

## 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 Hierarchical Index</b>	<b>3</b>
2.1 Class Hierarchy	3
<b>3 Class Index</b>	<b>5</b>
3.1 Class List	5
<b>4 Class Documentation</b>	<b>7</b>
4.1 INPUT::Adapter Class Reference	7
4.2 INPUT::AdapterHandler Class Reference	7
4.3 GPU::ArrayHandler Class Reference	8
4.3.1 Member Function Documentation	9
4.3.1.1 allocateBuffer()	9
4.3.1.2 areFreeBuffers()	9
4.3.1.3 areRepetitions()	10
4.3.1.4 getFirstAvailavleBuffer()	10
4.3.1.5 releaseBuffer()	10
4.4 Primitives::Color< T > Struct Template Reference	11
4.5 Primitives::ColorRGBA< T > Struct Template Reference	11
4.6 Controller Class Reference	12
4.7 Lights::DirectLight Class Reference	12
4.8 OSDL::DualLink Class Reference	13
4.9 Lights::GlobalLight Class Reference	13
4.10 GPU::GPU_Ref Class Reference	14
4.11 Lights::LightManager Class Reference	14
4.12 Primitives::Line Class Reference	15
4.12.1 Constructor & Destructor Documentation	16
4.12.1.1 Line()	16
4.12.2 Member Function Documentation	16
4.12.2.1 Display()	16
4.12.2.2 isVisible()	17
4.13 Model Class Reference	17
4.14 OSDL::Observer Class Reference	17
4.15 Primitives::Point2D< T > Struct Template Reference	18
4.16 Primitive Class Reference	18
4.17 Primitives::PrimitivesManager Class Reference	19
4.18 Primitives::Rectanagle Class Reference	19

4.18.1 Member Function Documentation	21
4.18.1.1 isNear()	21
4.18.1.2 isVisible()	21
4.18.1.3 Move()	22
4.18.1.4 rotate()	22
4.18.1.5 rotateVertices()	22
4.18.1.6 setAlpha()	22
4.18.1.7 setColor()	23
4.18.1.8 setG()	23
4.18.1.9 setHeight()	23
4.18.1.10 setPosition()	24
4.18.1.11 setR()	24
4.18.1.12 setRotation()	24
4.18.1.13 setTexture()	24
4.18.1.14 setWidth()	25
4.19 Lights::SphericalLight Class Reference	25
4.20 Primitives::Square Class Reference	26
4.21 OSDL::Subject Class Reference	26
4.22 AEG::Texture Class Reference	27
4.23 AEG::Textures Class Reference	28
4.24 VertexHandler Class Reference	28
4.25 View Class Reference	29
4.26 Global::WindowProperties Class Reference	29
4.26.1 Constructor & Destructor Documentation	30
4.26.1.1 WindowProperties()	30

<b>Index</b>	<b>31</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

# Hierarchical Index

### 2.1 Class Hierarchy

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

GPU::ArrayHandler . . . . .	8
Primitives::Color< T > . . . . .	11
Primitives::ColorRGBA< T > . . . . .	11
Lights::DirectLight . . . . .	12
Lights::GlobalLight . . . . .	13
GPU::GPU_Ref . . . . .	14
Lights::LightManager . . . . .	14
OSDL::Observer . . . . .	17
INPUT::Adapter . . . . .	7
OSDL::DualLink . . . . .	13
Controller . . . . .	12
Model . . . . .	17
View . . . . .	29
Primitives::Point2D< T > . . . . .	18
Primitives::PrimitivesManager . . . . .	19
Lights::SphericalLight . . . . .	25
OSDL::Subject . . . . .	26
INPUT::AdapterHandler . . . . .	7
OSDL::DualLink . . . . .	13
AEG::Texture . . . . .	27
AEG::Textures . . . . .	28
VertexHandler . . . . .	28
Primitive . . . . .	18
Primitives::Line . . . . .	15
Primitives::Rectanagle . . . . .	19
Primitives::Square . . . . .	26
Global::WindowProperties . . . . .	29





## Chapter 3

# Class Index

### 3.1 Class List

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

INPUT::Adapter	7
INPUT::AdapterHandler	7
GPU::ArrayHandler	8
Primitives::Color< T >	11
Primitives::ColorRGBA< T >	11
Controller	12
Lights::DirectLight	12
OSDL::DualLink	13
Lights::GlobalLight	13
GPU::GPU_Ref	14
Lights::LightManager	14
Primitives::Line	15
Model	17
OSDL::Observer	17
Primitives::Point2D< T >	18
Primitive	18
Primitives::PrimitivesManager	19
Primitives::Rectangle	19
Lights::SphericalLight	25
Primitives::Square	26
OSDL::Subject	26
AEG::Texture	27
AEG::Textures	28
VertexHandler	28
View	29
Global::WindowProperties	29

