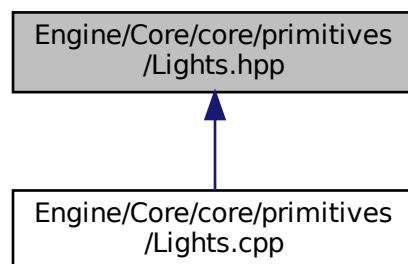
## 8.21 Engine/Core/core/primitives/Lights.hpp File Reference

```
#include "Light/DirectLight.hpp"
#include "Light/GlobalLight.hpp"
#include "Light/SphericalLight.hpp"
#include <vector>
#include "PrimitivesManager.hpp"
```

Include dependency graph for Lights.hpp:



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



## Classes

- class [Lights::LightManager](#)

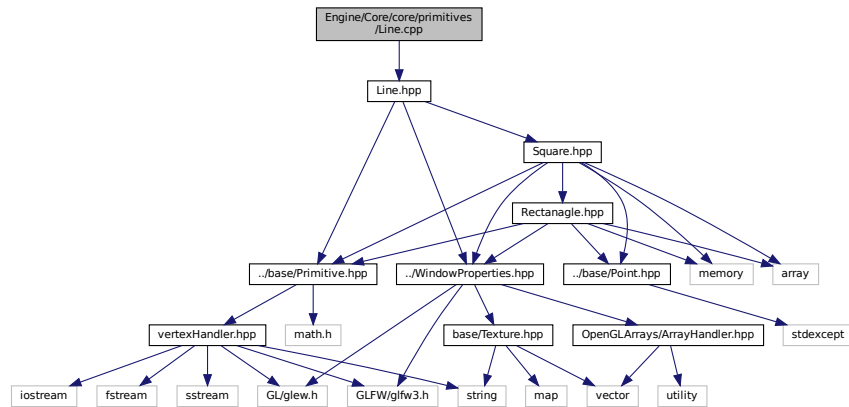
## Namespaces

- [Lights](#)

## 8.22 Engine/Core/core/primitives/Line.cpp File Reference

```
#include "Line.hpp"
```

Include dependency graph for Line.cpp:



### Namespaces

- [Primitives](#)

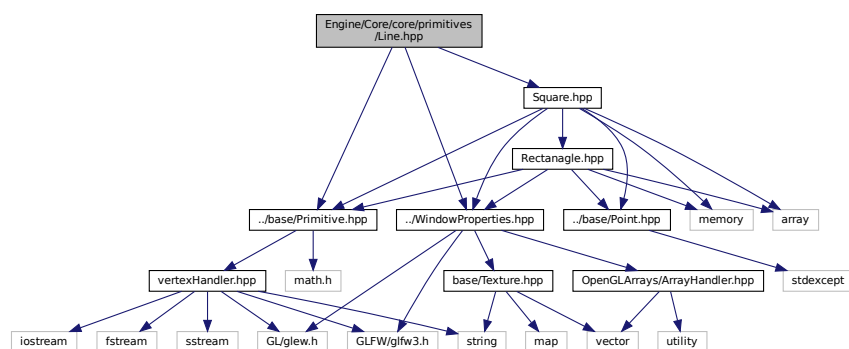
## 8.23 Engine/Core/core/primitives/Line.hpp File Reference

```
#include "../base/Primitive.hpp"
```

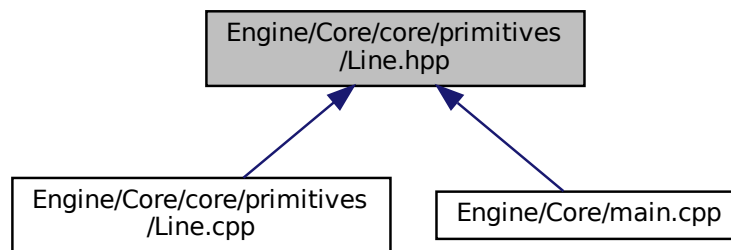
```
#include "../WindowProperties.hpp"
```

```
#include "Square.hpp"
```

Include dependency graph for Line.hpp:



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



## Classes

- class [Primitives::Line](#)

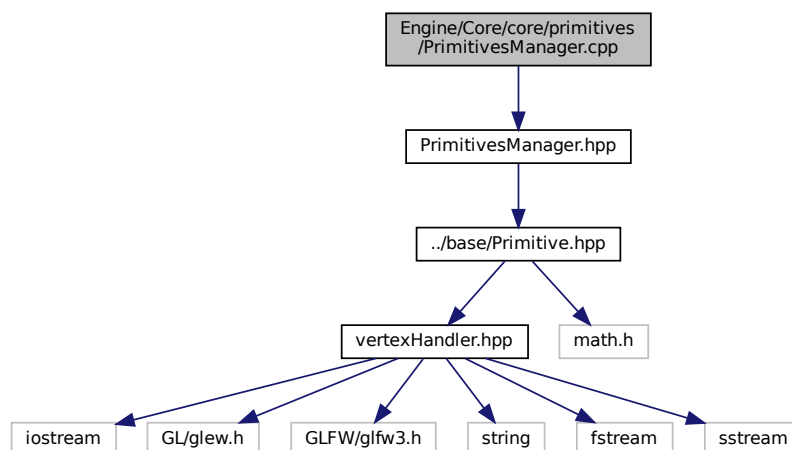
## Namespaces

- [Primitives](#)

## 8.24 Engine/Core/core/primitives/PrimitivesManager.cpp File Reference

```
#include "PrimitivesManager.hpp"
```

Include dependency graph for `PrimitivesManager.cpp`:



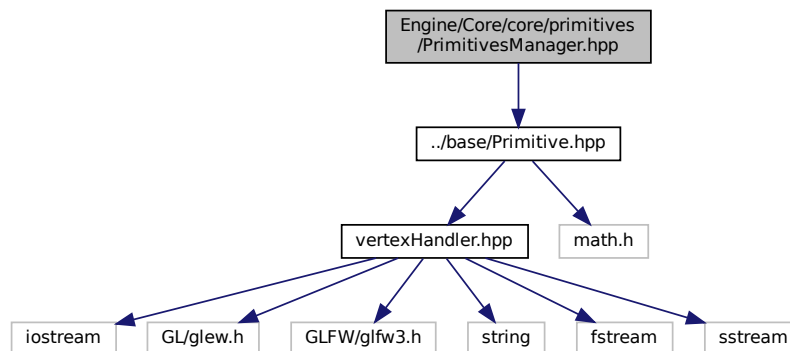
## Namespaces

- [Primitives](#)

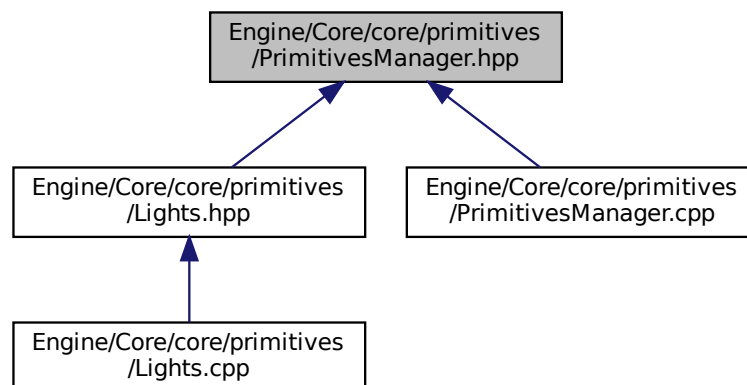
## 8.25 Engine/Core/core/primitives/PrimitivesManager.hpp File Reference

```
#include "../base/Primitive.hpp"
```

