Fast OpenGL Library

Generated by Doxygen 1.8.17

Chapter 1

Core of framework

Chapter 2

WIP: FastOGLib - Fast OpenGL Library

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

2.1 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
```

2.1.1 Linux/OSX

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

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

2.1.2 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
```

2.1.3 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.

2.2 Contributors

@mwawrzkow - Marcin Wawrzków - owner

Chapter 3

Namespace Index

3.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Chapter 4

Hierarchical Index

4.1 Class Hierarchy

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

Chapter 5

Class Index

5.1 Class List

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

Chapter 6

File Index

6.1 File List

Here is a list of all files with brief descriptions:

Chapter 7

Namespace Documentation

7.1 AEG Namespace Reference

Classes

- class Texture
- class Textures

7.2 Global Namespace Reference

Classes

- class WindowProperties

7.3 GPU Namespace Reference

Classes

- class ArrayHandler
- class GPU_Ref

7.4 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 }

7.4.1 Enumeration Type Documentation

7.4.1.1 KeyBoardKey

`enum INPUT::KeyBoardKey`

Enumerator

ERROR	
Key_1	
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	

Enumerator

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

7.5 Lights Namespace Reference

Classes

- class DirectLight
- class GlobalLight
- class LightManager
- class SphericalLight

7.6 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 *fragShader↩ Code)
- GLuint CreateProgramAndLinkShaders (const std::vector< GLuint > shaders)

7.6.1 Function Documentation

7.6.1.1 CreateAndCompileShaders()

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

7.6.1.2 CreateProgramAndLinkShaders()

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

7.6.1.3 InitWindow()

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

7.6.1.4 setClearColor()

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

7.7 OSDL Namespace Reference

Classes

- class DualLink
- class Observer
- class Subject

7.8 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 > ColorRGBAU
- 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 }

7.8.1 Typedef Documentation

7.8.1.1 ColorC

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

7.8.1.2 ColorD

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

7.8.1.3 ColorF

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

7.8.1.4 ColorI

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

7.8.1.5 ColorL

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

7.8.1.6 ColorLL

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

7.8.1.7 ColorRGBAC

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

7.8.1.8 ColorRGBAD

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

7.8.1.9 ColorRGBAF

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

7.8.1.10 ColorRGBAI

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

7.8.1.11 ColorRGBAL

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

7.8.1.12 ColorRGBALL

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

7.8.1.13 ColorRGBAS

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

7.8.1.14 ColorRGBAU

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

7.8.1.15 ColorRGBAUL

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

7.8.1.16 ColorRGBAULL

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

7.8.1.17 ColorS

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

7.8.1.18 ColorU

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

7.8.1.19 ColorUL

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

7.8.1.20 ColorULL

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

7.8.1.21 PointC

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

7.8.1.22 PointD

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

7.8.1.23 PointF

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

7.8.1.24 PointI

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

7.8.1.25 PointL

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

7.8.1.26 PointLL

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

7.8.1.27 PointS

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

7.8.1.28 PointU

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

7.8.1.29 PointUL

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

7.8.1.30 PointULL

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

7.8.1.31 PrimitivePtr

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

7.8.1.32 Primitives

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

7.8.1.33 PrimitivesRef

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

7.8.2 Enumeration Type Documentation

7.8.2.1 ColorType

```
enum Primitives::ColorType
```

Enumerator

RED	
GREEN	
BLUE	
ALPHA	
ERROR	

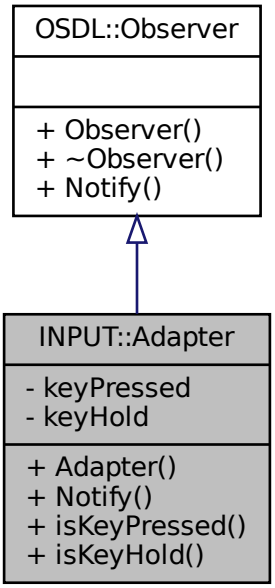
Chapter 8

Class Documentation

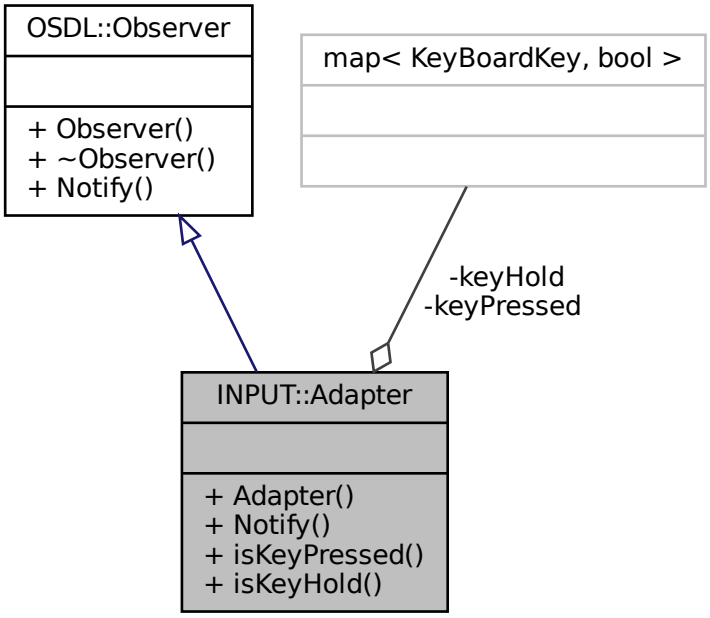
8.1 INPUT::Adapter Class Reference

`#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

8.1.1 Constructor & Destructor Documentation

8.1.1.1 Adapter()

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

8.1.2 Member Function Documentation

8.1.2.1 isKeyHold()

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

8.1.2.2 isKeyPressed()

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

8.1.2.3 Notify()

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

Reimplemented from OSDL::Observer.

8.1.3 Member Data Documentation

8.1.3.1 keyHold

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

8.1.3.2 keyPressed

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

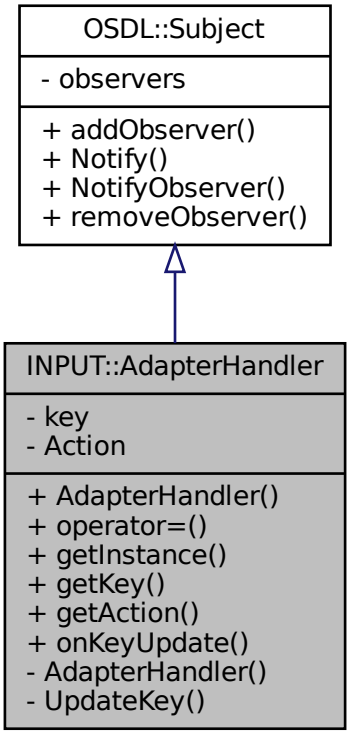
The documentation for this class was generated from the following files:

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

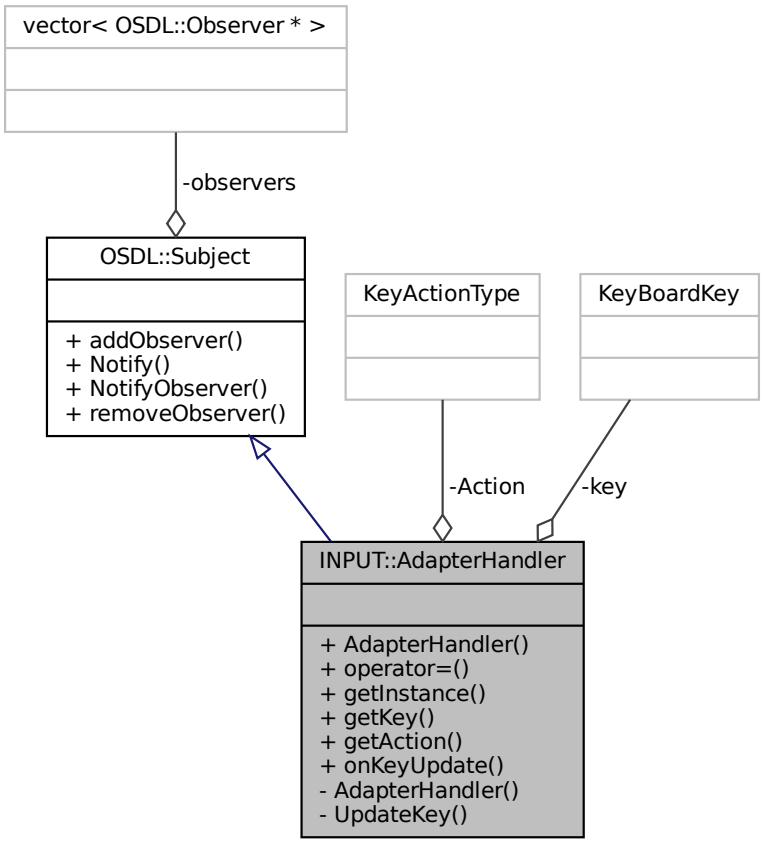
8.2 INPUT::AdapterHandler Class Reference

`#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

8.2.1 Member Enumeration Documentation

8.2.1.1 KeyActionType

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

Enumerator

Hold	
Press_down	
Press_Up	

8.2.2 Constructor & Destructor Documentation

8.2.2.1 AdapterHandler() [1/2]

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

8.2.2.2 AdapterHandler() [2/2]

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

8.2.3 Member Function Documentation

8.2.3.1 getAction()

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

8.2.3.2 getInstance()

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

8.2.3.3 getKey()

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

8.2.3.4 onKeyUpdate()

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

8.2.3.5 operator=()

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

8.2.3.6 UpdateKey()

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

8.2.4 Member Data Documentation

8.2.4.1 Action

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

8.2.4.2 key

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

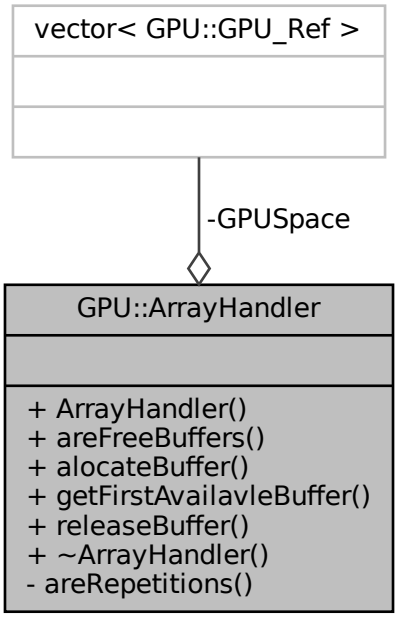
The documentation for this class was generated from the following files:

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

8.3 GPU::ArrayHandler Class Reference

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

Collaboration diagram for GPU::ArrayHandler:



Public Member Functions

- ArrayHandler ()
- 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.
- ∼ArrayHandler ()

Private Member Functions

- bool areRepetitions (unsigned int[ ], int)

  check if are repetitions in Array

Private Attributes

- std::vector< GPU_Ref > GPUSpace

8.3.1 Constructor & Destructor Documentation

8.3.1.1 ArrayHandler()

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

8.3.1.2 ∼ArrayHandler()

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

8.3.2 Member Function Documentation

8.3.2.1 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

8.3.2.2 areFreeBuffers()

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

Check if any buffers are avaliable.