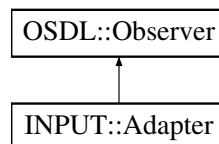


## Chapter 4

# Class Documentation

### 4.1 INPUT::Adapter Class Reference

Inheritance diagram for INPUT::Adapter:



#### Public Member Functions

- void **Notify** (OSDL::Subject \*)
- bool **isKeyPressed** (KeyBoardKey)
- bool **isKeyHold** (KeyBoardKey)

#### Private Attributes

- std::map< KeyBoardKey, bool > **keyPressed**
- std::map< KeyBoardKey, bool > **keyHold**

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

- Engine/Core/core/InputAdapter.hpp
- Engine/Core/core/InputAdapter.cpp

### 4.2 INPUT::AdapterHandler Class Reference

Inheritance 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

- void **UpdateKey** (int)

## Private Attributes

- KeyBoardKey **key**
- KeyActionType **Action**

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

- Engine/Core/core/InputAdapter.hpp
- Engine/Core/core/InputAdapter.cpp

## 4.3 GPU::ArrayHandler Class Reference

### Public Member Functions

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

### Private Member Functions

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

## Private Attributes

- `std::vector< GPU_Ref > GPUSpace`

## 4.3.1 Member Function Documentation

### 4.3.1.1 `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
-------------	-------------------------

### 4.3.1.2 `areFreeBuffers()`

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

Check if any buffers are available.

#### Note

#### Return values

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

#### 4.3.1.3 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 repetiotion
<i>false</i>	if there are no repetitions

#### 4.3.1.4 getFirstAvailavleBuffer()

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

Returns pointer to first Available Buffer.

##### Note

##### Return values

<a href="#"><i>GPU_Ref</i></a>	Pointer to buffer, in not buffers available return nullptr
--------------------------------	--

#### 4.3.1.5 releaseBuffer()

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

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

Note

Exceptions

------	--

Return values

None	
------	--

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

- Engine/Core/core/OpenGLArrays/ArrayHandler.hpp
- Engine/Core/core/OpenGLArrays/ArrayHandler.cpp

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

### Public Member Functions

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

### Public Attributes

- T **r** = 0
- T **g** = 0
- T **b** = 0

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

- Engine/Core/core/base/Point.hpp

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

### Public Member Functions

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

### Public Attributes

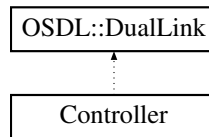
- **Color**< T > **color**
- T **alpha** = 0

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

- Engine/Core/core/base/Point.hpp

## 4.6 Controller Class Reference

Inheritance diagram for Controller:



### Additional Inherited Members

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

- Engine/Core/patterns/MVC/Controller.hpp

## 4.7 Lights::DirectLight Class Reference

### 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** ()

### Private Attributes

- float **x**
- float **y**
- float **lumen**
- float **alpha** = 1.0f
- float **r**
- float **g**
- float **b**
- float **coneAngle**
- float **coneWidth**
- float **coneLength**

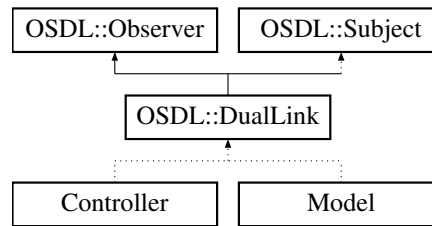
The documentation for this class was generated from the following files:

- Engine/Core/core/primitives/Light/DirectLight.hpp
- Engine/Core/core/primitives/Light/DirectLight.cpp



## 4.8 OSDL::DualLink Class Reference

Inheritance diagram for OSDL::DualLink:



### Additional Inherited Members

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

- Engine/Core/patterns/ObserverSubject/DualLink.hpp

## 4.9 Lights::GlobalLight Class Reference

### 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** ()

### Private Attributes

- float **x**
- float **y**
- float **lumen**
- float **alpha** = 1.0f
- float **r**
- float **g**
- float **b**

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

## 4.10 GPU::GPU\_Ref Class Reference

### 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**

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

- Engine/Core/core/OpenGLArrays/ArrayHandler.hpp
- Engine/Core/core/OpenGLArrays/ArrayHandler.cpp