Include dependency graph for PrimitivesManager.hpp:



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



## Classes

- class [Primitives::PrimitivesManager](#)

## Namespaces

- [Primitives](#)

## Typedefs

- typedef [Primitive](#) \* [Primitives::PrimitivePtr](#)
- typedef std::vector< [PrimitivePtr](#) > [Primitives::Primitives](#)
- typedef [Primitives](#) & [Primitives::PrimitivesRef](#)

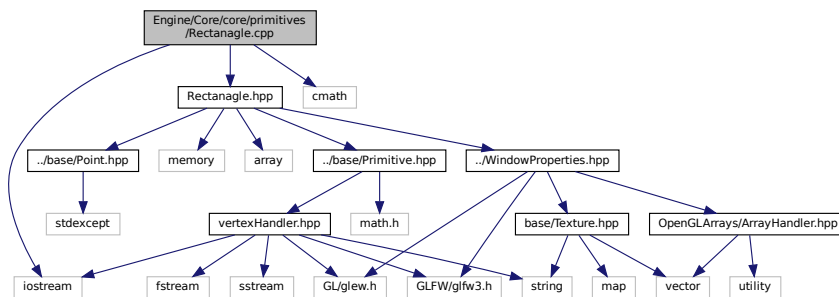
## 8.26 Engine/Core/core/primitives/Rectanagle.cpp File Reference

```
#include "Rectanagle.hpp"
```

```
#include <iostream>
```

```
#include <cmath>
```

Include dependency graph for Rectanagle.cpp:



## Namespaces

- [Primitives](#)

## 8.27 Engine/Core/core/primitives/Rectanagle.hpp File Reference

```
#include "../base/Primitive.hpp"
```

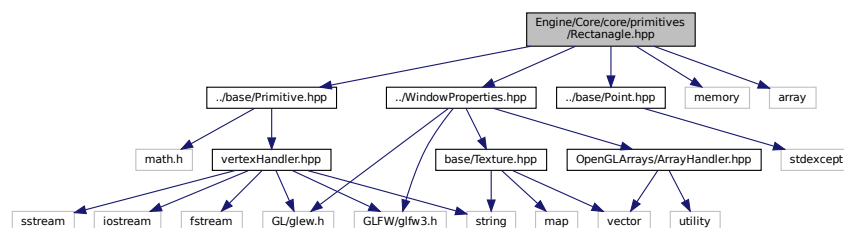
```
#include "../WindowProperties.hpp"
```

```
#include "../base/Point.hpp"
```

```
#include <memory>
```

```
#include <array>
```

Include dependency graph for Rectanagle.hpp:





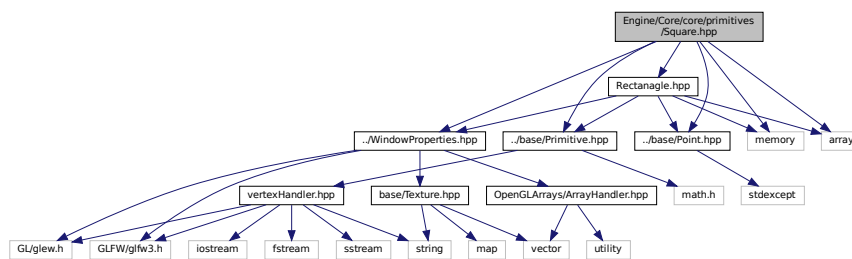
## Namespaces

- [Primitives](#)

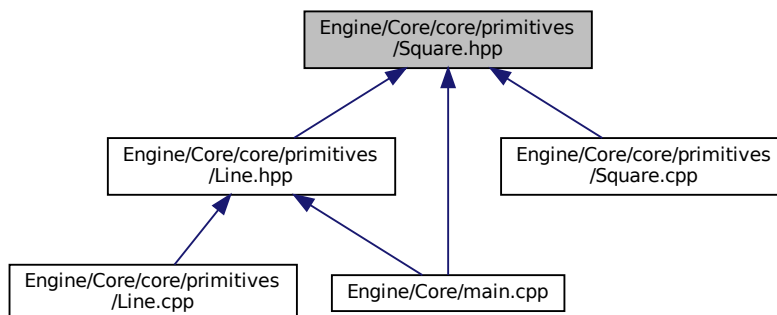
## 8.29 Engine/Core/core/primitives/Square.hpp File Reference

```
#include "../base/Primitive.hpp"
#include "../WindowProperties.hpp"
#include "../base/Point.hpp"
#include "Rectangle.hpp"
#include <memory>
#include <array>
```

Include dependency graph for Square.hpp:



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



## Classes

- class [Primitives::Square](#)

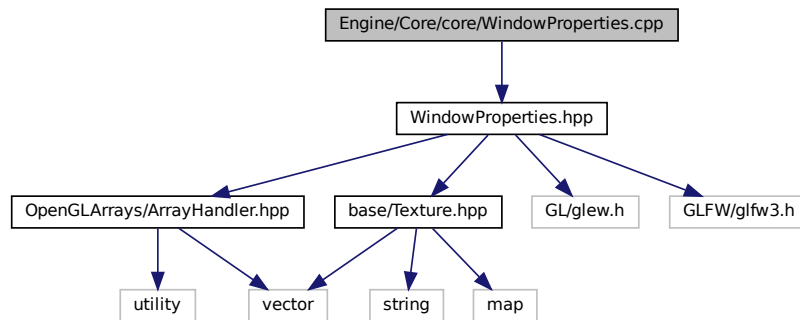
## Namespaces

- [Primitives](#)

## 8.30 Engine/Core/core/WindowProperties.cpp File Reference

```
#include "WindowProperties.hpp"
```

Include dependency graph for WindowProperties.cpp:



## Namespaces

- [Global](#)

## 8.31 Engine/Core/core/WindowProperties.hpp File Reference

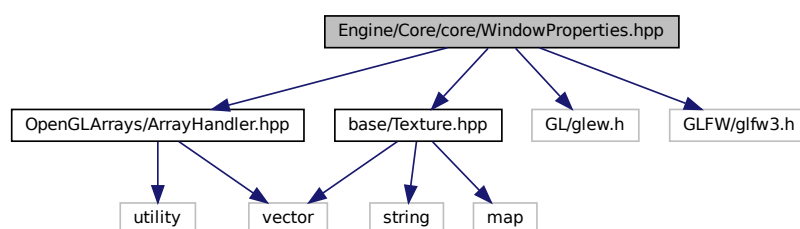
```
#include "OpenGLArrays/ArrayHandler.hpp"
```

```
#include "base/Texture.hpp"
```

```
#include <GL/glew.h>
```

```
#include <GLFW/glfw3.h>
```