Note

Return values

bool	are any buffers free

8.3.2.3 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

8.3.2.4 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

8.3.2.5 releaseBuffer()

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

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

Note

Exceptions

* *	

Return values

None	

8.3.3 Member Data Documentation

8.3.3.1 GPUSpace

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

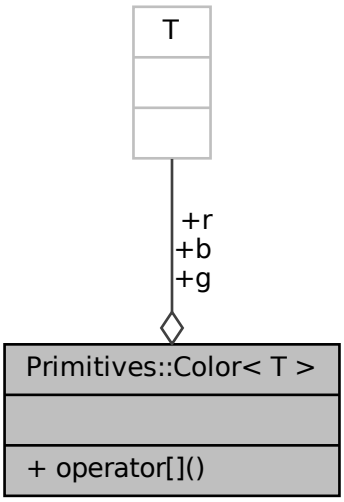
The documentation for this class was generated from the following files:

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

8.4 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

8.4.1 Member Function Documentation

8.4.1.1 operator[]()

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

8.4.2 Member Data Documentation

8.4.2.1 b

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

8.4.2.2 g

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

8.4.2.3 r

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

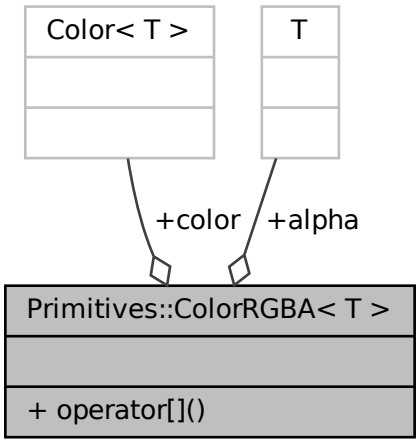
The documentation for this struct was generated from the following file:

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

8.5 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

8.5.1 Member Function Documentation

8.5.1.1 operator[]()

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

8.5.2 Member Data Documentation

8.5.2.1 alpha

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

8.5.2.2 color

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

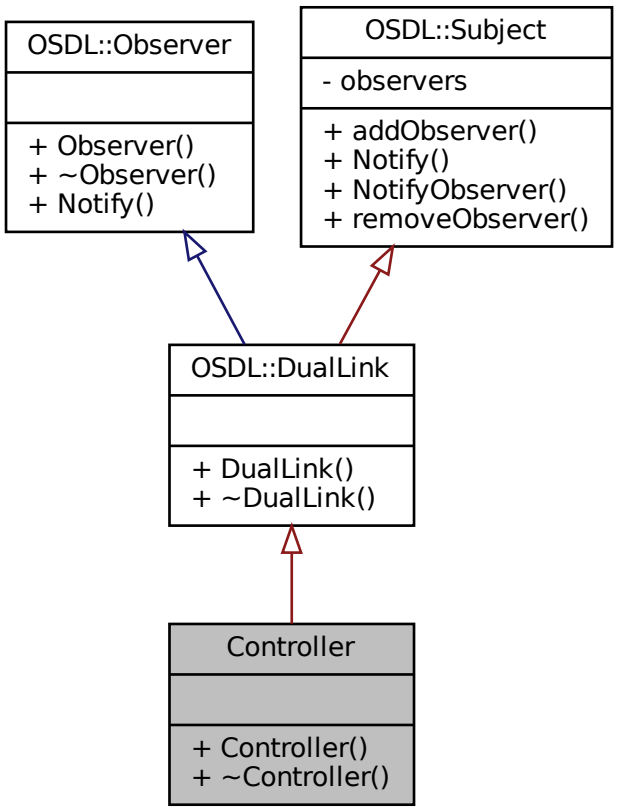
The documentation for this struct was generated from the following file:

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

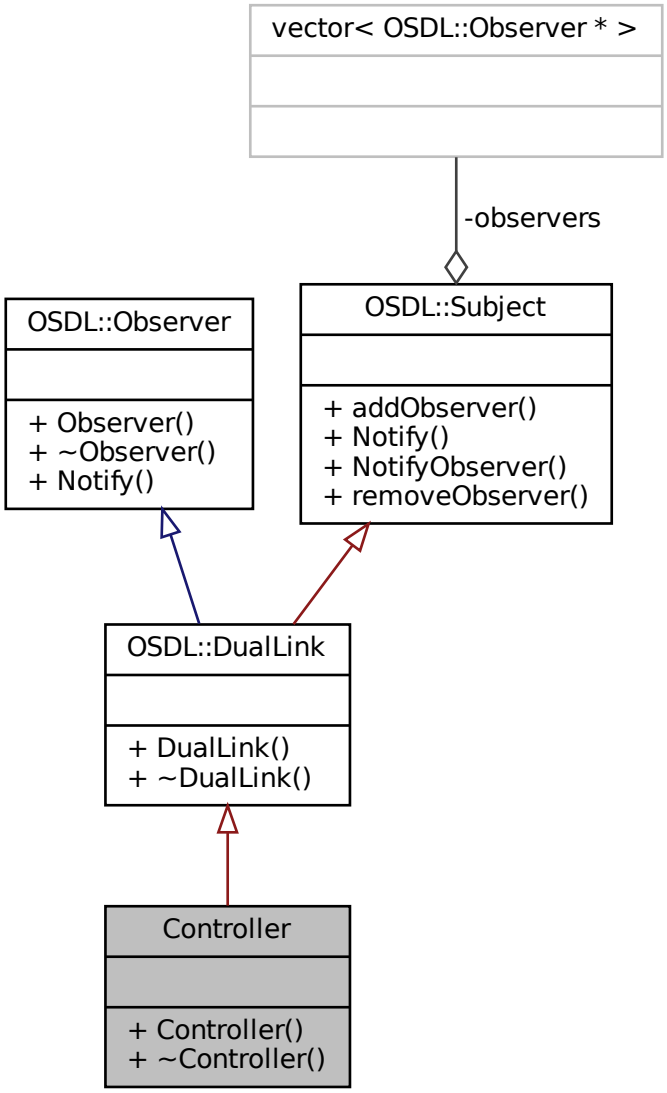
8.6 Controller Class Reference

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

Inheritance diagram for Controller:



Collaboration diagram for Controller:



Public Member Functions

- Controller ()
- ~Controller ()

Additional Inherited Members

8.6.1 Constructor & Destructor Documentation

8.6.1.1 Controller()

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

8.6.1.2 ~Controller()

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

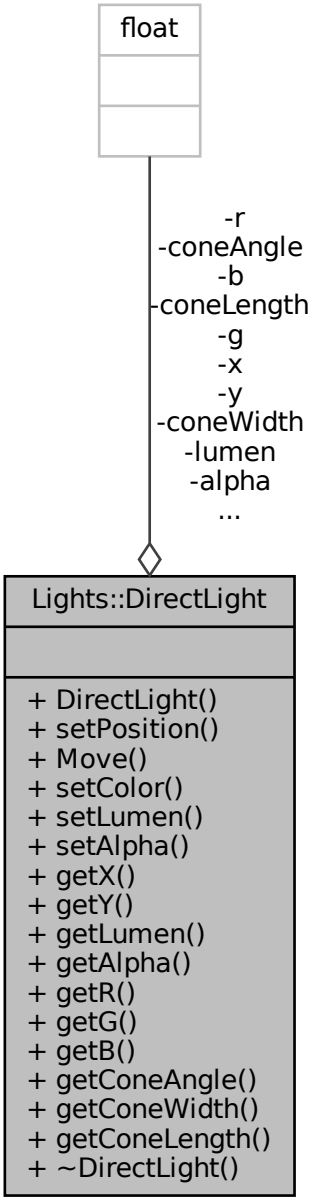
The documentation for this class was generated from the following file:

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

8.7 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

8.7.1 Constructor & Destructor Documentation

8.7.1.1 DirectLight()

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

8.7.1.2 ~DirectLight()

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

8.7.2 Member Function Documentation

8.7.2.1 getAlpha()

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

8.7.2.2 getB()

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

8.7.2.3 getConeAngle()

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

8.7.2.4 getConeLength()

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

8.7.2.5 getConeWidth()

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

8.7.2.6 getG()

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

8.7.2.7 getLumen()

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

8.7.2.8 getR()

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

8.7.2.9 getX()

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

8.7.2.10 getY()

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

8.7.2.11 Move()

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

8.7.2.12 setAlpha()

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

8.7.2.13 setColor()

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

8.7.2.14 setLumen()

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

8.7.2.15 setPosition()

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

8.7.3 Member Data Documentation

8.7.3.1 alpha

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

8.7.3.2 b

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

8.7.3.3 coneAngle

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

8.7.3.4 coneLength

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

8.7.3.5 coneWidth

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

8.7.3.6 g

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

8.7.3.7 lumen

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

8.7.3.8 r

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

8.7.3.9 x

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

8.7.3.10 y

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

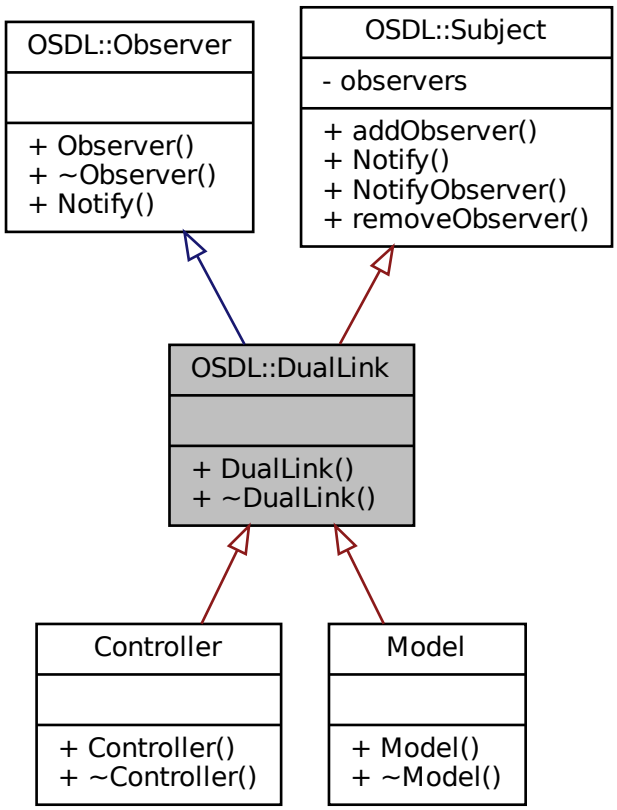
The documentation for this class was generated from the following files:

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

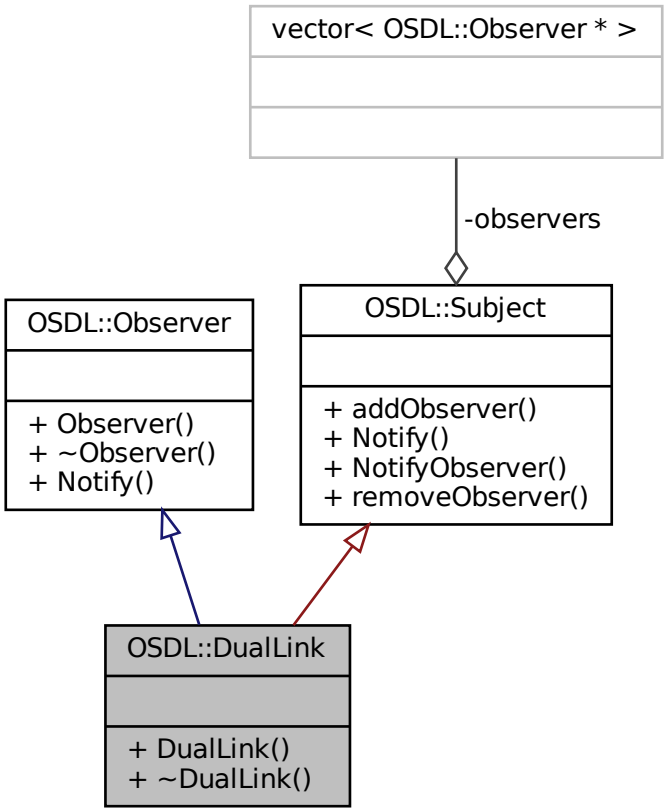
8.8 OSDL::DualLink Class Reference

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

Inheritance diagram for OSDL::DualLink:



Collaboration diagram for OSDL::DualLink:



Public Member Functions

- DualLink ()
- ~DualLink ()

Additional Inherited Members

8.8.1 Constructor & Destructor Documentation

8.8.1.1 DualLink()

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

8.8.1.2 ~DualLink()

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

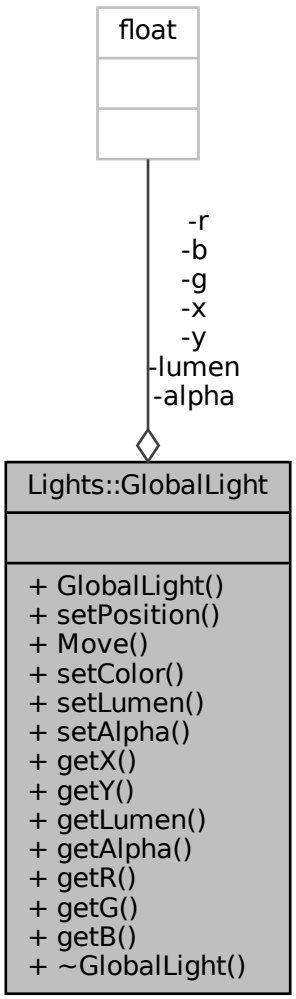
The documentation for this class was generated from the following file:

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

8.9 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

8.9.1 Constructor & Destructor Documentation

8.9.1.1 GlobalLight()

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

8.9.1.2 ~GlobalLight()

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

8.9.2 Member Function Documentation

8.9.2.1 getAlpha()

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

8.9.2.2 getB()

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

8.9.2.3 getG()

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

8.9.2.4 getLumen()

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

8.9.2.5 getR()

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

8.9.2.6 getX()

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

8.9.2.7 getY()

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

8.9.2.8 Move()

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

8.9.2.9 setAlpha()

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

8.9.2.10 setColor()

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

8.9.2.11 setLumen()

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

8.9.2.12 setPosition()