## 4.11 Lights::LightManager Class Reference

### Public Member Functions

- 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, 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

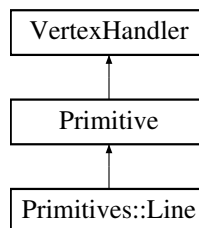
- std::vector< [DirectLight](#) \* > **directLights**
- std::vector< [SphericalLight](#) \* > **sphericalLights**
- std::vector< [GlobalLight](#) \* > **globalLights**

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

- Engine/Core/core/primitives/Lights.hpp
- Engine/Core/core/primitives/Lights.cpp

## 4.12 Primitives::Line Class Reference

Inheritance 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

## Private Attributes

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

## Additional Inherited Members

### 4.12.1 Constructor & Destructor Documentation

#### 4.12.1.1 Line()

```
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

### 4.12.2 Member Function Documentation

#### 4.12.2.1 Display()

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

Display the [Line](#).

Implements [Primitive](#).

#### 4.12.2.2 isVisible()

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

is the [Line](#) visible

##### Returns

true  
false

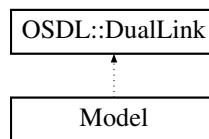
Implements [Primitive](#).

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

- Engine/Core/core/primitives/Line.hpp
- Engine/Core/core/primitives/Line.cpp

## 4.13 Model Class Reference

Inheritance diagram for Model:



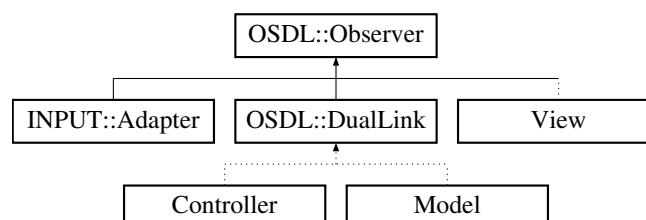
### Additional Inherited Members

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

- Engine/Core/patterns/MVC/Model.hpp

## 4.14 OSDL::Observer Class Reference

Inheritance diagram for OSDL::Observer:



## Public Member Functions

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

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

- Engine/Core/patterns/ObserverSubject/Observer.hpp

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

### Public Attributes

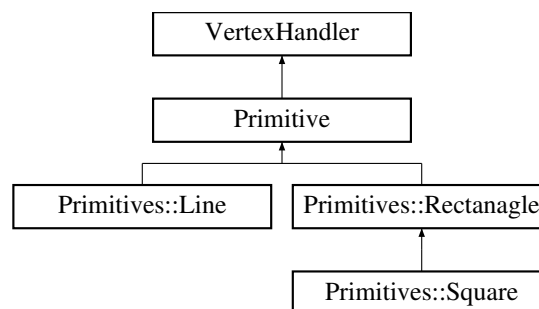
- T **x** = 0
- T **y** = 0

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

- Engine/Core/core/base/Point.hpp

## 4.16 Primitive Class Reference

Inheritance diagram for Primitive:



### Public Member Functions

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

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

- Engine/Core/core/base/Primitive.hpp
- Engine/Core/core/base/Primitive.cpp

## 4.17 Primitives::PrimitivesManager Class Reference

### Public Member Functions

- void **addPrimitive** (Primitive \*)
- void **removePrimitive** (Primitive \*)
- void **removePrimitive** (int)
- void **removeAllPrimitives** ()
- void **DisplayAll** ()
- PrimitivesRef **getPrimitives** ()
- Primitives **getPrimitivesNear** (float, float, float)

### Private Attributes

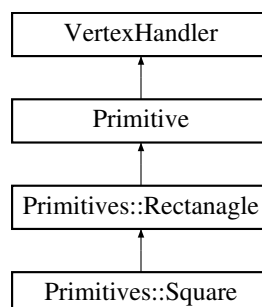
- Primitives **primitives**

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

- Engine/Core/core/primitives/PrimitivesManager.hpp
- Engine/Core/core/primitives/PrimitivesManager.cpp

## 4.18 Primitives::Rectanagle Class Reference

Inheritance diagram for Primitives::Rectanagle:



## Public Member Functions

- **Rectanagle** (float, float, float, float, float=1.0f)
- **Rectanagle** (const [Rectanagle](#) &s)
- [Rectanagle](#) & **operator=** (const [Rectanagle](#) &s)
- 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 **SetVerticeColor** (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**
- vertices\_array **verticies\_arr** = {}
- colors\_array **color\_arr** = {}
- float **angle** = 0.0f
- unsigned int **inc** [6] = {0, 1, 2, 0, 2, 3}

## 4.18.1 Member Function Documentation

### 4.18.1.1 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](#).

### 4.18.1.2 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](#).

#### 4.18.1.3 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

#### 4.18.1.4 rotate()

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

rotate the square relative to the current rotation

##### Parameters

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

#### 4.18.1.5 rotateVertices()