Include dependency graph for WindowProperties.hpp:





```

graph TD
    WP_H["Engine/Core/core/WindowProperties.hpp"]
    VH_Cpp["Engine/Core/core/base/VertexHandler.cpp"]
    Rect_H["Engine/Core/core/primitives/Rectangle.hpp"]
    Rect_Cpp["Engine/Core/core/primitives/Rectangle.cpp"]
    Squ_H["Engine/Core/core/primitives/Square.hpp"]
    Squ_Cpp["Engine/Core/core/primitives/Square.cpp"]
    Line_H["Engine/Core/core/primitives/Line.hpp"]
    Line_Cpp["Engine/Core/core/primitives/Line.cpp"]
    Main_Cpp["Engine/Core/main.cpp"]
    WP_Cpp["Engine/Core/core/WindowProperties.cpp"]

    WP_H --> VH_Cpp
    WP_H --> Rect_H
    WP_H --> Rect_Cpp
    Rect_H --> Squ_H
    Rect_H --> Rect_Cpp
    Squ_H --> Line_H
    Squ_H --> Squ_Cpp
    Line_H --> Line_Cpp
    Line_H --> Main_Cpp
    Squ_Cpp --> Main_Cpp
    Main_Cpp --> WP_Cpp
  
```

- class `Global::WindowProperties`

- Global

```
#include <iostream>
#include <GL/glew.h>
#include <GLFW/glfw3.h>
#include <string>
#include <fstream>
#include <sstream>
#include "core/base/vertexHandler.hpp"
#include "core/WindowProperties.hpp"
#include "core/primitives/Square.hpp"
#include "core/primitives/Line.hpp"
#include <vector>
#include <math.h>
#include "core/base/OpenGLStart.hpp"
#include <algorithm>
#include <chrono>
#include <thread>
```

[illegible]

## Macros

- `#define` [STB\\_IMAGE\\_IMPLEMENTATION](#)

## Functions

- void [onKeyPress](#) (int key)
- void [onKeyCallback](#) (GLFWwindow \*window, int key, int status, int action, int mods)
- std::string [readFile](#) (const std::string &fileLoc)
- int [main](#) (int argc, char \*\*argv)

## Variables

- [Primitives::Square](#) \* [player](#)

## 8.32.1 Macro Definition Documentation

### 8.32.1.1 STB\_IMAGE\_IMPLEMENTATION

```
#define STB_IMAGE_IMPLEMENTATION
```

## 8.32.2 Function Documentation

### 8.32.2.1 main()

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

### 8.32.2.2 onKeyCallback()

```
void onKeyCallback (  
    GLFWwindow * window,  
    int key,  
    int status,  
    int action,  
    int mods )
```

### 8.32.2.3 onKeyPress()

```
void onKeyPress (
    int key )
```

### 8.32.2.4 readFile()

```
std::string readFile (
    const std::string & fileLoc )
```

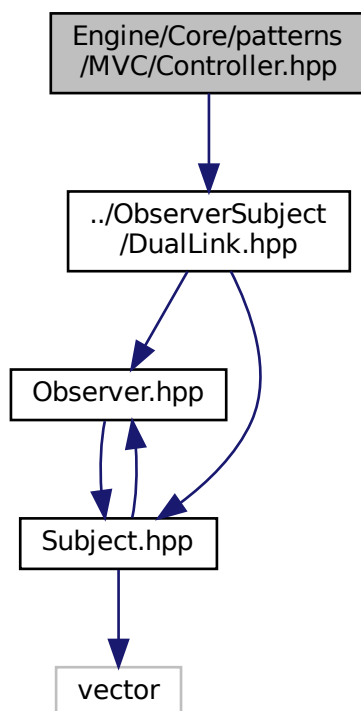
## 8.32.3 Variable Documentation

### 8.32.3.1 player

```
Primitives::Square* player
```

## 8.33 Engine/Core/patterns/MVC/Controller.hpp File Reference

```
#include "../ObserverSubject/DualLink.hpp"
Include dependency graph for Controller.hpp:
```



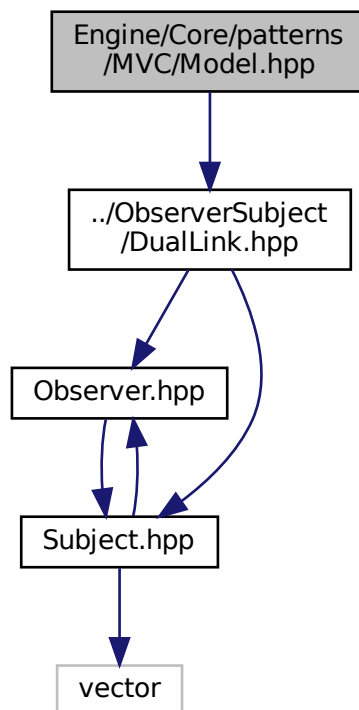
## Classes

- class [Controller](#)

### 8.34 Engine/Core/patterns/MVC/Model.hpp File Reference

```
#include "../ObserverSubject/DualLink.hpp"
```

Include dependency graph for Model.hpp:



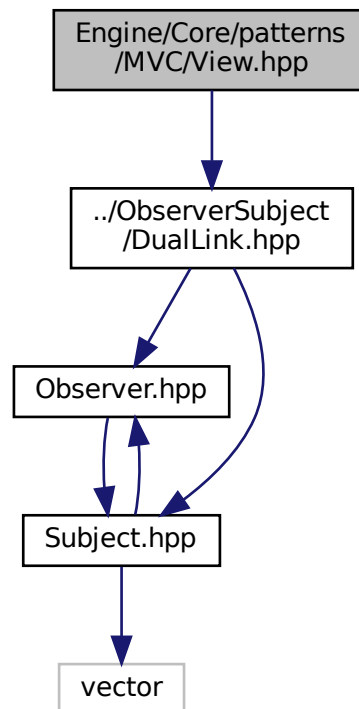
## Classes

- class [Model](#)

### 8.35 Engine/Core/patterns/MVC/View.hpp File Reference

```
#include "../ObserverSubject/DualLink.hpp"
```

Include dependency graph for View.hpp:



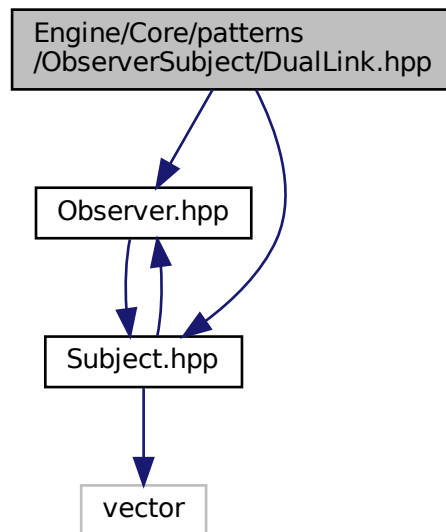
## Classes

- class [View](#)

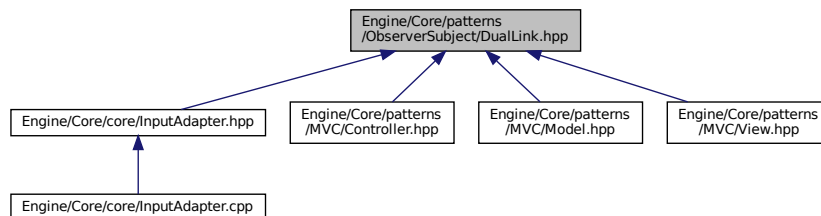
## 8.36 Engine/Core/patterns/ObserverSubject/DualLink.hpp File Reference

```
#include "Observer.hpp"
#include "Subject.hpp"
```