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

8.9.3 Member Data Documentation

8.9.3.1 alpha

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

8.9.3.2 b

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

8.9.3.3 g

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

8.9.3.4 lumen

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

8.9.3.5 r

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

8.9.3.6 x

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

8.9.3.7 y

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

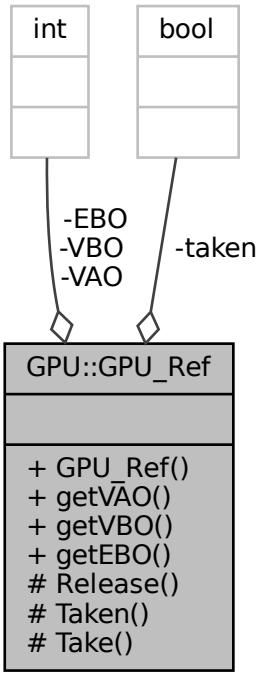
The documentation for this class was generated from the following files:

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

8.10 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

8.10.1 Constructor & Destructor Documentation

8.10.1.1 GPU_Ref()

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

8.10.2 Member Function Documentation

8.10.2.1 getEBO()

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

8.10.2.2 getVAO()

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

8.10.2.3 getVBO()

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

8.10.2.4 Release()

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

8.10.2.5 Take()

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

8.10.2.6 Taken()

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

8.10.3 Friends And Related Function Documentation

8.10.3.1 ArrayHandler

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

8.10.4 Member Data Documentation

8.10.4.1 EBO

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

8.10.4.2 taken

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

8.10.4.3 VAO

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

8.10.4.4 VBO

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

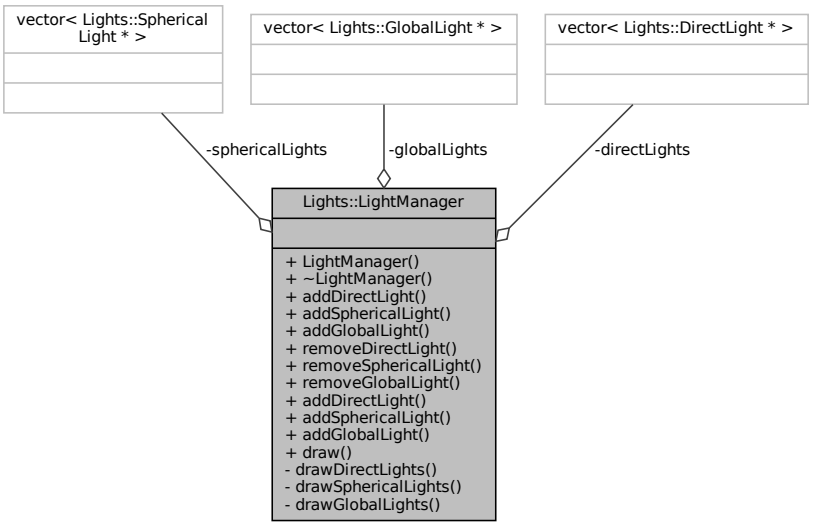
The documentation for this class was generated from the following files:

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

8.11 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, 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

8.11.1 Constructor & Destructor Documentation

8.11.1.1 LightManager()

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

8.11.1.2 ~LightManager()

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

8.11.2 Member Function Documentation

8.11.2.1 addDirectLight() [1/2]

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

8.11.2.2 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 )
```

8.11.2.3 addGlobalLight() [1/2]

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

8.11.2.4 addGlobalLight() [2/2]

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

8.11.2.5 addSphericalLight() [1/2]

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

8.11.2.6 addSphericalLight() [2/2]

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

8.11.2.7 draw()

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

8.11.2.8 drawDirectLights()

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

8.11.2.9 drawGlobalLights()

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

8.11.2.10 drawSphericalLights()

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

8.11.2.11 removeDirectLight()

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

8.11.2.12 removeGlobalLight()

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

8.11.2.13 removeSphericalLight()

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

8.11.3 Member Data Documentation

8.11.3.1 directLights

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

8.11.3.2 globalLights

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

8.11.3.3 sphericalLights

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

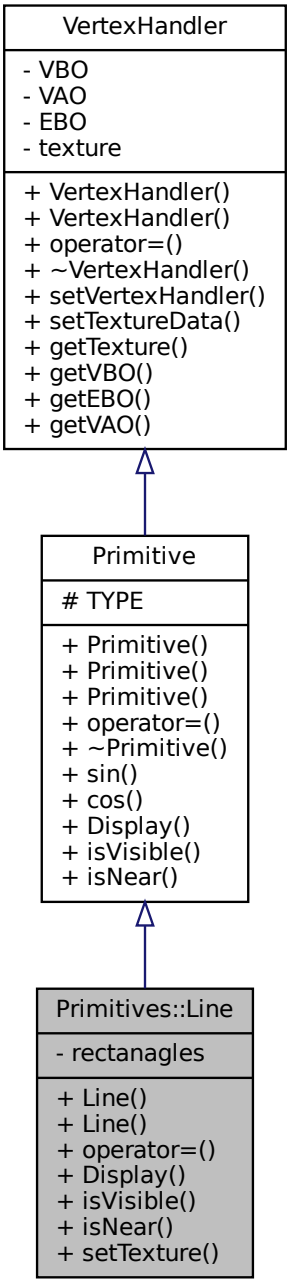
The documentation for this class was generated from the following files:

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

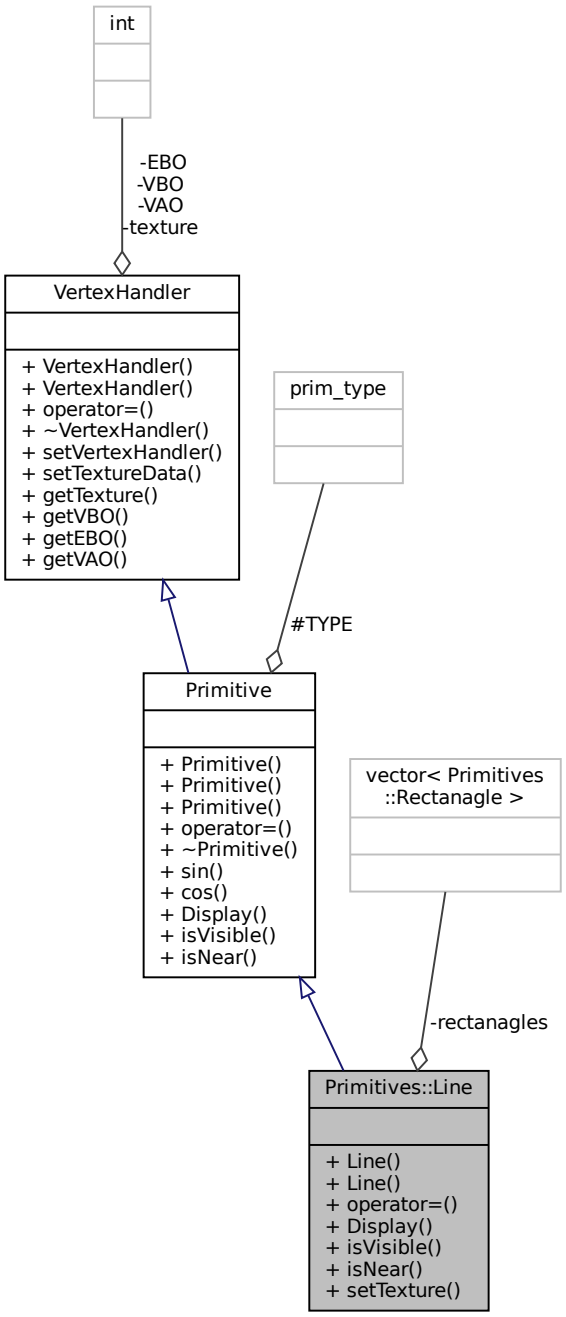
8.12 Primitives::Line Class Reference

```
#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< Rectanagle > rectanagles

Additional Inherited Members

8.12.1 Constructor & Destructor Documentation

8.12.1.1 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

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

8.12.1.2 Line() [2/2]

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

8.12.2 Member Function Documentation

8.12.2.1 Display()

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

Display the Line.

Implements Primitive.

8.12.2.2 isNear()

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

Implements Primitive.

8.12.2.3 isVisible()

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

is the Line visible

Returns

true

false

Implements Primitive.

8.12.2.4 operator=()

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

8.12.2.5 setTexture()

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

8.12.3 Member Data Documentation

8.12.3.1 rectanagles

```
std::vector<Rectanagle> Primitives::Line::rectanagles  [private]
```

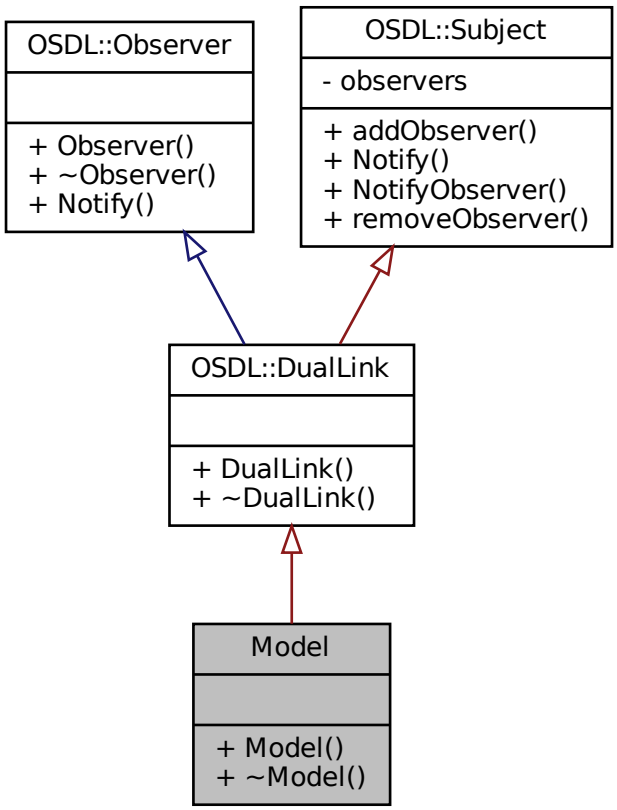
The documentation for this class was generated from the following files:

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

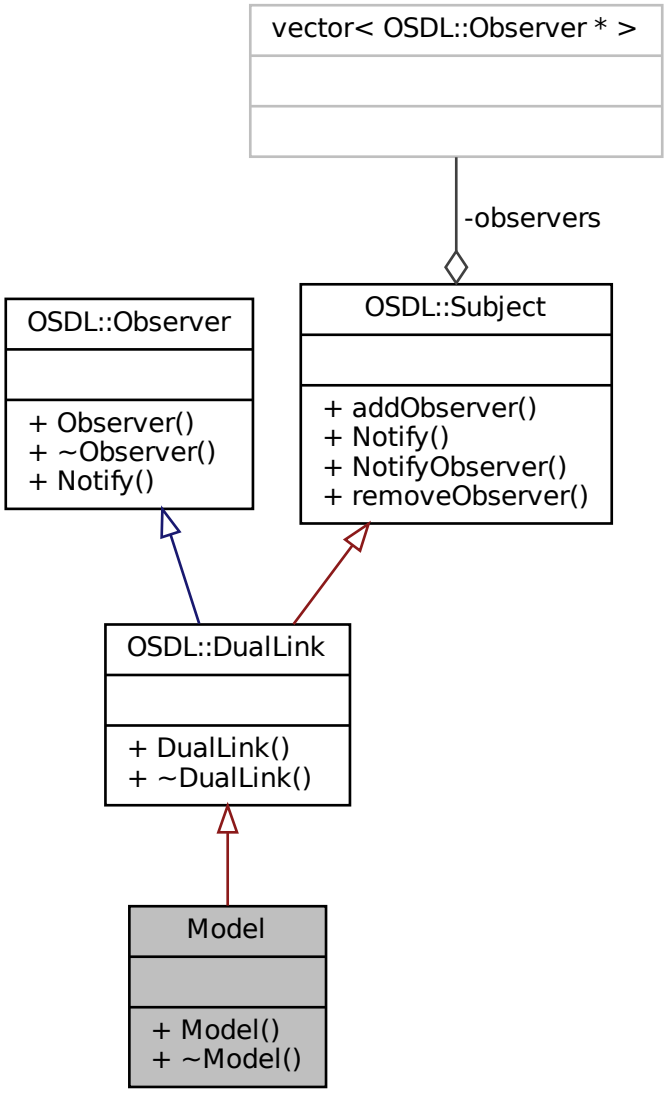
8.13 Model Class Reference

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

Inheritance diagram for Model:



Collaboration diagram for Model:



Public Member Functions

- Model ()
- ~Model ()

Additional Inherited Members

8.13.1 Constructor & Destructor Documentation

8.13.1.1 Model()

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

8.13.1.2 ~Model()