```
Rectanagle::verticies_array Primitives::Rectanagle::rotateVertices ( ) [protected]
```

Rotate the vertices of the square.

##### Parameters

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

#### 4.18.1.6 setAlpha()

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

Set the Alpha value.

## Parameters

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

**4.18.1.7 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

**4.18.1.8 setG()**

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

Set G color value.

## Parameters

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

**4.18.1.9 setHeight()**

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

Set the Height value.

#### 4.18.1.10 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

#### 4.18.1.11 setR()

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

Set R color value.

##### Parameters

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

#### 4.18.1.12 setRotation()

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

Set the rotation of the square.

##### Parameters

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

#### 4.18.1.13 setTexture()

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

Set the Texture object.

## Parameters

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

4.18.1.14 `setWidth()`

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

Set the Width value.

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

- Engine/Core/core/primitives/Rectanagle.hpp
- Engine/Core/core/primitives/Rectanagle.cpp

## 4.19 Lights::SphericalLight Class Reference

## 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** ()

## Private Attributes

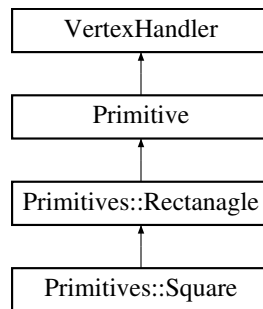
- float **x**
- float **y**
- float **lumen**
- float **alpha** = 1.0f
- float **r**
- float **g**
- float **b**

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

## 4.20 Primitives::Square Class Reference

Inheritance diagram for Primitives::Square:



### Public Member Functions

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

### Private Attributes

- float **size** = 0

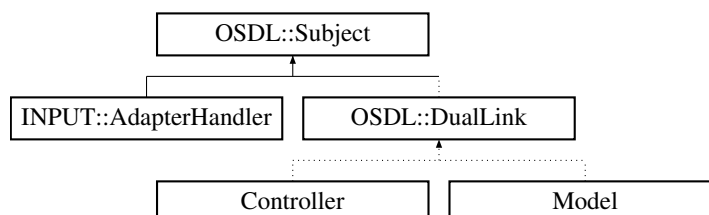
### Additional Inherited Members

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

- Engine/Core/core/primitives/Square.hpp
- Engine/Core/core/primitives/Square.cpp

## 4.21 OSDL::Subject Class Reference

Inheritance diagram for OSDL::Subject:



## Public Member Functions

- void **addObserver** ([Observer](#) &)
- void **Notify** ()
- void **NotifyObserver** ([Observer](#) &)
- void **removeObserver** ([Observer](#) &)

## Private Attributes

- std::vector< [Observer](#) \* > **observers**

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

- Engine/Core/patterns/ObserverSubject/Subject.hpp
- Engine/Core/patterns/ObserverSubject/Subject.cpp

## 4.22 AEG::Texture Class Reference

### 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**

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

- Engine/Core/core/base/Texture.hpp
- Engine/Core/core/base/Texture.cpp

## 4.23 AEG::Textures Class Reference

### Public Member Functions

- 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)

### Private Member Functions

- void **LoadTexture** (int, const std::string &)

### Private Attributes

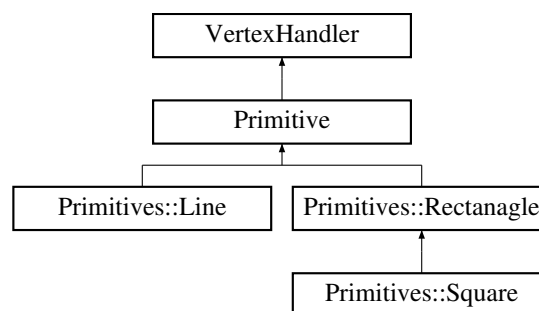
- std::vector< [Texture](#) > **TextureMap**

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

- Engine/Core/core/base/Texture.hpp
- Engine/Core/core/base/Texture.cpp

## 4.24 VertexHandler Class Reference

Inheritance diagram for VertexHandler:



### Public Member Functions

- **VertexHandler** (bool isDefaultPrimitive)