Include dependency graph for DualLink.hpp:



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



## Classes

- class [OSDL::DualLink](#)

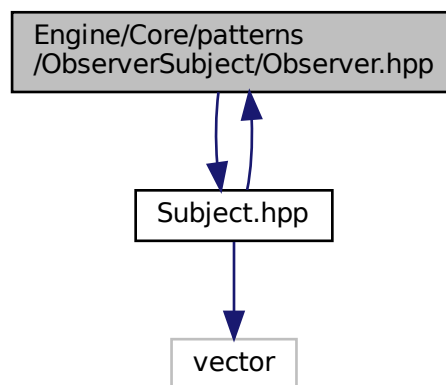
## Namespaces

- [OSDL](#)

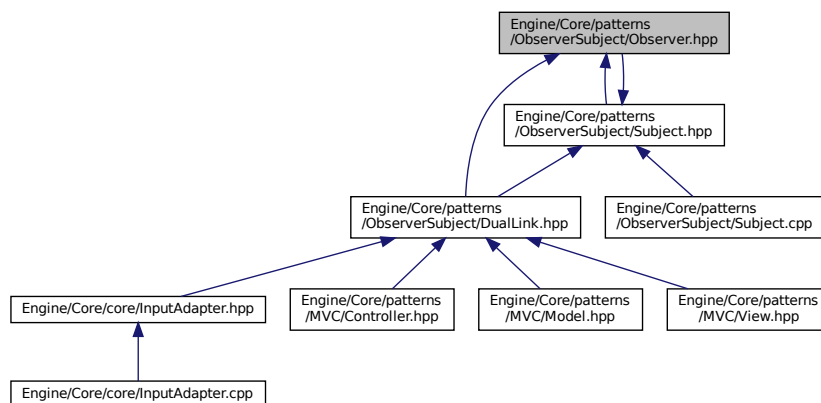
## 8.37 Engine/Core/patterns/ObserverSubject/Observer.hpp File Reference

```
#include "Subject.hpp"
```

Include dependency graph for Observer.hpp:



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



### Classes

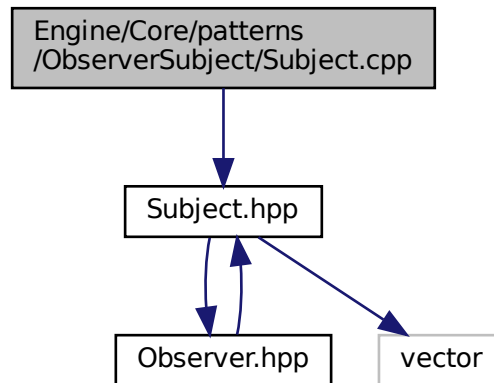
- class [OSDL::Observer](#)

### Namespaces

- [OSDL](#)

### 8.38 Engine/Core/patterns/ObserverSubject/Subject.cpp File Reference

```
#include "Subject.hpp"  
Include dependency graph for Subject.cpp:
```

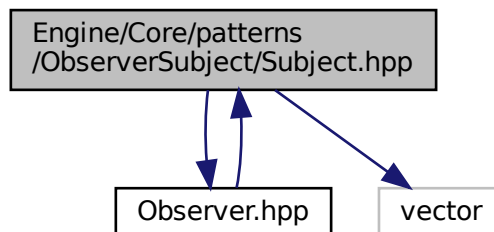


#### Namespaces

- [OSDL](#)

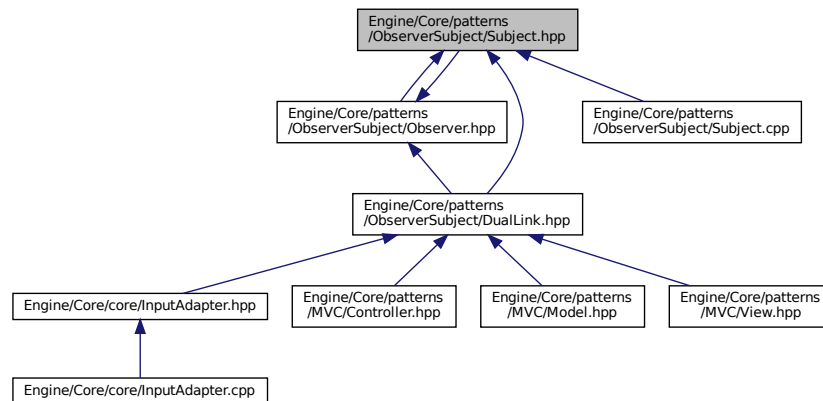
### 8.39 Engine/Core/patterns/ObserverSubject/Subject.hpp File Reference

```
#include "Observer.hpp"  
#include <vector>  
Include dependency graph for Subject.hpp:
```





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



## Classes

- class [OSDL::Subject](#)

## Namespaces

- [OSDL](#)

## 8.40 README.md File Reference



# Index

- ~ArrayHandler
  - GPU::ArrayHandler, 29
- ~Controller
  - Controller, 37
- ~DirectLight
  - Lights::DirectLight, 39
- ~DualLink
  - OSDL::DualLink, 45
- ~GlobalLight
  - Lights::GlobalLight, 47
- ~LightManager
  - Lights::LightManager, 55
- ~Model
  - Model, 65
- ~Observer
  - OSDL::Observer, 66
- ~Primitive
  - Primitive, 71
- ~PrimitivesManager
  - Primitives::PrimitivesManager, 74
- ~Rectanagle
  - Primitives::Rectanagle, 79
- ~SphericalLight
  - Lights::SphericalLight, 88
- ~Square
  - Primitives::Square, 94
- ~Textures
  - AEG::Textures, 102
- ~VertexHandler
  - VertexHandler, 106
- ~View
  - View, 109
- ~WindowProperties
  - Global::WindowProperties, 111

## A

- INPUT, 12
- Action
  - INPUT::AdapterHandler, 27
- Adapter
  - INPUT::Adapter, 22
- AdapterHandler
  - INPUT::AdapterHandler, 26
- addDirectLight
  - Lights::LightManager, 55
- addGlobalLight
  - Lights::LightManager, 55, 56
- addObserver
  - OSDL::Subject, 96
- addPrimitive

- Primitives::PrimitivesManager, 74
- addSphericalLight
  - Lights::LightManager, 56
- AEG, 11
- AEG::Texture, 97
  - Clear, 99
  - getLocation, 99
  - getTextureID, 99
  - ID, 100
  - location, 100
  - setTextureID, 99
  - setTextureLocation, 99
  - Texture, 99
  - Textures, 100
- AEG::Textures, 100
  - ~Textures, 102
  - allocateTextureSpace, 102
  - getTextureIndex, 102
  - LoadTexture, 102
  - loadTexture, 102
  - TextureMap, 103
  - Textures, 101
  - unloadTexture, 102, 103
- allocateBuffer
  - GPU::ArrayHandler, 29
- allocateTextureSpace
  - AEG::Textures, 102
- ALPHA
  - Primitives, 20
- alpha
  - Lights::DirectLight, 42
  - Lights::GlobalLight, 49
  - Lights::SphericalLight, 90
  - Primitives::ColorRGBA< T >, 34
- angle
  - Primitives::Rectanagle, 85
- areFreeBuffers
  - GPU::ArrayHandler, 30
- areRepetitions
  - GPU::ArrayHandler, 30
- ArrayHandler
  - GPU::ArrayHandler, 29
  - GPU::GPU\_Ref, 53
- Arrow\_Down
  - INPUT, 13
- Arrow\_Left
  - INPUT, 13
- Arrow\_Right
  - INPUT, 13