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

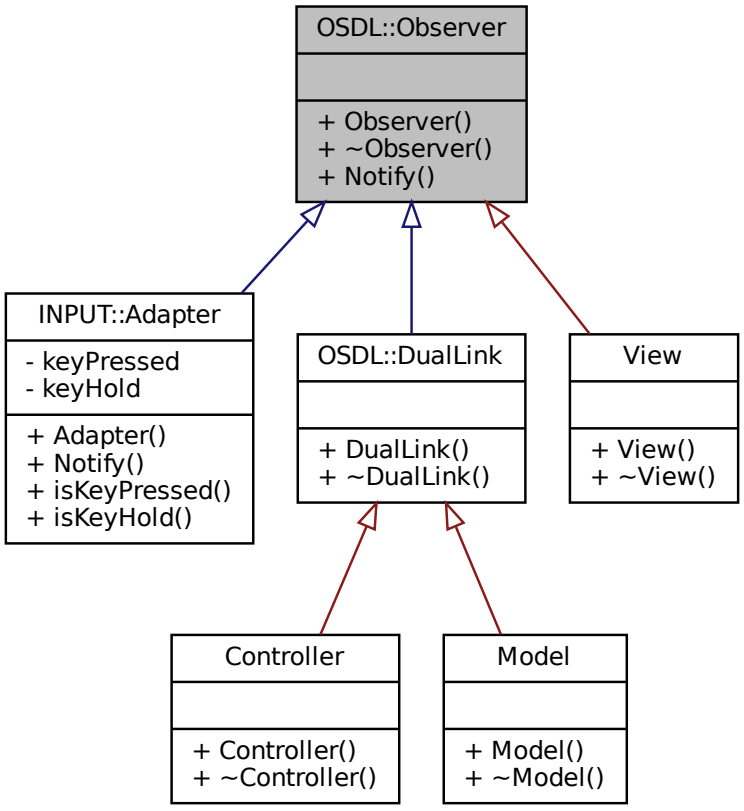
The documentation for this class was generated from the following file:

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

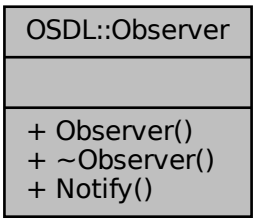
8.14 OSDL::Observer Class Reference

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

Inheritance diagram for OSDL::Observer:



Collaboration diagram for OSDL::Observer:



Public Member Functions

- Observer ()
- ~Observer ()
- virtual void Notify (Subject *)

8.14.1 Constructor & Destructor Documentation

8.14.1.1 Observer()

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

8.14.1.2 ~Observer()

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

8.14.2 Member Function Documentation

8.14.2.1 Notify()

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

Reimplemented in INPUT::Adapter.

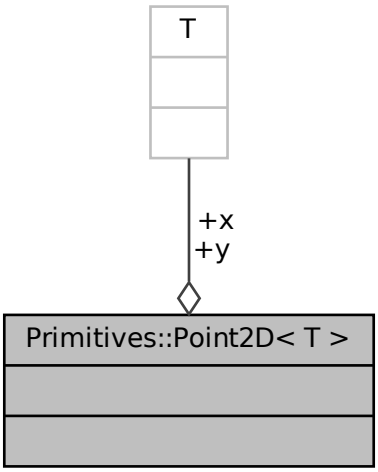
The documentation for this class was generated from the following file:

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

8.15 Primitives::Point2D< T > Struct Template Reference

`#include <Point.hpp>`

Collaboration diagram for Primitives::Point2D< T >:



Public Attributes

- T x = 0
- T y = 0

8.15.1 Member Data Documentation

8.15.1.1 x

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

8.15.1.2 y

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

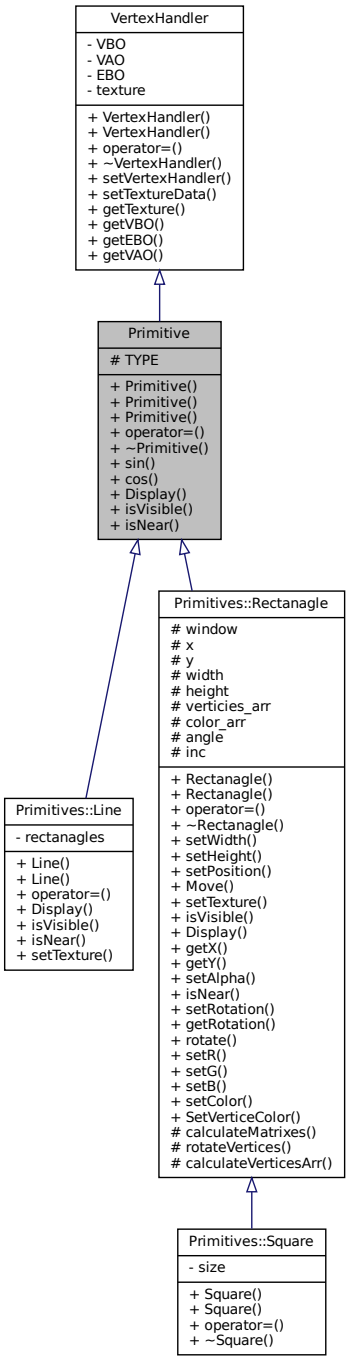
The documentation for this struct was generated from the following file:

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

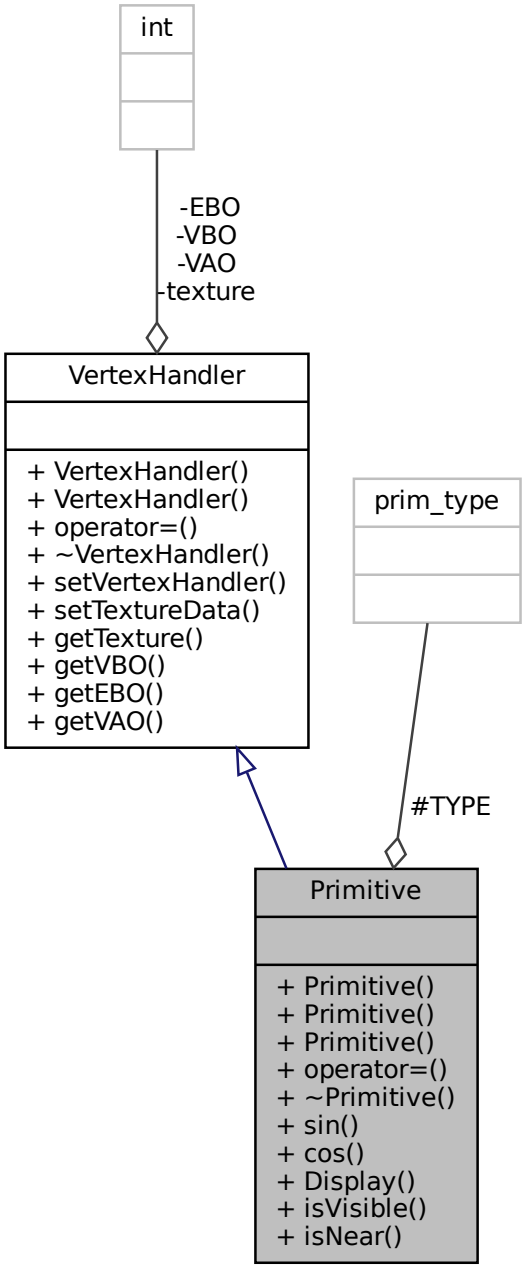
8.16 Primitive Class Reference

`#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

8.16.1 Member Enumeration Documentation

8.16.1.1 prim_type

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

Enumerator

RECTANAGLE	
TRIANGLE	
SQUARE	
LINE	

8.16.2 Constructor & Destructor Documentation

8.16.2.1 Primitive() [1/3]

```
Primitive::Primitive (
            prim_type type )
```

8.16.2.2 Primitive() [2/3]

```
Primitive::Primitive (
            prim_type type,
            bool isDefaultPrimitive )
```

8.16.2.3 Primitive() [3/3]

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

8.16.2.4 ~Primitive()

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

8.16.3 Member Function Documentation

8.16.3.1 cos()

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

8.16.3.2 Display()

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

Implemented in Primitives::Rectanagle, and Primitives::Line.

8.16.3.3 isNear()

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

Implemented in Primitives::Rectanagle, and Primitives::Line.

8.16.3.4 isVisible()

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

Implemented in Primitives::Rectanagle, and Primitives::Line.

8.16.3.5 operator=()

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

8.16.3.6 sin()

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

8.16.4 Member Data Documentation

8.16.4.1 TYPE

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

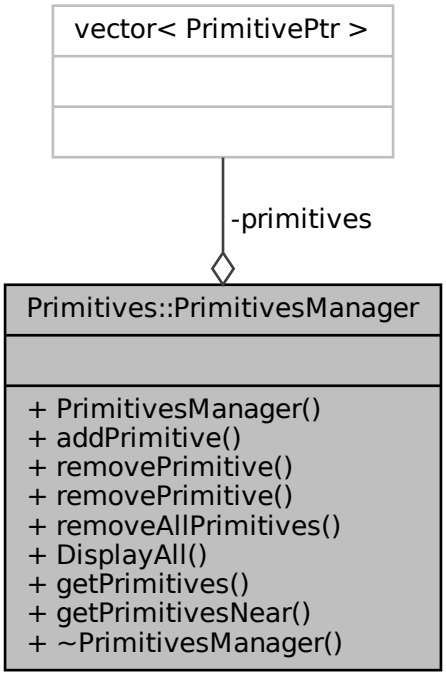
The documentation for this class was generated from the following files:

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

8.17 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

8.17.1 Constructor & Destructor Documentation

8.17.1.1 PrimitivesManager()

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

8.17.1.2 ~PrimitivesManager()

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

8.17.2 Member Function Documentation

8.17.2.1 addPrimitive()

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

8.17.2.2 DisplayAll()

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

8.17.2.3 getPrimitives()

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

8.17.2.4 getPrimitivesNear()

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

8.17.2.5 removeAllPrimitives()

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

8.17.2.6 removePrimitive() [1/2]

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

8.17.2.7 removePrimitive() [2/2]

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

8.17.3 Member Data Documentation

8.17.3.1 primitives

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

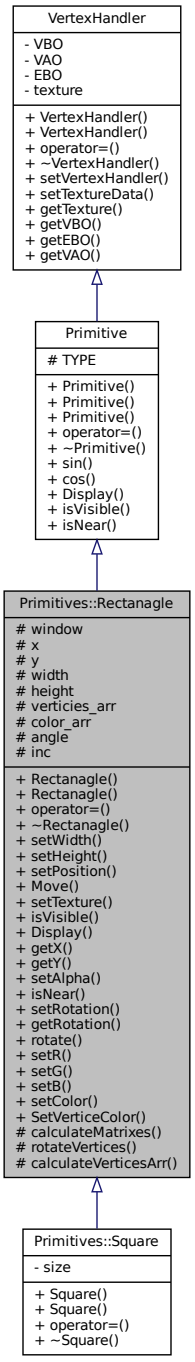
The documentation for this class was generated from the following files:

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

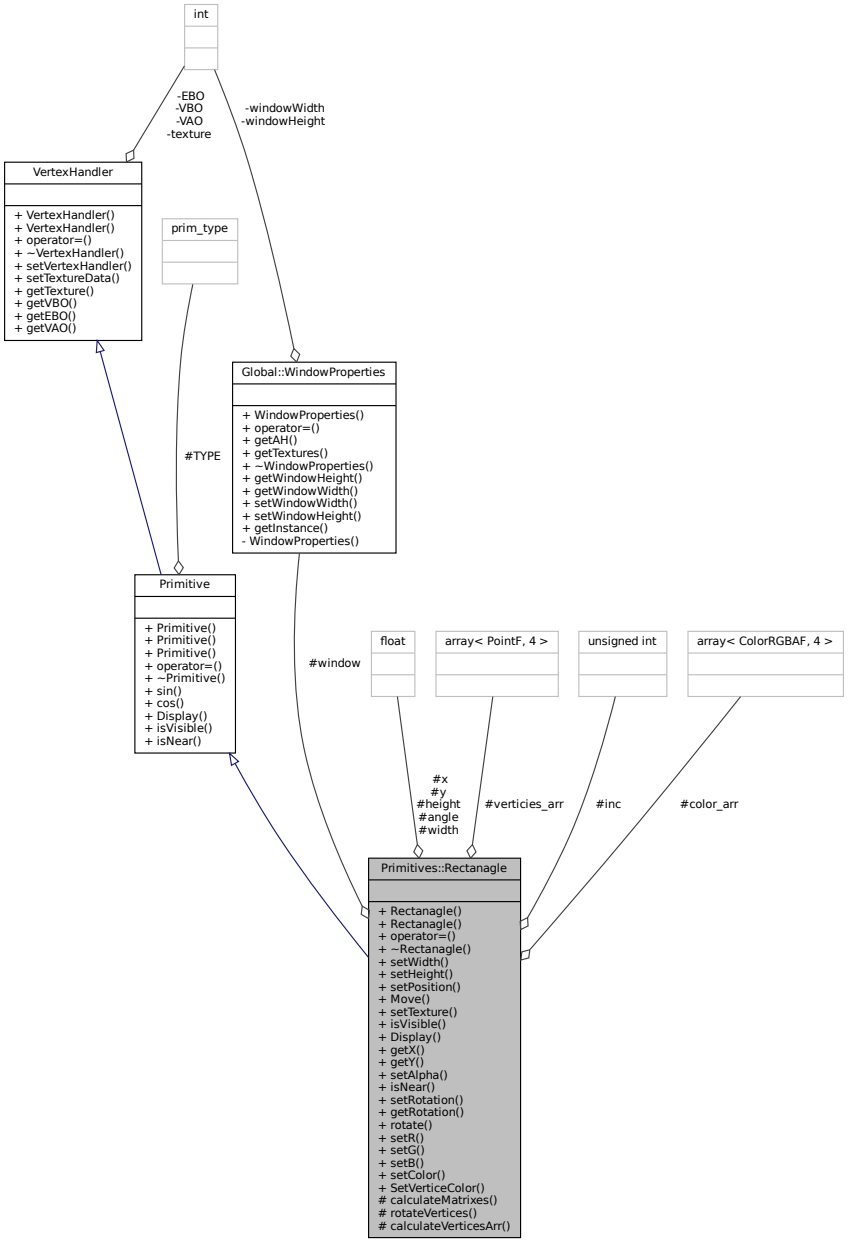
8.18 Primitives::Rectanagle Class Reference

