Fast OpenGL Library

Generated by Doxygen 1.8.17

Chapter 1

WIP: FastOGLib - Fast OpenGL Library

Fast and scalable library, able to create OpenGL progams 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

Unfortunetly on Windows the GNU compiler with make is required (MVC option is no tested - Feel free to test 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 developement on this library the `main.cpp` file has to be modified. In future months I'll be adding more functionality, which can me 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:

Chapter 3

Class Index

3.1 Class List

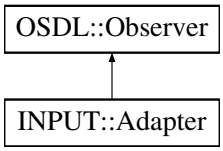
Here are the classes, structs, unions and interfaces with brief descriptions:

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

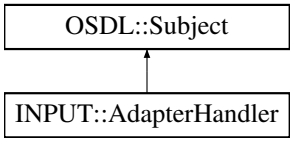
- std::map< KeyBoardKey, bool > **keyPressed**
- std::map< KeyBoardKey, bool > **keyHold**

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 avaliable.
- bool alocateBuffer (int)

  Alocates 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 alocateBuffer()

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

Alocates GPU space for x buffers.

Parameters

int	size amount of buffers

Note

Return values

bool	if buffers were created

4.3.1.2 areFreeBuffers()

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

Check if any buffers are avaliable.

Note

Return values

bool	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

int[ ]	array
int	size of array

Return values

true	If they're repetiotion
false	if there are no repetitions

4.3.1.4 getFirstAvailavleBuffer()

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

Returns pointer to first Available Buffer.

Note

Return values

GPU_Ref	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 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 Attributes

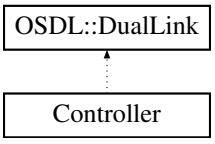
- Color< T > **color**
- T **alpha** = 0

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

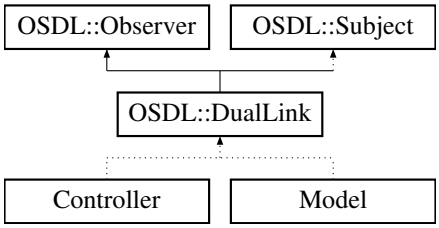
- float **x**
- float **y**
- float **lumen**
- float **alpha** = 1.0f
- float **r**
- float **g**
- float **b**
- float **coneAngle**
- float **coneWidth**
- float **coneLength**

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

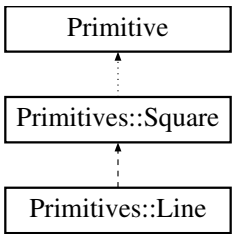
- std::vector< DirectLight * > **directLights**
- std::vector< SphericalLight * > **sphericalLights**
- std::vector< GlobalLight * > **globalLights**

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 w, int r, int g, int b, float alpha)

  Construct a new Line object.
- void Display ()

  Display the Line.
- bool isVisible () override

  is the Line visible

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 w,
            int r,
            int g,
            int b,
            float alpha )
```

Construct a new Line object.

Parameters

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

4.12.2 Member Function Documentation

4.12.2.1 Display()