- Arrow\_Up
  - INPUT, [13](#)
- B
  - INPUT, [13](#)
- b
  - Lights::DirectLight, [42](#)
  - Lights::GlobalLight, [49](#)
  - Lights::SphericalLight, [90](#)
  - Primitives::Color< T >, [33](#)
- BLUE
  - Primitives, [20](#)
- C
  - INPUT, [13](#)
- calculateMatrixes
  - Primitives::Rectanagle, [79](#)
- calculateVerticesArr
  - Primitives::Rectanagle, [80](#)
- Clear
  - AEG::Texture, [99](#)
- color
  - Primitives::ColorRGBA< T >, [35](#)
- color\_arr
  - Primitives::Rectanagle, [85](#)
- ColorC
  - Primitives, [15](#)
- ColorD
  - Primitives, [15](#)
- ColorF
  - Primitives, [16](#)
- ColorI
  - Primitives, [16](#)
- ColorL
  - Primitives, [16](#)
- ColorLL
  - Primitives, [16](#)
- ColorRGBAC
  - Primitives, [16](#)
- ColorRGBAD
  - Primitives, [16](#)
- ColorRGBAf
  - Primitives, [16](#)
- ColorRGBAI
  - Primitives, [16](#)
- ColorRGBAL
  - Primitives, [17](#)
- ColorRGBALL
  - Primitives, [17](#)
- ColorRGBAS
  - Primitives, [17](#)
- ColorRGBAU
  - Primitives, [17](#)
- ColorRGBAUl
  - Primitives, [17](#)
- ColorRGBAUll
  - Primitives, [17](#)
- ColorS
  - Primitives, [17](#)
- colors\_array
  - Primitives::Rectanagle, [79](#)
- ColorType
  - Primitives, [19](#)
- ColorU
  - Primitives, [17](#)
- ColorUL
  - Primitives, [18](#)
- ColorULL
  - Primitives, [18](#)
- coneAngle
  - Lights::DirectLight, [42](#)
- coneLength
  - Lights::DirectLight, [42](#)
- coneWidth
  - Lights::DirectLight, [42](#)
- Controller, [35](#)
  - ~Controller, [37](#)
  - Controller, [36](#)
- cos
  - Primitive, [71](#)
- CreateAndCompileShaders
  - OpenGLInstance, [13](#)
- CreateProgramAndLinkShaders
  - OpenGLInstance, [13](#)
- D
  - INPUT, [12](#)
- DirectLight
  - Lights::DirectLight, [39](#)
- directLights
  - Lights::LightManager, [57](#)
- Display
  - Primitive, [71](#)
  - Primitives::Line, [62](#)
  - Primitives::Rectanagle, [80](#)
- DisplayAll
  - Primitives::PrimitivesManager, [74](#)
- draw
  - Lights::LightManager, [56](#)
- drawDirectLights
  - Lights::LightManager, [56](#)
- drawGlobalLights
  - Lights::LightManager, [57](#)
- drawSphericalLights
  - Lights::LightManager, [57](#)
- DualLink
  - OSDL::DualLink, [45](#)
- E
  - INPUT, [12](#)
- EBO
  - GPU::GPU\_Ref, [53](#)
  - VertexHandler, [107](#)
- Engine/Core/core/base/OpenGLStart.cpp, [115](#)
- Engine/Core/core/base/OpenGLStart.hpp, [116](#)
- Engine/Core/core/base/Point.hpp, [117](#)
- Engine/Core/core/base/Primitive.cpp, [118](#)
- Engine/Core/core/base/Primitive.hpp, [119](#)

- Engine/Core/core/base/Texture.cpp, [120](#)
- Engine/Core/core/base/Texture.hpp, [121](#)
- Engine/Core/core/base/vertexHandler.cpp, [122](#)
- Engine/Core/core/base/vertexHandler.hpp, [122](#)
- Engine/Core/core/InputAdapter.cpp, [123](#)
- Engine/Core/core/InputAdapter.hpp, [124](#)
- Engine/Core/core/OpenGLArrays/ArrayHandler.cpp, [125](#)
- Engine/Core/core/OpenGLArrays/ArrayHandler.hpp, [126](#)
- Engine/Core/core/primitives/Light/DirectLight.cpp, [127](#)
- Engine/Core/core/primitives/Light/DirectLight.hpp, [127](#)
- Engine/Core/core/primitives/Light/GlobalLight.cpp, [128](#)
- Engine/Core/core/primitives/Light/GlobalLight.hpp, [128](#)
- Engine/Core/core/primitives/Light/SphericalLight.cpp, [129](#)
- Engine/Core/core/primitives/Light/SphericalLight.hpp, [130](#)
- Engine/Core/core/primitives/Lights.cpp, [130](#)
- Engine/Core/core/primitives/Lights.hpp, [131](#)
- Engine/Core/core/primitives/Line.cpp, [132](#)
- Engine/Core/core/primitives/Line.hpp, [132](#)
- Engine/Core/core/primitives/PrimitivesManager.cpp, [133](#)
- Engine/Core/core/primitives/PrimitivesManager.hpp, [134](#)
- Engine/Core/core/primitives/Rectanagle.cpp, [135](#)
- Engine/Core/core/primitives/Rectanagle.hpp, [135](#)
- Engine/Core/core/primitives/Square.cpp, [136](#)
- Engine/Core/core/primitives/Square.hpp, [137](#)
- Engine/Core/core/WindowProperties.cpp, [138](#)
- Engine/Core/core/WindowProperties.hpp, [138](#)
- Engine/Core/main.cpp, [139](#)
- Engine/Core/patterns/MVC/Controller.hpp, [141](#)
- Engine/Core/patterns/MVC/Model.hpp, [142](#)
- Engine/Core/patterns/MVC/View.hpp, [142](#)
- Engine/Core/patterns/ObserverSubject/DualLink.hpp, [143](#)
- Engine/Core/patterns/ObserverSubject/Observer.hpp, [145](#)
- Engine/Core/patterns/ObserverSubject/Subject.cpp, [146](#)
- Engine/Core/patterns/ObserverSubject/Subject.hpp, [146](#)
- ERROR
  - INPUT, [12](#)
  - Primitives, [20](#)
- F
  - INPUT, [12](#)
- G
  - INPUT, [12](#)
- g
  - Lights::DirectLight, [42](#)
  - Lights::GlobalLight, [50](#)
  - Lights::SphericalLight, [90](#)
  - Primitives::Color< T >, [33](#)
- getAction
  - INPUT::AdapterHandler, [26](#)
- getAH
  - Global::WindowProperties, [111](#)
- getAlpha
  - Lights::DirectLight, [39](#)
  - Lights::GlobalLight, [47](#)
  - Lights::SphericalLight, [88](#)
- getB
  - Lights::DirectLight, [40](#)
  - Lights::GlobalLight, [48](#)
  - Lights::SphericalLight, [88](#)
- getConeAngle
  - Lights::DirectLight, [40](#)
- getConeLength
  - Lights::DirectLight, [40](#)
- getConeWidth
  - Lights::DirectLight, [40](#)
- getEBO
  - GPU::GPU\_Ref, [52](#)
  - VertexHandler, [106](#)
- getFirstAvailavleBuffer
  - GPU::ArrayHandler, [31](#)
- getG
  - Lights::DirectLight, [40](#)
  - Lights::GlobalLight, [48](#)
  - Lights::SphericalLight, [89](#)
- getInstance
  - Global::WindowProperties, [112](#)
  - INPUT::AdapterHandler, [27](#)
- getKey
  - INPUT::AdapterHandler, [27](#)
- getLocation
  - AEG::Texture, [99](#)
- getLumen
  - Lights::DirectLight, [40](#)
  - Lights::GlobalLight, [48](#)
  - Lights::SphericalLight, [89](#)
- getPrimitives
  - Primitives::PrimitivesManager, [74](#)
- getPrimitivesNear
  - Primitives::PrimitivesManager, [74](#)
- getR
  - Lights::DirectLight, [40](#)
  - Lights::GlobalLight, [48](#)
  - Lights::SphericalLight, [89](#)
- getRotation
  - Primitives::Rectanagle, [80](#)
- getTexture
  - VertexHandler, [106](#)
- getTextureID
  - AEG::Texture, [99](#)
- getTextureIndex
  - AEG::Textures, [102](#)
- getTextures
  - Global::WindowProperties, [112](#)
- getVAO
  - GPU::GPU\_Ref, [52](#)
  - VertexHandler, [106](#)