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

Inheritance diagram for Primitives::Rectanagle:



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 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
- verticies_array verticies_arr = {}
- colors_array color_arr = {}
- float angle = 0.0f
- unsigned int inc [6] = {0, 1, 2, 0, 2, 3}

8.18.1 Member Typedef Documentation

8.18.1.1 colors_array

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

8.18.1.2 verticies_array

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

8.18.2 Constructor & Destructor Documentation

8.18.2.1 Rectanagle() [1/2]

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

8.18.2.2 Rectanagle() [2/2]

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

8.18.2.3 ~Rectanagle()

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

8.18.3 Member Function Documentation

8.18.3.1 calculateMatrixes()

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

8.18.3.2 calculateVerticesArr()

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

8.18.3.3 Display()

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

Draw the square.

Implements Primitive.

8.18.3.4 getRotation()

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

8.18.3.5 getX()

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

8.18.3.6 getY()

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

8.18.3.7 isNear()

```
bool Primitives::Rectanagle::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.

8.18.3.8 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.

8.18.3.9 Move()

```
void Primitives::Rectanagle::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

8.18.3.10 operator=()

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

8.18.3.11 rotate()

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

rotate the square relative to the current rotation

Parameters

float	angle - angle in radians

8.18.3.12 rotateVertices()

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

Rotate the vertices of the square.

Parameters

vertices	- vertices of the square
rot	- rotation matrix

8.18.3.13 setAlpha()

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

Set the Alpha value.

Parameters

float	alpha - alpha value

8.18.3.14 setB()

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

8.18.3.15 setColor()

```
void Primitives::Rectanagle::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

8.18.3.16 setG()

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

Set G color value.

Parameters

g	- green value

8.18.3.17 setHeight()

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

Set the Height value.

8.18.3.18 setPosition()

```
void Primitives::Rectanagle::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

8.18.3.19 setR()

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

Set R color value.

Parameters

r	- red value

8.18.3.20 setRotation()

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

Set the rotation of the square.

Parameters

float	angle - angle in radians of the square

8.18.3.21 setTexture()

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

Set the Texture object.

Parameters

std::string	data - path to the texture

8.18.3.22 SetVerticeColor()

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

8.18.3.23 setWidth()

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

Set the Width value.

8.18.4 Member Data Documentation

8.18.4.1 angle

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

8.18.4.2 color_arr

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

8.18.4.3 height

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

8.18.4.4 inc

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

8.18.4.5 verticies_arr

```
verticies_array Primitives::Rectanagle::verticies_arr = {}  [protected]
```

8.18.4.6 width

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

8.18.4.7 window

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

8.18.4.8 x

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

8.18.4.9 y

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

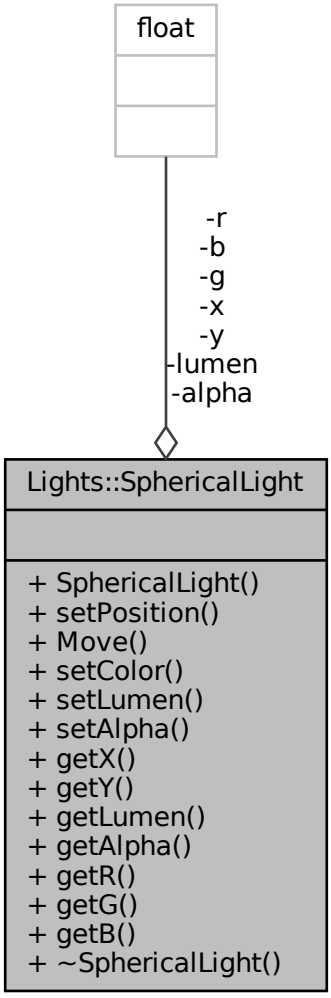
The documentation for this class was generated from the following files:

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

8.19 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

8.19.1 Constructor & Destructor Documentation

8.19.1.1 SphericalLight()

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

8.19.1.2 ~SphericalLight()

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

8.19.2 Member Function Documentation

8.19.2.1 getAlpha()

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

8.19.2.2 getB()

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

8.19.2.3 getG()

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

8.19.2.4 getLumen()

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

8.19.2.5 getR()

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

8.19.2.6 getX()

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

8.19.2.7 getY()

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

8.19.2.8 Move()

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

8.19.2.9 setAlpha()

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

8.19.2.10 setColor()

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

8.19.2.11 setLumen()

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

8.19.2.12 setPosition()

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

8.19.3 Member Data Documentation

8.19.3.1 alpha

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

8.19.3.2 b

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

8.19.3.3 g

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

8.19.3.4 lumen

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

8.19.3.5 r

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

8.19.3.6 x

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

8.19.3.7 y

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

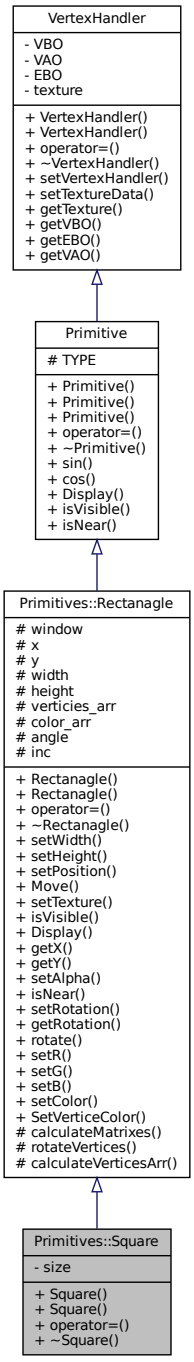
The documentation for this class was generated from the following files:

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

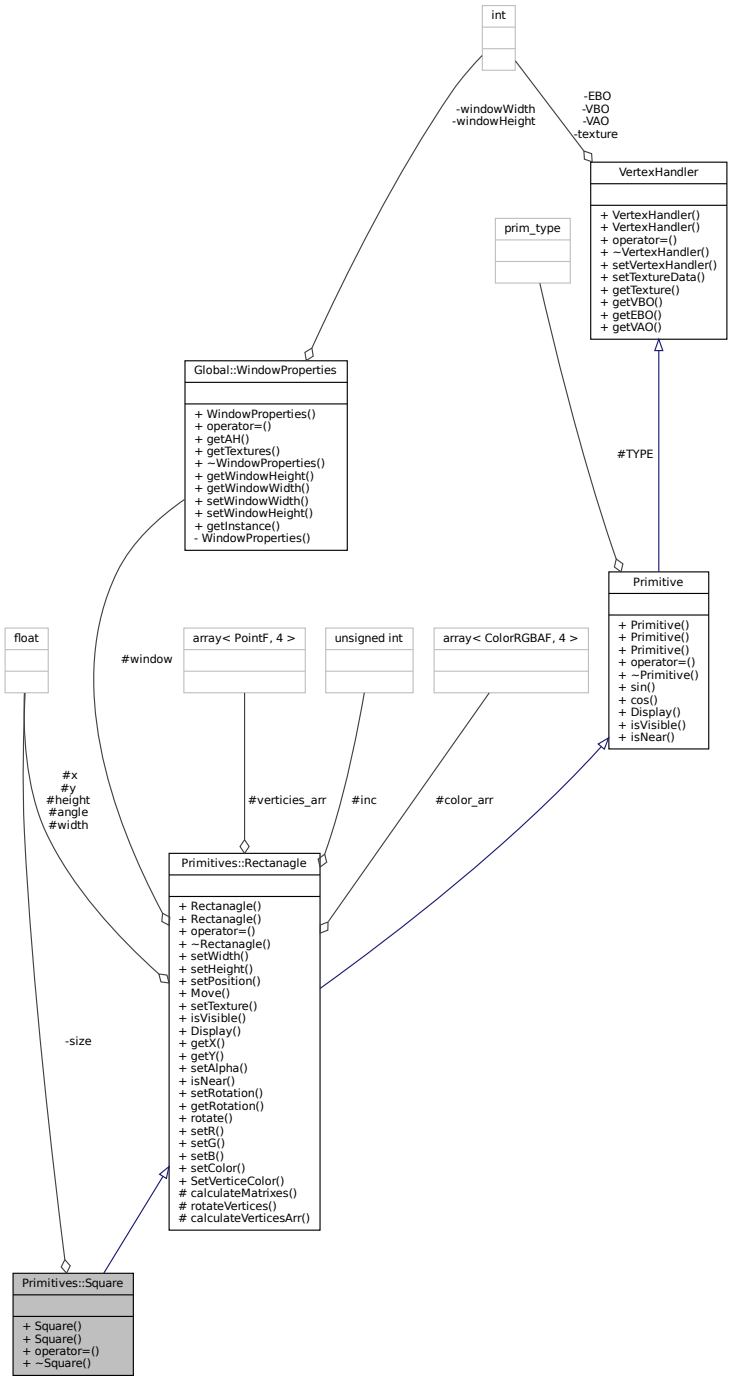
8.20 Primitives::Square Class Reference

`#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

8.20.1 Constructor & Destructor Documentation

8.20.1.1 Square() [1/2]

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

8.20.1.2 Square() [2/2]

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

8.20.1.3 ~Square()

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

8.20.2 Member Function Documentation

8.20.2.1 operator=()

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

8.20.3 Member Data Documentation

8.20.3.1 size

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

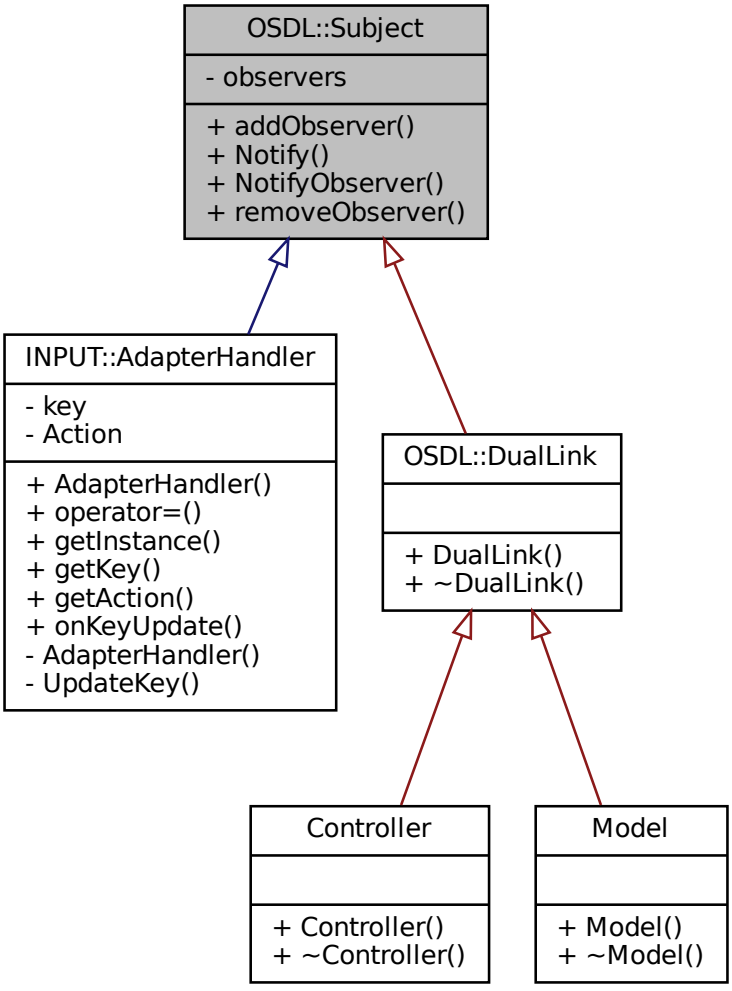
The documentation for this class was generated from the following files:

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

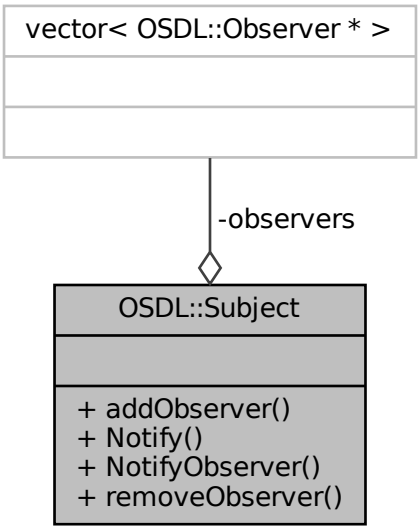
8.21 OSDL::Subject Class Reference

```
#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

8.21.1 Member Function Documentation

8.21.1.1 addObserver()

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

8.21.1.2 Notify()

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

8.21.1.3 NotifyObserver()

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

8.21.1.4 removeObserver()

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

8.21.2 Member Data Documentation

8.21.2.1 observers

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

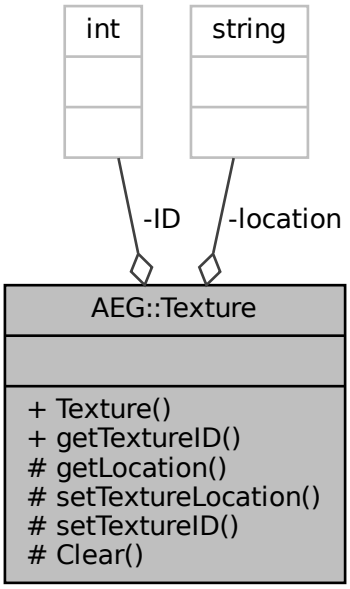
The documentation for this class was generated from the following files:

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

8.22 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

8.22.1 Constructor & Destructor Documentation

8.22.1.1 Texture()

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

8.22.2 Member Function Documentation

8.22.2.1 Clear()

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

8.22.2.2 getLocation()

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

8.22.2.3 getTextureID()

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

8.22.2.4 setTextureID()

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

8.22.2.5 setTextureLocation()

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

8.22.3 Friends And Related Function Documentation

8.22.3.1 Textures

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

8.22.4 Member Data Documentation

8.22.4.1 ID

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

8.22.4.2 location

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

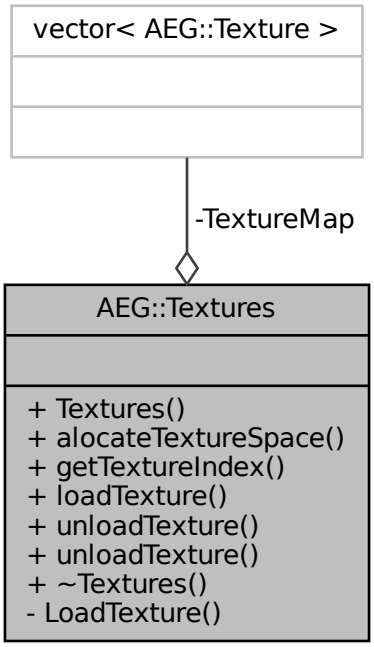
The documentation for this class was generated from the following files:

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

8.23 AEG::Textures Class Reference

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

Collaboration diagram for AEG::Textures:



Public Member Functions

- Textures ()
- 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)
- ~Textures ()

Private Member Functions

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

Private Attributes

- std::vector< Texture > TextureMap

8.23.1 Constructor & Destructor Documentation

8.23.1.1 Textures()

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

8.23.1.2 ~Textures()

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

8.23.2 Member Function Documentation

8.23.2.1 alocateTextureSpace()

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

8.23.2.2 getTextureIndex()

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

8.23.2.3 loadTexture()

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

8.23.2.4 LoadTexture()

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

8.23.2.5 unloadTexture() [1/2]

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

8.23.2.6 unloadTexture() [2/2]

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

8.23.3 Member Data Documentation

8.23.3.1 TextureMap

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

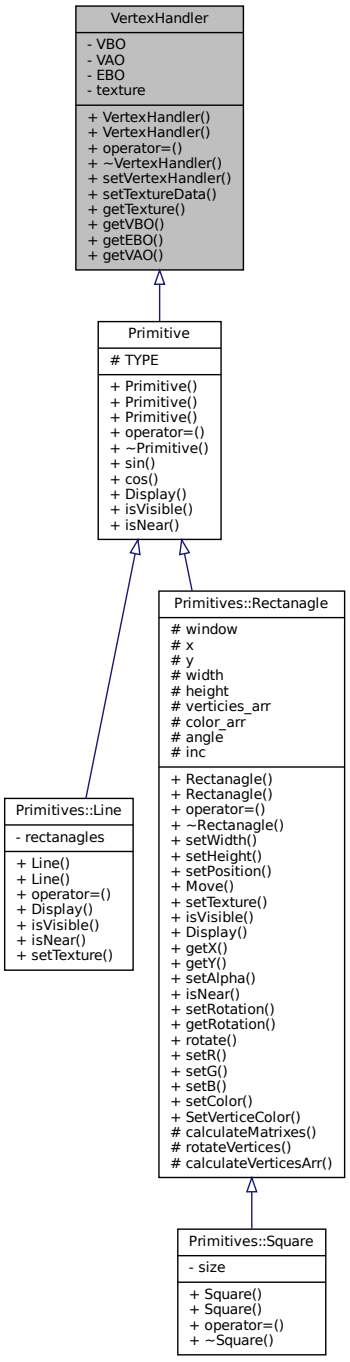
The documentation for this class was generated from the following files:

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

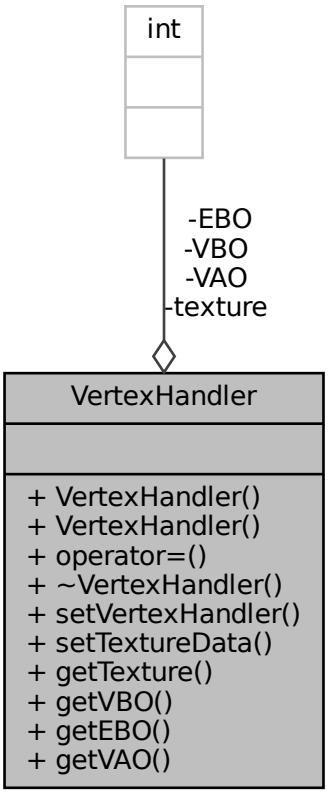
8.24 VertexHandler Class Reference

```
#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

8.24.1 Constructor & Destructor Documentation

8.24.1.1 VertexHandler() [1/2]

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

8.24.1.2 VertexHandler() [2/2]

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

8.24.1.3 ~VertexHandler()

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

8.24.2 Member Function Documentation

8.24.2.1 getEBO()

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

8.24.2.2 getTexture()

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

8.24.2.3 getVAO()

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

8.24.2.4 getVBO()

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

8.24.2.5 operator=()

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

8.24.2.6 setTextureData()

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

8.24.2.7 setVertexHandler()

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

8.24.3 Member Data Documentation

8.24.3.1 EBO

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

8.24.3.2 texture

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

8.24.3.3 VAO

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

8.24.3.4 VBO

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

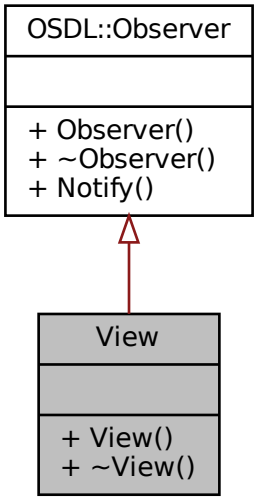
The documentation for this class was generated from the following files:

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

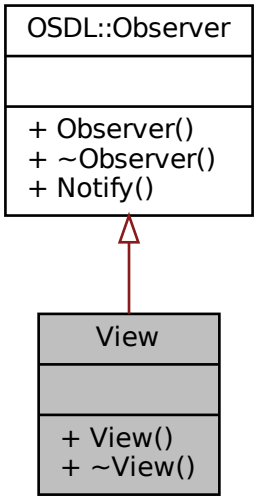
8.25 View Class Reference

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

Inheritance diagram for View:



Collaboration diagram for View:



Public Member Functions

- View ()
- ~View ()

Additional Inherited Members

8.25.1 Constructor & Destructor Documentation

8.25.1.1 View()

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

8.25.1.2 ~View()

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

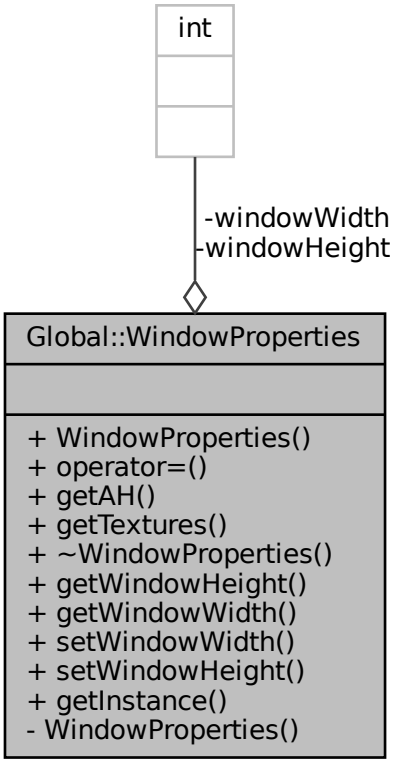
The documentation for this class was generated from the following file:

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

8.26 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

8.26.1 Constructor & Destructor Documentation

8.26.1.1 WindowProperties() [1/2]

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

8.26.1.2 WindowProperties() [2/2]

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

ArrayHandler for All GPU bindings.

Note

Return values

None	

8.26.1.3 ~WindowProperties()

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

8.26.2 Member Function Documentation

8.26.2.1 getAH()

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

8.26.2.2 getInstance()

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

8.26.2.3 getTextures()

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

8.26.2.4 getWindowHeight()

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

8.26.2.5 getWindowWidth()

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

8.26.2.6 operator=()

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

8.26.2.7 setWindowHeight()

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

8.26.2.8 setWindowWidth()

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

8.26.3 Member Data Documentation

8.26.3.1 windowHeight

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

8.26.3.2 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

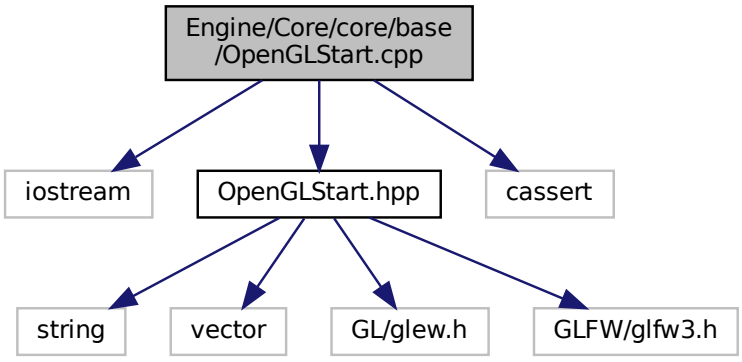
Chapter 9

File Documentation

9.1 Engine/Core/core/base/OpenGLStart.cpp File Reference

```
#include <iostream>
#include "OpenGLStart.hpp"
#include <cassert>
```

Include dependency graph for OpenGLStart.cpp:



Namespaces

- OpenGLInstance

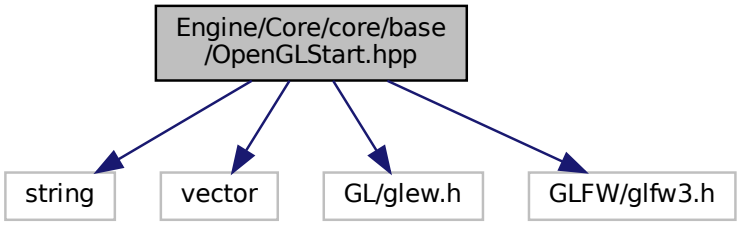
Functions

- 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)

