

## Fast OpenGL Library

Generated by Doxygen 1.8.17



1	<a href="https://github.com/mwawrzkow/FastOGLib/actions/workflows/main.yml">https://github.com/mwawrzkow/FastOGLib/actions/workflows/main.yml</a>		
	<a href="https://github.com/mwawrzkow/FastOGLib/actions/workflows/main.yml/badge.svg">https://github.com/mwawrzkow/FastOGLib/actions/workflows/main.yml/badge.svg</a>		
	<b>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		2
1.2	Contributors		2
2	<b>Hierarchical Index</b>		<b>3</b>
2.1	Class Hierarchy		3
3	<b>Class Index</b>		<b>5</b>
3.1	Class List		5
4	<b>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	Controller Class Reference		11
4.5	Lights::DirectLight Class Reference		11
4.6	OSDL::DualLink Class Reference		12
4.7	Lights::GlobalLight Class Reference		12
4.8	GPU::GPU_Ref Class Reference		13
4.9	Lights::LightManager Class Reference		14
4.10	Primitives::Line Class Reference		14
4.11	Model Class Reference		15
4.12	OSDL::Observer Class Reference		15
4.13	Primitive Class Reference		15
4.14	Primitives::PrimitivesManager Class Reference		16
4.15	Lights::SphericalLight Class Reference		17
4.16	Primitives::Square Class Reference		17
4.17	OSDL::Subject Class Reference		18
4.18	AEG::Texture Class Reference		19
4.19	AEG::Textures Class Reference		20
4.20	vertexHandler Class Reference		20
4.21	View Class Reference		21
4.22	Global::WindowProperties Class Reference		21

4.22.1 Constructor & Destructor Documentation . . . . .	22
4.22.1.1 WindowProperties() . . . . .	22
<b>Index</b>	<b>23</b>

## Chapter 1

`<a href="https://github.com/mwawrzkow/FastOGLib/actions/workflows/main.yml"></a>` **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 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 developement 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
Lights::DirectLight . . . . .	11
Lights::GlobalLight . . . . .	12
GPU::GPU_Ref . . . . .	13
Lights::LightManager . . . . .	14
Primitives::Line . . . . .	14
OSDL::Observer . . . . .	15
INPUT::Adapter . . . . .	7
OSDL::DualLink . . . . .	12
Controller . . . . .	11
Model . . . . .	15
View . . . . .	21
Primitives::PrimitivesManager . . . . .	16
Lights::SphericalLight . . . . .	17
OSDL::Subject . . . . .	18
INPUT::AdapterHandler . . . . .	7
OSDL::DualLink . . . . .	12
AEG::Texture . . . . .	19
AEG::Textures . . . . .	20
vertexHandler . . . . .	20
Primitive . . . . .	15
Primitives::Square . . . . .	17
Global::WindowProperties . . . . .	21





## 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
Controller	11
Lights::DirectLight	11
OSDL::DualLink	12
Lights::GlobalLight	12
GPU::GPU_Ref	13
Lights::LightManager	14
Primitives::Line	14
Model	15
OSDL::Observer	15
Primitive	15
Primitives::PrimitivesManager	16
Lights::SphericalLight	17
Primitives::Square	17
OSDL::Subject	18
AEG::Texture	19
AEG::Textures	20
vertexHandler	20
View	21
Global::WindowProperties	21

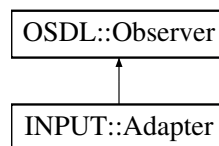


## Chapter 4

# 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

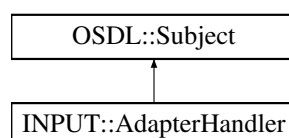
- std::map< KeyBoardKey, bool > **keyPressed**
- std::map< KeyBoardKey, bool > **keyHold**

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 repetition
<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

<i>GPU_Ref</i>	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

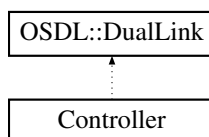
None	
------	--

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 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

## 4.5 Lights::DirectLight Class Reference

### Public Member Functions

- **DirectLight** (float, 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

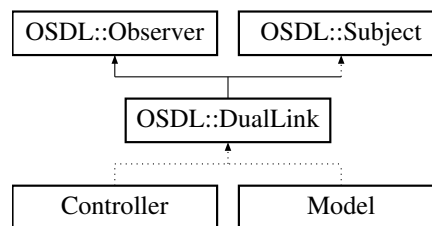
- float **x**
- float **y**
- float **lumen**
- float **alpha** = 1.0f
- float **r**
- float **g**
- float **b**
- float **coneAngle**
- float **coneWidth**
- float **coneLength**

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

## 4.6 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

## 4.7 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

## 4.8 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

## 4.9 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

## 4.10 Primitives::Line Class Reference

### Public Member Functions

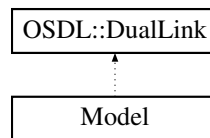
- **Line** (int x0, int y0, int x1, int y1)

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

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

## 4.11 Model Class Reference

Inheritance diagram for Model:



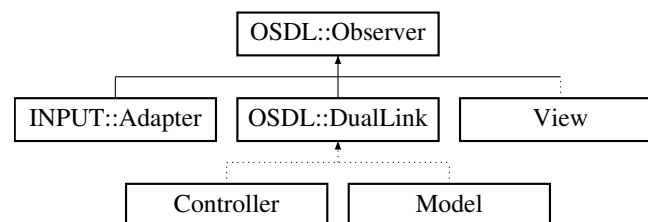
### Additional Inherited Members

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

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

## 4.12 OSDL::Observer Class Reference

Inheritance diagram for OSDL::Observer:



### Public Member Functions

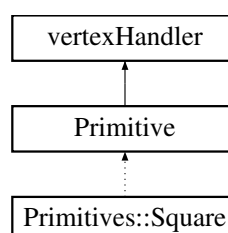
- virtual void **Notify** ([Subject](#) \*)

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

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

## 4.13 Primitive Class Reference

Inheritance diagram for Primitive:



## Public Member Functions

- **Primitive** (prim\_type)
- virtual bool **isVisible** ()=0
- virtual bool **isNear** (float, float, float)=0

## Protected Types

- enum **prim\_type** { RECTANAGLE, TRIANGLE, SQUARE }

## 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

## 4.14 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

## 4.15 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

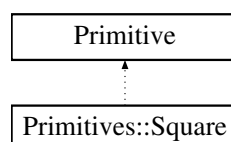
- float **x**
- float **y**
- float **lumen**
- float **alpha** = 1.0f
- float **r**
- float **g**
- float **b**

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

## 4.16 Primitives::Square Class Reference

Inheritance diagram for Primitives::Square:



## Public Member Functions

- **Square** (float, float, float, float)
- void **setPosition** (float, float)
- void **Move** (float, float)
- void **setTexture** (std::string)
- bool **isVisible** () override
- void **Display** ()
- float **getX** ()
- float **getY** ()
- void **setAlpha** (float)
- bool **isNear** (float, float, float) override

## Private Member Functions

- void **calculateMatrixes** ()

## Private Attributes

- [Global::WindowProperties](#) \* **window**
- float **x**
- float **y**
- float **size**
- float **alpha** = 1.0f

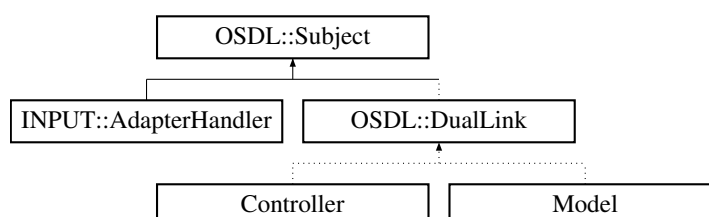
## Additional Inherited Members

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

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

## 4.17 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

## 4.18 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

## 4.19 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

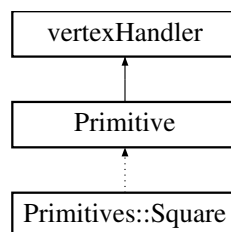
- std::vector< [Texture](#) > **TextureMap**

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

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

## 4.20 vertexHandler Class Reference

Inheritance diagram for vertexHandler:



### Public Member Functions

- 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

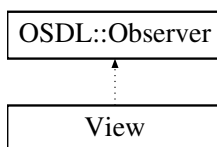
- unsigned int **VBO**
- unsigned int **VAO**
- unsigned int **EBO**
- unsigned int **texture**

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

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

## 4.21 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

## 4.22 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**

## 4.22.1 Constructor & Destructor Documentation

### 4.22.1.1 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, [19](#)
- AEG::Textures, [20](#)
- allocateBuffer
  - GPU::ArrayHandler, [9](#)
- areFreeBuffers
  - GPU::ArrayHandler, [9](#)
- areRepetitions
  - GPU::ArrayHandler, [9](#)
- Controller, [11](#)
- getFirstAvailavleBuffer
  - GPU::ArrayHandler, [10](#)
- Global::WindowProperties, [21](#)
  - WindowProperties, [22](#)
- GPU::ArrayHandler, [8](#)
  - allocateBuffer, [9](#)
  - areFreeBuffers, [9](#)
  - areRepetitions, [9](#)
  - getFirstAvailavleBuffer, [10](#)
  - releaseBuffer, [10](#)
- GPU::GPU\_Ref, [13](#)
- INPUT::Adapter, [7](#)
- INPUT::AdapterHandler, [7](#)
- Lights::DirectLight, [11](#)
- Lights::GlobalLight, [12](#)
- Lights::LightManager, [14](#)
- Lights::SphericalLight, [17](#)
- Model, [15](#)
- OSDL::DualLink, [12](#)
- OSDL::Observer, [15](#)
- OSDL::Subject, [18](#)
- Primitive, [15](#)
- Primitives::Line, [14](#)
- Primitives::PrimitivesManager, [16](#)
- Primitives::Square, [17](#)
- releaseBuffer
  - GPU::ArrayHandler, [10](#)
- vertexHandler, [20](#)
- View, [21](#)
- WindowProperties
  - Global::WindowProperties, [22](#)