- getVBO
  - GPU::GPU\_Ref, 52
  - VertexHandler, 106
- getWindowHeight
  - Global::WindowProperties, 112
- getWidth
  - Global::WindowProperties, 112
- getX
  - Lights::DirectLight, 40
  - Lights::GlobalLight, 48
  - Lights::SphericalLight, 89
  - Primitives::Rectanagle, 80
- getY
  - Lights::DirectLight, 41
  - Lights::GlobalLight, 48
  - Lights::SphericalLight, 89
  - Primitives::Rectanagle, 80
- Global, 11
- Global::WindowProperties, 110
  - ~WindowProperties, 111
  - getAH, 111
  - getInstance, 112
  - getTextures, 112
  - getWindowHeight, 112
  - getWidth, 112
  - operator=, 112
  - setWindowHeight, 112
  - setWidth, 112
  - windowHeight, 113
  - WindowProperties, 111
  - windowWidth, 113
- GlobalLight
  - Lights::GlobalLight, 47
- globalLights
  - Lights::LightManager, 58
- GPU, 11
- GPU::ArrayHandler, 28
  - ~ArrayHandler, 29
  - allocateBuffer, 29
  - areFreeBuffers, 30
  - areRepetitions, 30
  - ArrayHandler, 29
  - getFirstAvailavleBuffer, 31
  - GPUSpace, 31
  - releaseBuffer, 31
- GPU::GPU\_Ref, 51
  - ArrayHandler, 53
  - EBO, 53
  - getEBO, 52
  - getVAO, 52
  - getVBO, 52
  - GPU\_Ref, 52
  - Release, 52
  - Take, 52
  - Taken, 53
  - taken, 53
  - VAO, 53
  - VBO, 53
- GPU\_Ref
  - GPU::GPU\_Ref, 52
- GPUSpace
  - GPU::ArrayHandler, 31
- GREEN
  - Primitives, 20
- H
  - INPUT, 13
- height
  - Primitives::Rectanagle, 85
- Hold
  - INPUT::AdapterHandler, 26
- I
  - INPUT, 12
- ID
  - AEG::Texture, 100
- inc
  - Primitives::Rectanagle, 85
- InitWindow
  - OpenGLInstance, 14
- INPUT, 11
  - A, 12
  - Arrow\_Down, 13
  - Arrow\_Left, 13
  - Arrow\_Right, 13
  - Arrow\_Up, 13
  - B, 13
  - C, 13
  - D, 12
  - E, 12
  - ERROR, 12
  - F, 12
  - G, 12
  - H, 13
  - I, 12
  - J, 13
  - K, 13
  - Key\_0, 12
  - Key\_1, 12
  - Key\_2, 12
  - Key\_3, 12
  - Key\_4, 12
  - Key\_5, 12
  - Key\_6, 12
  - Key\_7, 12
  - Key\_8, 12
  - Key\_9, 12
  - KeyBoardKey, 12
  - L, 13
  - M, 13
  - N, 13
  - O, 12
  - P, 12
  - Q, 12
  - R, 12
  - S, 12
  - T, 12

- U, [12](#)
- V, [13](#)
- W, [12](#)
- X, [13](#)
- Y, [12](#)
- Z, [13](#)
- INPUT::Adapter, [21](#)
  - Adapter, [22](#)
  - isKeyHold, [23](#)
  - isKeyPressed, [23](#)
  - keyHold, [23](#)
  - keyPressed, [23](#)
  - Notify, [23](#)
- INPUT::AdapterHandler, [24](#)
  - Action, [27](#)
  - AdapterHandler, [26](#)
  - getAction, [26](#)
  - getInstance, [27](#)
  - getKey, [27](#)
  - Hold, [26](#)
  - key, [28](#)
  - KeyActionType, [26](#)
  - onKeyUpdate, [27](#)
  - operator=, [27](#)
  - Press\_down, [26](#)
  - Press\_Up, [26](#)
  - UpdateKey, [27](#)
- isKeyHold
  - INPUT::Adapter, [23](#)
- isKeyPressed
  - INPUT::Adapter, [23](#)
- isNear
  - Primitive, [71](#)
  - Primitives::Line, [62](#)
  - Primitives::Rectanagle, [80](#)
- isVisible
  - Primitive, [71](#)
  - Primitives::Line, [62](#)
  - Primitives::Rectanagle, [81](#)
- J
  - INPUT, [13](#)
- K
  - INPUT, [13](#)
- key
  - INPUT::AdapterHandler, [28](#)
- Key\_0
  - INPUT, [12](#)
- Key\_1
  - INPUT, [12](#)
- Key\_2
  - INPUT, [12](#)
- Key\_3
  - INPUT, [12](#)
- Key\_4
  - INPUT, [12](#)
- Key\_5
  - INPUT, [12](#)
- Key\_6
  - INPUT, [12](#)
- Key\_7
  - INPUT, [12](#)
- Key\_8
  - INPUT, [12](#)
- Key\_9
  - INPUT, [12](#)
- KeyActionType
  - INPUT::AdapterHandler, [26](#)
- KeyboardKey
  - INPUT, [12](#)
- keyHold
  - INPUT::Adapter, [23](#)
- keyPressed
  - INPUT::Adapter, [23](#)
- L
  - INPUT, [13](#)
- LightManager
  - Lights::LightManager, [55](#)
- Lights, [13](#)
- Lights::DirectLight, [37](#)
  - ~DirectLight, [39](#)
  - alpha, [42](#)
  - b, [42](#)
  - coneAngle, [42](#)
  - coneLength, [42](#)
  - coneWidth, [42](#)
  - DirectLight, [39](#)
  - g, [42](#)
  - getAlpha, [39](#)
  - getB, [40](#)
  - getConeAngle, [40](#)
  - getConeLength, [40](#)
  - getConeWidth, [40](#)
  - getG, [40](#)
  - getLumen, [40](#)
  - getR, [40](#)
  - getX, [40](#)
  - getY, [41](#)
  - lumen, [42](#)
  - Move, [41](#)
  - r, [42](#)
  - setAlpha, [41](#)
  - setColor, [41](#)
  - setLumen, [41](#)
  - setPosition, [41](#)
  - x, [43](#)
  - y, [43](#)
- Lights::GlobalLight, [46](#)
  - ~GlobalLight, [47](#)
  - alpha, [49](#)
  - b, [49](#)
  - g, [50](#)
  - getAlpha, [47](#)
  - getB, [48](#)
  - getG, [48](#)
  - getLumen, [48](#)