```
void Primitives::Line::Display ( )
```

Display the Line.

4.12.2.2 isVisible()

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

is the Line visible

Returns

true

false

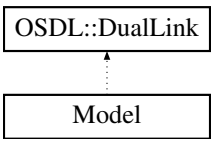
Implements Primitive.

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:



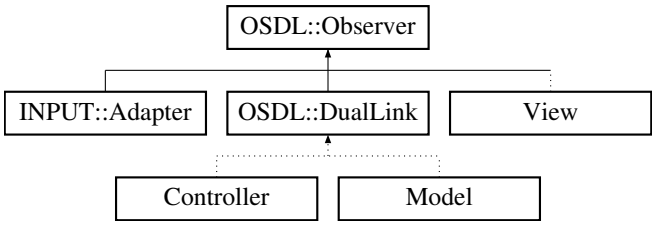
Additional Inherited Members

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

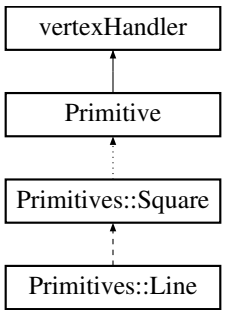
- T **x** = 0
- T **y** = 0

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

- float **sin** (float r)
- float **cos** (float r)
- **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

4.17 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 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

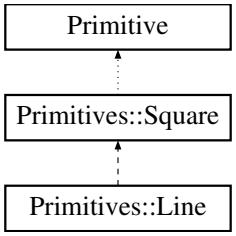
- float **x**
- float **y**
- float **lumen**
- float **alpha** = 1.0f
- float **r**
- float **g**
- float **b**

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

4.19 Primitives::Square Class Reference

Inheritance diagram for Primitives::Square:



Public Member Functions

- **Square** (float, float, float, float)
- 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 ()

  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.
- **Square** (const Square &s)
- Square & **operator=** (const Square &s)

Private Types

- typedef std::array< std::unique_ptr< PointF >, 4 > **verticies_ptr_array**
- typedef std::array< std::unique_ptr< ColorRGBAF >, 4 > **colors_ptr_array**

Private Member Functions

- void **calculateMatrixes** ()
- verticies_ptr_array rotateVertices ()

  Rotate the vertices of the square.

Private Attributes

- Global::WindowProperties * **window**
- float **x**
- float **y**
- float **size**
- verticies_ptr_array **verticies_ptr** = {}
- colors_ptr_array **color_ptr**
- float **angle** = 0.0f

4.19.1 Member Function Documentation

4.19.1.1 isNear()

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

is the Object int the radius

Parameters

float	x - x coordinate of the center of the circle
float	y - y coordinate of the center of the circle
float	radius - radius of the circle

Returns

true - if object is in the radius

false - if object is not in the radius

Implements Primitive.

4.19.1.2 isVisible()

```
bool Primitives::Square::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.

4.19.1.3 Move()

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

move the square relative to the current position

Parameters

float	x - offset in x direction
float	y - offset in y direction

4.19.1.4 rotate()

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

rotate the square relative to the current rotation

Parameters

float	angle - angle in radians

4.19.1.5 rotateVertices()

```
Square::verticies_ptr_array Primitives::Square::rotateVertices ( )  [private]
```

Rotate the vertices of the square.

Parameters

vertices	- vertices of the square
rot	- rotation matrix

4.19.1.6 setAlpha()

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

Set the Alpha value.

Parameters

float	alpha - alpha value

4.19.1.7 setColor()

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

Set the Color value.

Parameters

float	r - red value
float	g - green value
float	b - blue value
float	alpha - alpha value

4.19.1.8 setG()

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

Set G color value.

Parameters

g	- green value

4.19.1.9 setPosition()

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

Set the Position object.

Parameters

float	x - x coordinate of the center of the square
float	y - y coordinate of the center of the square

4.19.1.10 setR()

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

Set R color value.

Parameters

r	- red value

4.19.1.11 setRotation()

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

Set the rotation of the square.

Parameters

float	angle - angle in radians of the square

4.19.1.12 setTexture()

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

Set the Texture object.

Parameters

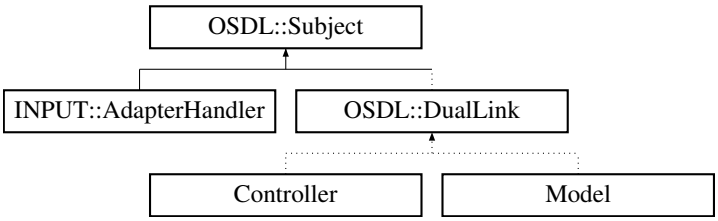
std::string	data - path to the texture

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

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

4.20 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

4.21 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

4.22 AEG::Textures Class Reference

Public Member Functions

- void **alocateTextureSpace** (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

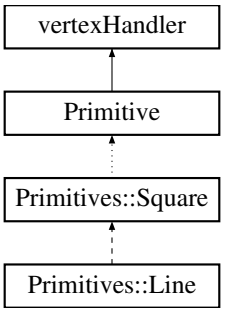
- std::vector< Texture > **TextureMap**

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

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

4.23 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

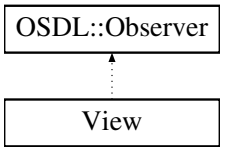
- unsigned int **VBO**
- unsigned int **VAO**
- unsigned int **EBO**
- unsigned int **texture**

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

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

4.24 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

4.25 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**

4.25.1 Constructor & Destructor Documentation

4.25.1.1 WindowProperties()

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

ArrayHandler for All GPU bindings.

Note

Return values

None	

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

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

Index