- **VertexHandler** (const [VertexHandler](#) &other)
- [VertexHandler](#) & **operator=** (const [VertexHandler](#) &other)
- 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

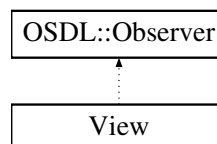
- unsigned int **VBO**
- unsigned int **VAO**
- unsigned int **EBO**
- unsigned int **texture**

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

- Engine/Core/core/base/vertexHandler.hpp
- Engine/Core/core/base/vertexHandler.cpp

## 4.25 View Class Reference

Inheritance diagram for View:



### Additional Inherited Members

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

- Engine/Core/patterns/MVC/View.hpp

## 4.26 Global::WindowProperties Class Reference

### Public Member Functions

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

### Static Public Member Functions

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

## Private Attributes

- int **windowHeight**
- int **windowWidth**

## 4.26.1 Constructor & Destructor Documentation

### 4.26.1.1 WindowProperties()

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

ArrayHandler for All GPU bindings.

#### Note

#### Return values

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

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

- Engine/Core/core/WindowProperties.hpp
- Engine/Core/core/WindowProperties.cpp

# Index

- AEG::Texture, [27](#)
- AEG::Textures, [28](#)
- allocateBuffer
  - GPU::ArrayHandler, [9](#)
- areFreeBuffers
  - GPU::ArrayHandler, [9](#)
- areRepetitions
  - GPU::ArrayHandler, [9](#)
- Controller, [12](#)
- Display
  - Primitives::Line, [16](#)
- getFirstAvailavleBuffer
  - GPU::ArrayHandler, [10](#)
- Global::WindowProperties, [29](#)
  - WindowProperties, [30](#)
- GPU::ArrayHandler, [8](#)
  - allocateBuffer, [9](#)
  - areFreeBuffers, [9](#)
  - areRepetitions, [9](#)
  - getFirstAvailavleBuffer, [10](#)
  - releaseBuffer, [10](#)
- GPU::GPU\_Ref, [14](#)
- INPUT::Adapter, [7](#)
- INPUT::AdapterHandler, [7](#)
- isNear
  - Primitives::Rectanagle, [21](#)
- isVisible
  - Primitives::Line, [16](#)
  - Primitives::Rectanagle, [21](#)
- Lights::DirectLight, [12](#)
- Lights::GlobalLight, [13](#)
- Lights::LightManager, [14](#)
- Lights::SphericalLight, [25](#)
- Line
  - Primitives::Line, [16](#)
- Model, [17](#)
- Move
  - Primitives::Rectanagle, [21](#)
- OSDL::DualLink, [13](#)
- OSDL::Observer, [17](#)
- OSDL::Subject, [26](#)
- Primitive, [18](#)
- Primitives::Color< T >, [11](#)
- Primitives::Line, [15](#)
  - Display, [16](#)
  - isVisible, [16](#)
  - Line, [16](#)
- Primitives::Point2D< T >, [18](#)
- Primitives::PrimitivesManager, [19](#)
- Primitives::Rectanagle, [19](#)
  - isNear, [21](#)
  - isVisible, [21](#)
  - Move, [21](#)
  - rotate, [22](#)
  - rotateVertices, [22](#)
  - setAlpha, [22](#)
  - setColor, [23](#)
  - setG, [23](#)
  - setHeight, [23](#)
  - setPosition, [23](#)
  - setR, [24](#)
  - setRotation, [24](#)
  - setTexture, [24](#)
  - setWidth, [25](#)
- Primitives::Square, [26](#)
- releaseBuffer
  - GPU::ArrayHandler, [10](#)
- rotate
  - Primitives::Rectanagle, [22](#)
- rotateVertices
  - Primitives::Rectanagle, [22](#)
- setAlpha
  - Primitives::Rectanagle, [22](#)
- setColor
  - Primitives::Rectanagle, [23](#)
- setG
  - Primitives::Rectanagle, [23](#)
- setHeight
  - Primitives::Rectanagle, [23](#)
- setPosition
  - Primitives::Rectanagle, [23](#)
- setR
  - Primitives::Rectanagle, [24](#)
- setRotation
  - Primitives::Rectanagle, [24](#)
- setTexture
  - Primitives::Rectanagle, [24](#)
- setWidth
  - Primitives::Rectanagle, [25](#)

VertexHandler, [28](#)

View, [29](#)

WindowProperties

Global::WindowProperties, [30](#)