- getR, 48
- getX, 48
- getY, 48
- GlobalLight, 47
- lumen, 50
- Move, 48
- r, 50
- setAlpha, 49
- setColor, 49
- setLumen, 49
- setPosition, 49
- x, 50
- y, 50
- Lights::LightManager, 54
  - ~LightManager, 55
  - addDirectLight, 55
  - addGlobalLight, 55, 56
  - addSphericalLight, 56
  - directLights, 57
  - draw, 56
  - drawDirectLights, 56
  - drawGlobalLights, 57
  - drawSphericalLights, 57
  - globalLights, 58
  - LightManager, 55
  - removeDirectLight, 57
  - removeGlobalLight, 57
  - removeSphericalLight, 57
  - sphericalLights, 58
- Lights::SphericalLight, 87
  - ~SphericalLight, 88
  - alpha, 90
  - b, 90
  - g, 90
  - getAlpha, 88
  - getB, 88
  - getG, 89
  - getLumen, 89
  - getR, 89
  - getX, 89
  - getY, 89
  - lumen, 91
  - Move, 89
  - r, 91
  - setAlpha, 89
  - setColor, 90
  - setLumen, 90
  - setPosition, 90
  - SphericalLight, 88
  - x, 91
  - y, 91
- LINE
  - Primitive, 70
- Line
  - Primitives::Line, 61
- LoadTexture
  - AEG::Textures, 102
- loadTexture
  - AEG::Textures, 102
- location
  - AEG::Texture, 100
- lumen
  - Lights::DirectLight, 42
  - Lights::GlobalLight, 50
  - Lights::SphericalLight, 91
- M
  - INPUT, 13
- M\_PI
  - Primitive.hpp, 120
- main
  - main.cpp, 140
- main.cpp
  - main, 140
  - onKeyCallback, 140
  - onKeyPress, 140
  - player, 141
  - readFile, 141
  - STB\_IMAGE\_IMPLEMENTATION, 140
- Model, 63
  - ~Model, 65
  - Model, 64
- Move
  - Lights::DirectLight, 41
  - Lights::GlobalLight, 48
  - Lights::SphericalLight, 89
  - Primitives::Rectanagle, 81
- N
  - INPUT, 13
- Notify
  - INPUT::Adapter, 23
  - OSDL::Observer, 66
  - OSDL::Subject, 96
- NotifyObserver
  - OSDL::Subject, 97
- O
  - INPUT, 12
- Observer
  - OSDL::Observer, 66
- observers
  - OSDL::Subject, 97
- onKeyCallback
  - main.cpp, 140
- onKeyPress
  - main.cpp, 140
- onKeyUpUpdate
  - INPUT::AdapterHandler, 27
- OpenGLInstance, 13
  - CreateAndCompileShaders, 13
  - CreateProgramAndLinkShaders, 13
  - InitWindow, 14
  - setClearColor, 14
- operator=
  - Global::WindowProperties, 112
  - INPUT::AdapterHandler, 27



- Primitive, 71
- Primitives::Line, 62
- Primitives::Rectanagle, 81
- Primitives::Square, 94
- VertexHandler, 107
- operator[]
  - Primitives::Color< T >, 33
  - Primitives::ColorRGBA< T >, 34
- OSDL, 14
- OSDL::DualLink, 43
  - ~DualLink, 45
  - DualLink, 45
- OSDL::Observer, 65
  - ~Observer, 66
  - Notify, 66
  - Observer, 66
- OSDL::Subject, 95
  - addObserver, 96
  - Notify, 96
  - NotifyObserver, 97
  - observers, 97
  - removeObserver, 97
- P
  - INPUT, 12
- player
  - main.cpp, 141
- PointC
  - Primitives, 18
- PointD
  - Primitives, 18
- PointF
  - Primitives, 18
- PointI
  - Primitives, 18
- PointL
  - Primitives, 18
- PointLL
  - Primitives, 18
- PointS
  - Primitives, 19
- PointU
  - Primitives, 19
- PointUL
  - Primitives, 19
- PointULL
  - Primitives, 19
- Press\_down
  - INPUT::AdapterHandler, 26
- Press\_Up
  - INPUT::AdapterHandler, 26
- prim\_type
  - Primitive, 70
- Primitive, 68
  - ~Primitive, 71
  - cos, 71
  - Display, 71
  - isNear, 71
  - isVisible, 71
- LINE, 70
  - operator=, 71
  - prim\_type, 70
  - Primitive, 70
  - RECTANAGLE, 70
  - sin, 72
  - SQUARE, 70
  - TRIANGLE, 70
  - TYPE, 72
- Primitive.hpp
  - M\_PI, 120
- PrimitivePtr
  - Primitives, 19
- Primitives, 14
  - ALPHA, 20
  - BLUE, 20
  - ColorC, 15
  - ColorD, 15
  - ColorF, 16
  - ColorI, 16
  - ColorL, 16
  - ColorLL, 16
  - ColorRGBAC, 16
  - ColorRGBAD, 16
  - ColorRGBAF, 16
  - ColorRGBAI, 16
  - ColorRGBAL, 17
  - ColorRGBALL, 17
  - ColorRGBAS, 17
  - ColorRGBAU, 17
  - ColorRGBAUL, 17
  - ColorRGBUULL, 17
  - ColorS, 17
  - ColorType, 19
  - ColorU, 17
  - ColorUL, 18
  - ColorULL, 18
  - ERROR, 20
  - GREEN, 20
  - PointC, 18
  - PointD, 18
  - PointF, 18
  - PointI, 18
  - PointL, 18
  - PointLL, 18
  - PointS, 19
  - PointU, 19
  - PointUL, 19
  - PointULL, 19
  - PrimitivePtr, 19
  - Primitives, 19
  - PrimitivesRef, 19
  - RED, 20
- primitives
  - Primitives::PrimitivesManager, 75
- Primitives::Color< T >, 32
  - b, 33
  - g, 33

- operator[], 33
- r, 33
- Primitives::ColorRGBA< T >, 34
  - alpha, 34
  - color, 35
  - operator[], 34
- Primitives::Line, 58
  - Display, 62
  - isNear, 62
  - isVisible, 62
  - Line, 61
  - operator=, 62
  - rectanagles, 63
  - setTexture, 62
- Primitives::Point2D< T >, 67
  - x, 67
  - y, 67
- Primitives::PrimitivesManager, 72
  - ~PrimitivesManager, 74
  - addPrimitive, 74
  - DisplayAll, 74
  - getPrimitives, 74
  - getPrimitivesNear, 74
  - primitives, 75
  - PrimitivesManager, 73
  - removeAllPrimitives, 74
  - removePrimitive, 74, 75
- Primitives::Rectanagle, 75
  - ~Rectanagle, 79
  - angle, 85
  - calculateMatrixes, 79
  - calculateVerticesArr, 80
  - color\_arr, 85
  - colors\_array, 79
  - Display, 80
  - getRotation, 80
  - getX, 80
  - getY, 80
  - height, 85
  - inc, 85
  - isNear, 80
  - isVisible, 81
  - Move, 81
  - operator=, 81
  - Rectanagle, 79
  - rotate, 82
  - rotateVertices, 82
  - setAlpha, 82
  - setB, 82
  - setColor, 82
  - setG, 83
  - setHeight, 83
  - setPosition, 83
  - setR, 84
  - setRotation, 84
  - setTexture, 84
  - SetVerticeColor, 84
  - setWidth, 85
  - verticies\_arr, 85
  - verticies\_array, 79
  - width, 86
  - window, 86
  - x, 86
  - y, 86
- Primitives::Square, 92
  - ~Square, 94
  - operator=, 94
  - size, 94
  - Square, 94
- PrimitivesManager
  - Primitives::PrimitivesManager, 73
- PrimitivesRef
  - Primitives, 19
- Q
  - INPUT, 12
- R
  - INPUT, 12
- r
  - Lights::DirectLight, 42
  - Lights::GlobalLight, 50
  - Lights::SphericalLight, 91
  - Primitives::Color< T >, 33
- readFile
  - main.cpp, 141
- README.md, 147
- RECTANAGLE
  - Primitive, 70
- Rectanagle
  - Primitives::Rectanagle, 79
- rectanagles
  - Primitives::Line, 63
- RED
  - Primitives, 20
- Release
  - GPU::GPU\_Ref, 52
- releaseBuffer
  - GPU::ArrayHandler, 31
- removeAllPrimitives
  - Primitives::PrimitivesManager, 74
- removeDirectLight
  - Lights::LightManager, 57
- removeGlobalLight
  - Lights::LightManager, 57
- removeObserver
  - OSDL::Subject, 97
- removePrimitive
  - Primitives::PrimitivesManager, 74, 75
- removeSphericalLight
  - Lights::LightManager, 57
- rotate
  - Primitives::Rectanagle, 82
- rotateVertices
  - Primitives::Rectanagle, 82
- S

- INPUT, [12](#)
- setAlpha
  - Lights::DirectLight, [41](#)
  - Lights::GlobalLight, [49](#)
  - Lights::SphericalLight, [89](#)
  - Primitives::Rectanagle, [82](#)
- setB
  - Primitives::Rectanagle, [82](#)
- setClearColor
  - OpenGLInstance, [14](#)
- setColor
  - Lights::DirectLight, [41](#)
  - Lights::GlobalLight, [49](#)
  - Lights::SphericalLight, [90](#)
  - Primitives::Rectanagle, [82](#)
- setG
  - Primitives::Rectanagle, [83](#)
- setHeight
  - Primitives::Rectanagle, [83](#)
- setLumen
  - Lights::DirectLight, [41](#)
  - Lights::GlobalLight, [49](#)
  - Lights::SphericalLight, [90](#)
- setPosition
  - Lights::DirectLight, [41](#)
  - Lights::GlobalLight, [49](#)
  - Lights::SphericalLight, [90](#)
  - Primitives::Rectanagle, [83](#)
- setR
  - Primitives::Rectanagle, [84](#)
- setRotation
  - Primitives::Rectanagle, [84](#)
- setTexture
  - Primitives::Line, [62](#)
  - Primitives::Rectanagle, [84](#)
- setTextureData
  - VertexHandler, [107](#)
- setTextureID
  - AEG::Texture, [99](#)
- setTextureLocation
  - AEG::Texture, [99](#)
- setVertexHandler
  - VertexHandler, [107](#)
- SetVerticeColor
  - Primitives::Rectanagle, [84](#)
- setWidth
  - Primitives::Rectanagle, [85](#)
- setWindowHeight
  - Global::WindowProperties, [112](#)
- setWindowWidth
  - Global::WindowProperties, [112](#)
- sin
  - Primitive, [72](#)
- size
  - Primitives::Square, [94](#)
- SphericalLight
  - Lights::SphericalLight, [88](#)
- sphericalLights

- Lights::LightManager, [58](#)
- SQUARE
  - Primitive, [70](#)
- Square
  - Primitives::Square, [94](#)
- STB\_IMAGE\_IMPLEMENTATION
  - main.cpp, [140](#)
  - Texture.cpp, [121](#)
- T
  - INPUT, [12](#)
- Take
  - GPU::GPU\_Ref, [52](#)
- Taken
  - GPU::GPU\_Ref, [53](#)
- taken
  - GPU::GPU\_Ref, [53](#)
- Texture
  - AEG::Texture, [99](#)
- texture
  - VertexHandler, [107](#)
- Texture.cpp
  - STB\_IMAGE\_IMPLEMENTATION, [121](#)
- TextureMap
  - AEG::Textures, [103](#)
- Textures
  - AEG::Texture, [100](#)
  - AEG::Textures, [101](#)
- TRIANGLE
  - Primitive, [70](#)
- TYPE
  - Primitive, [72](#)
- U
  - INPUT, [12](#)
- unloadTexture
  - AEG::Textures, [102](#), [103](#)
- UpdateKey
  - INPUT::AdapterHandler, [27](#)
- V
  - INPUT, [13](#)
- VAO
  - GPU::GPU\_Ref, [53](#)
  - VertexHandler, [107](#)
- VBO
  - GPU::GPU\_Ref, [53](#)
  - VertexHandler, [108](#)
- VertexHandler, [103](#)
  - ~VertexHandler, [106](#)
  - EBO, [107](#)
  - getEBO, [106](#)
  - getTexture, [106](#)
  - getVAO, [106](#)
  - getVBO, [106](#)
  - operator=, [107](#)
  - setTextureData, [107](#)
  - setVertexHandler, [107](#)
  - texture, [107](#)

- VAO, [107](#)
- VBO, [108](#)
- VertexHandler, [106](#)
- vertices\_arr
  - Primitives::Rectanagle, [85](#)
- vertices\_array
  - Primitives::Rectanagle, [79](#)
- View, [108](#)
  - ~View, [109](#)
  - View, [109](#)
- W
  - INPUT, [12](#)
- width
  - Primitives::Rectanagle, [86](#)
- window
  - Primitives::Rectanagle, [86](#)
- windowHeight
  - Global::WindowProperties, [113](#)
- WindowProperties
  - Global::WindowProperties, [111](#)
- windowWidth
  - Global::WindowProperties, [113](#)
- X
  - INPUT, [13](#)
- x
  - Lights::DirectLight, [43](#)
  - Lights::GlobalLight, [50](#)
  - Lights::SphericalLight, [91](#)
  - Primitives::Point2D< T >, [67](#)
  - Primitives::Rectanagle, [86](#)
- Y
  - INPUT, [12](#)
- y
  - Lights::DirectLight, [43](#)
  - Lights::GlobalLight, [50](#)
  - Lights::SphericalLight, [91](#)
  - Primitives::Point2D< T >, [67](#)
  - Primitives::Rectanagle, [86](#)
- Z
  - INPUT, [13](#)