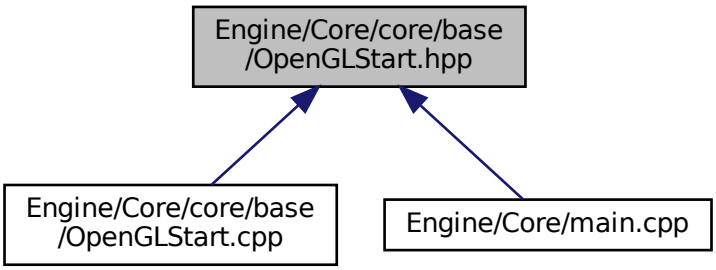
9.2 Engine/Core/core/base/OpenGLStart.hpp File Reference

```
#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

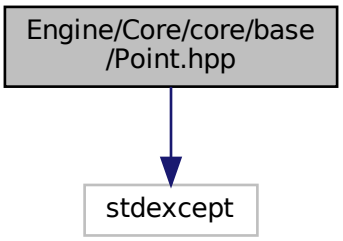
- OpenGLInstance

Functions

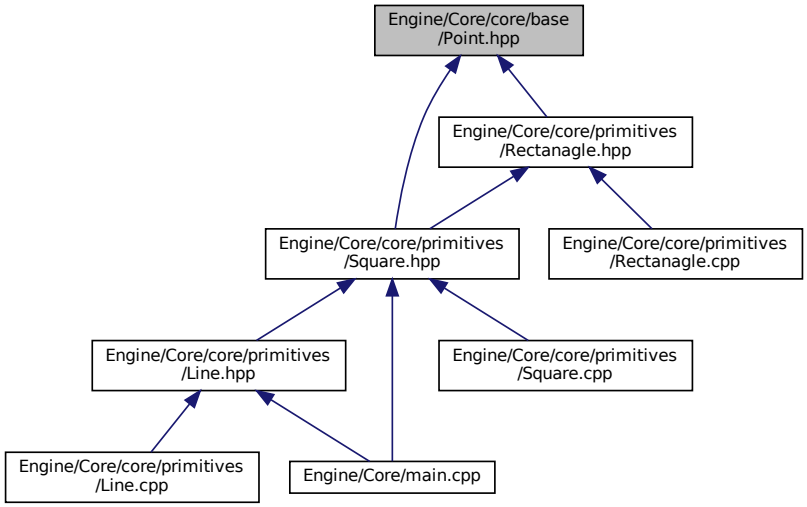
- GLFWwindow ∗ OpenGLInstance::InitWindow (int width, int height, std::string name)
- void OpenGLInstance::setClearColor (float R, float G, float B, float A)
- std::vector< GLuint > OpenGLInstance::CreateAndCompileShaders (const char ∗vertShaderCode, const char ∗fragShaderCode)
- GLuint OpenGLInstance::CreateProgramAndLinkShaders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

9.3 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::ColorRGBAU
- typedef ColorRGBA< unsigned long > Primitives::ColorRGBAUL
- typedef ColorRGBA< unsigned long long > Primitives::ColorRGBAULL
- typedef ColorRGBA< long > Primitives::ColorRGBAL
- typedef ColorRGBA< long long > Primitives::ColorRGBALL
- typedef ColorRGBA< char > Primitives::ColorRGBAC
- typedef ColorRGBA< short > Primitives::ColorRGBAS

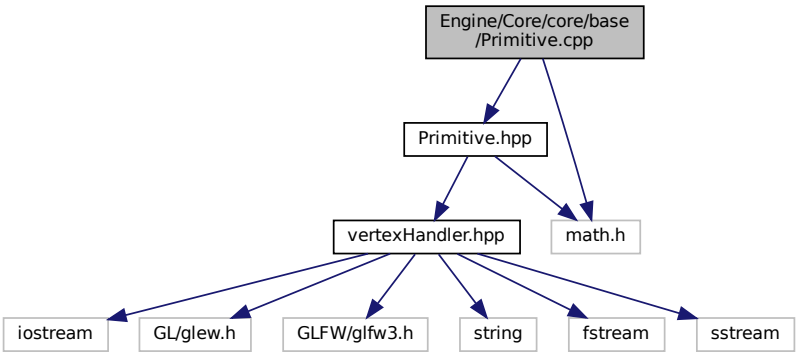
Enumerations

- enum Primitives::ColorType {
  Primitives::RED, Primitives::GREEN, Primitives::BLUE, Primitives::ALPHA,
  Primitives::ERROR }

9.4 Engine/Core/core/base/Primitive.cpp File Reference

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

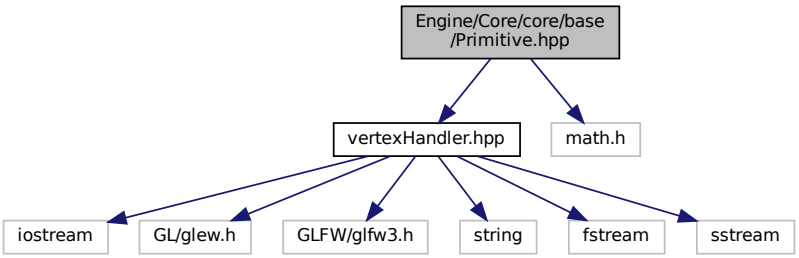
Include dependency graph for Primitive.cpp:



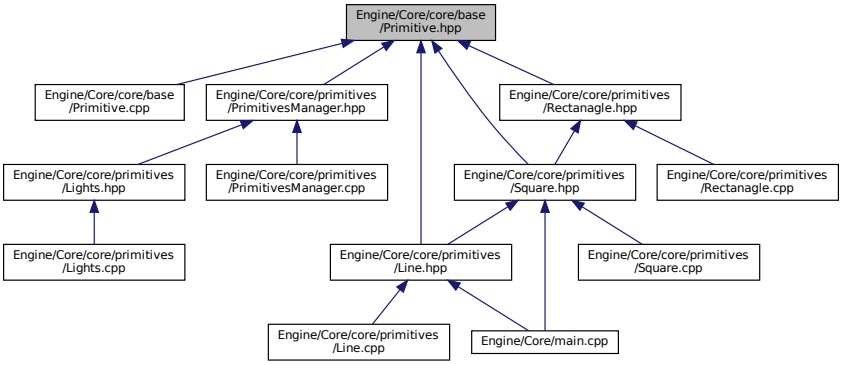
9.5 Engine/Core/core/base/Primitive.hpp File Reference

```
#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

9.5.1 Macro Definition Documentation

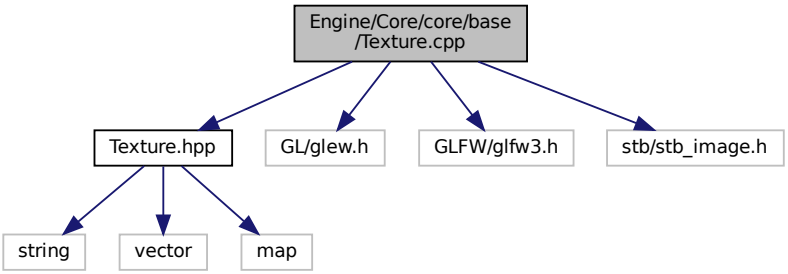
9.5.1.1 M_PI

```
#define M_PI 3.14159265358979323846
```

9.6 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

9.6.1 Macro Definition Documentation

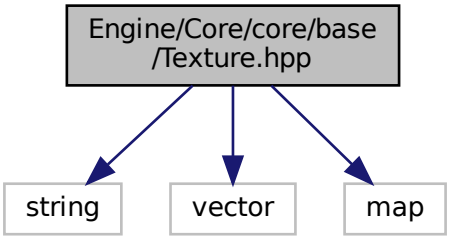
9.6.1.1 STB_IMAGE_IMPLEMENTATION

```
#define STB_IMAGE_IMPLEMENTATION
```

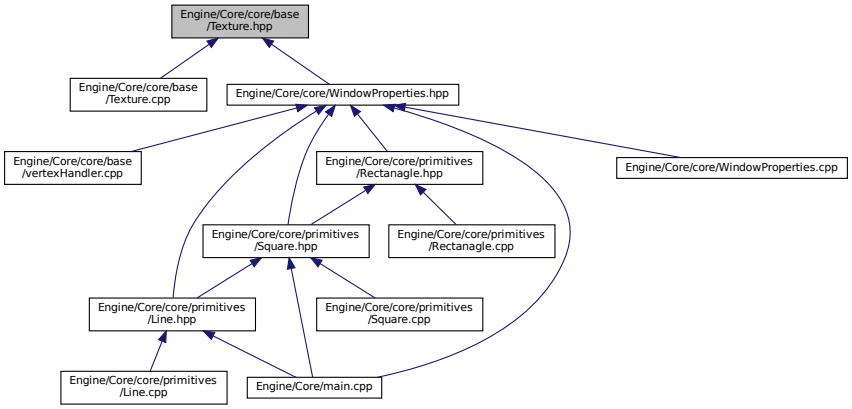
9.7 Engine/Core/core/base/Texture.hpp File Reference

```
#include <string>
#include <vector>
#include <map>
```

Include dependency graph for Texture.hpp:



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



Classes

- class AEG::Texture
- class AEG::Textures

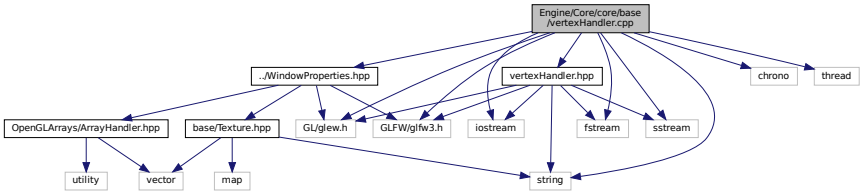
Namespaces

- AEG

9.8 Engine/Core/core/base/vertexHandler.cpp File Reference

```
#include <iostream>
#include <GL/glew.h>
#include <GLFW/glfw3.h>
#include <string>
#include <fstream>
#include <sstream>
#include "vertexHandler.hpp"
#include <chrono>
#include <thread>
#include "../WindowProperties.hpp"
```

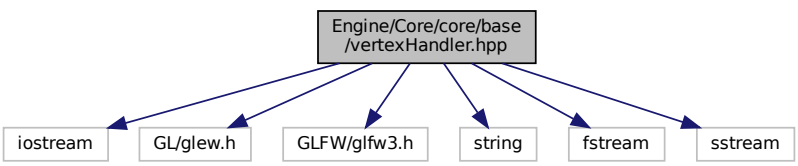
Include dependency graph for vertexHandler.cpp:



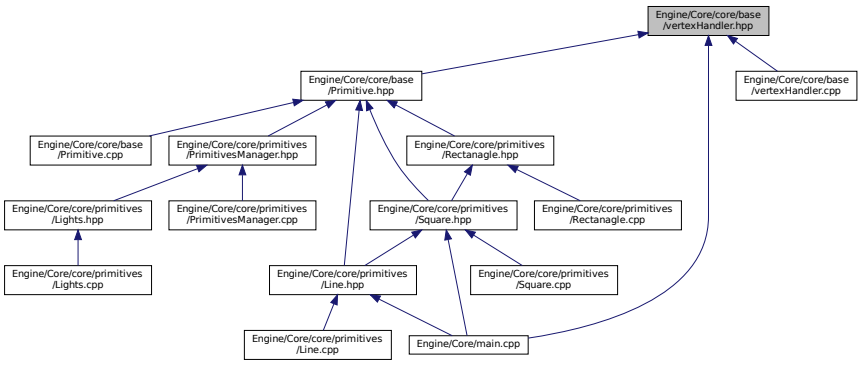
9.9 Engine/Core/core/base/vertexHandler.hpp File Reference

```
#include <iostream>
#include <GL/glew.h>
#include <GLFW/glfw3.h>
#include <string>
#include <fstream>
#include <sstream>
```

Include dependency graph for vertexHandler.hpp:



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



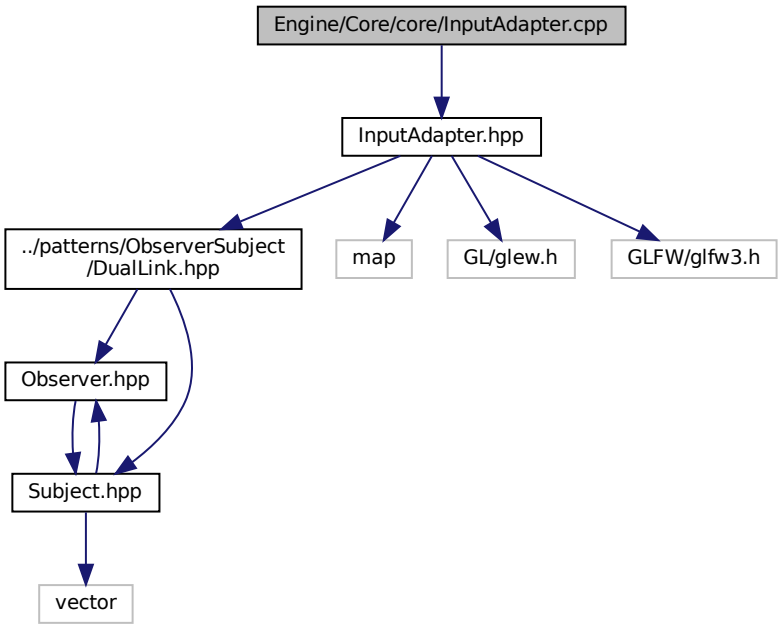
Classes

- class VertexHandler

9.10 Engine/Core/core/InputAdapter.cpp File Reference

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

Include dependency graph for InputAdapter.cpp:



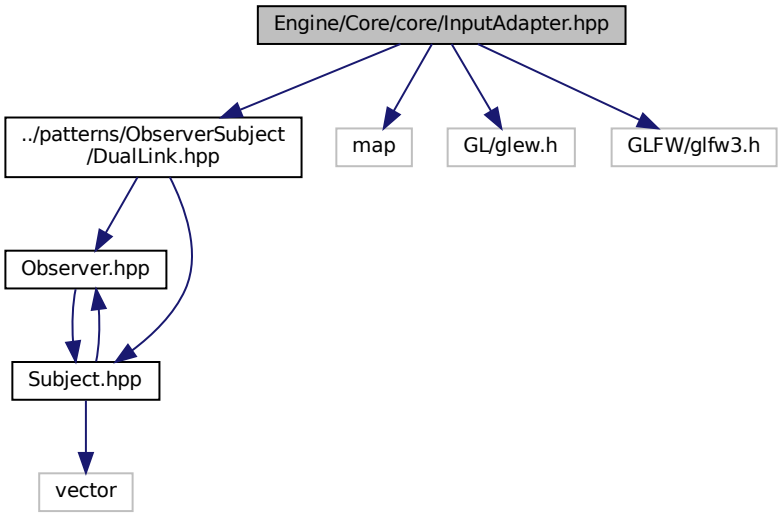
Namespaces

- INPUT

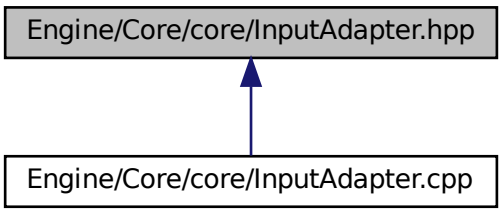
9.11 Engine/Core/core/InputAdapter.hpp File Reference

```
#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

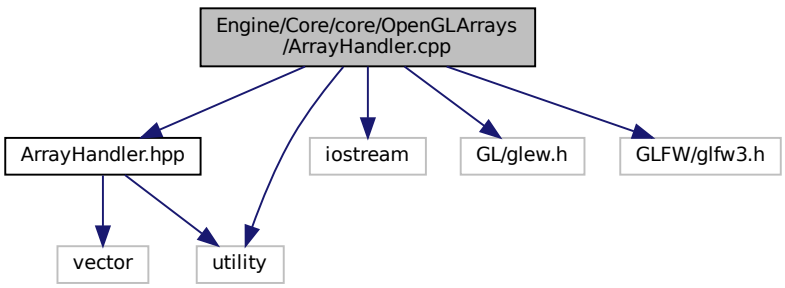
Enumerations

- 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 }

9.12 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:



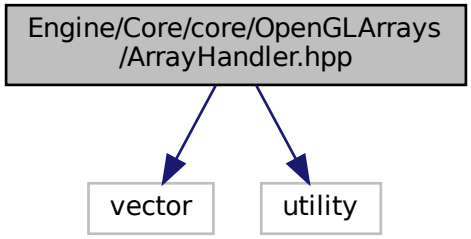
Namespaces

- GPU

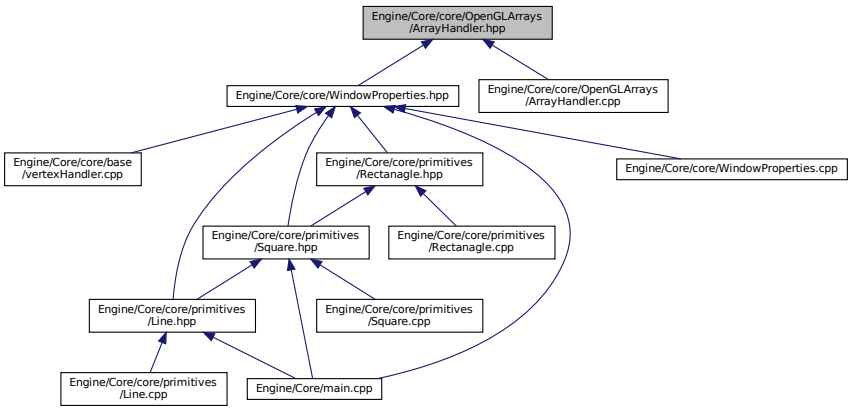
9.13 Engine/Core/core/OpenGLArrays/ArrayHandler.hpp File Reference

```
#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

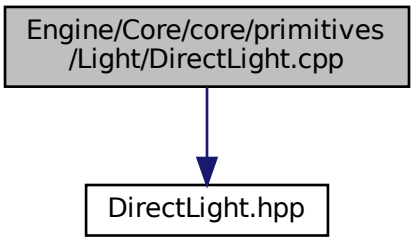
Namespaces

- GPU

9.14 Engine/Core/core/primitives/Light/DirectLight.cpp File Reference

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

Include dependency graph for DirectLight.cpp:

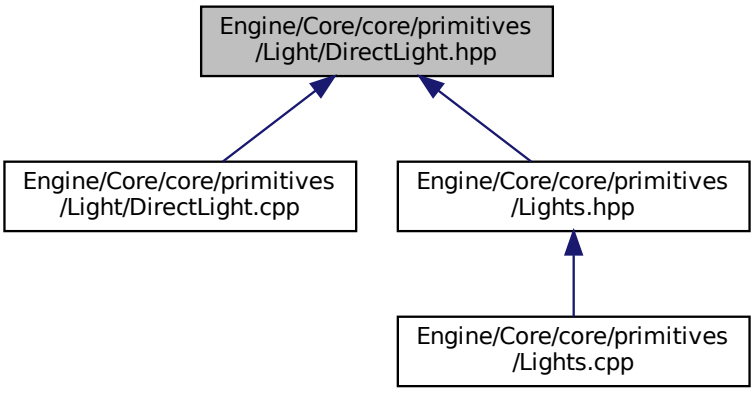


Namespaces

- Lights

9.15 Engine/Core/core/primitives/Light/DirectLight.hpp File Reference

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



Classes

- class Lights::DirectLight

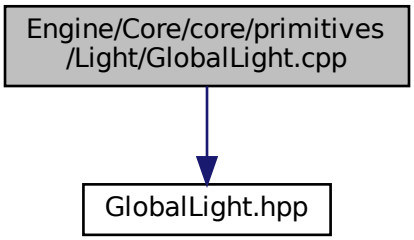
Namespaces

- Lights

9.16 Engine/Core/core/primitives/Light/GlobalLight.cpp File Reference

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

Include dependency graph for GlobalLight.cpp:

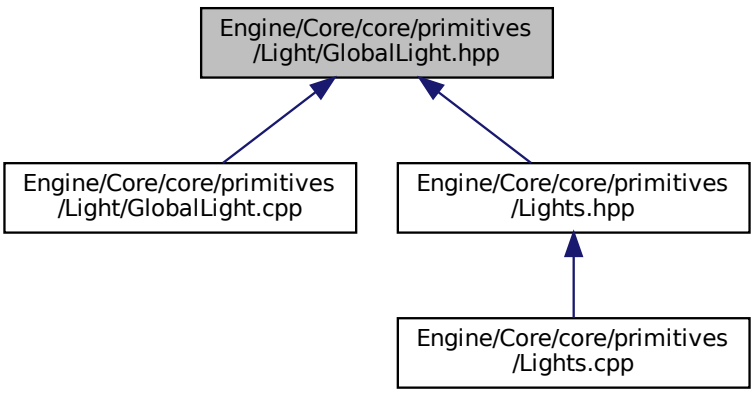


Namespaces

- Lights

9.17 Engine/Core/core/primitives/Light/GlobalLight.hpp File Reference

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



Classes

- class Lights::GlobalLight

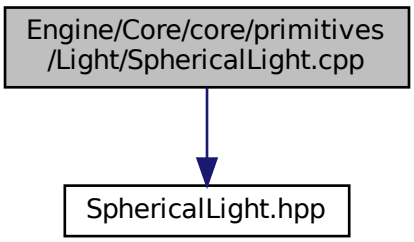
Namespaces

- Lights

9.18 Engine/Core/core/primitives/Light/SphericalLight.cpp File Reference

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

Include dependency graph for SphericalLight.cpp:

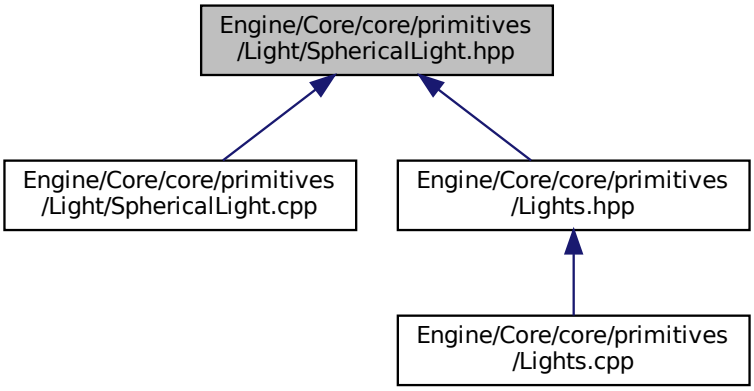


Namespaces

- Lights

9.19 Engine/Core/core/primitives/Light/SphericalLight.hpp File Reference

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



Classes

- class Lights::SphericalLight

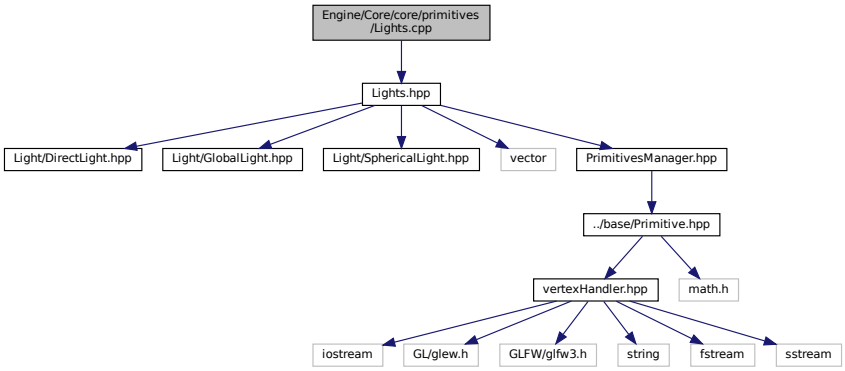
Namespaces

- Lights

9.20 Engine/Core/core/primitives/Lights.cpp File Reference

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

Include dependency graph for Lights.cpp:



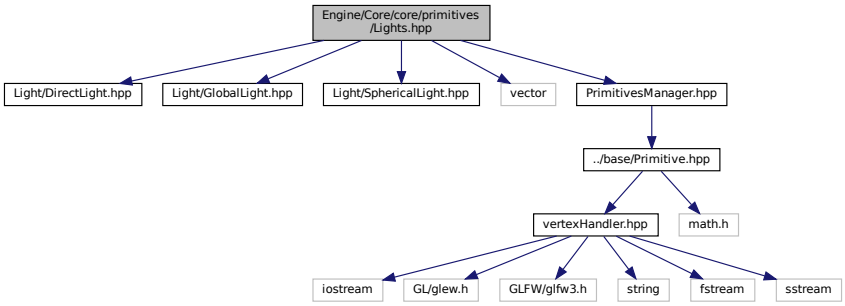
Namespaces

- Lights

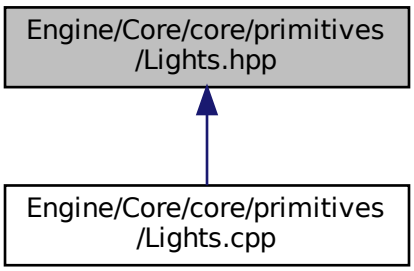
9.21 Engine/Core/core/primitives/Lights.hpp File Reference

```
#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

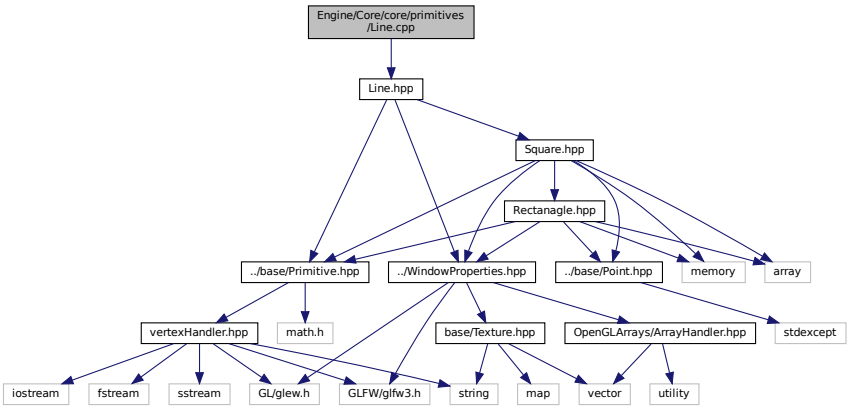
Namespaces

- Lights

9.22 Engine/Core/core/primitives/Line.cpp File Reference

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

Include dependency graph for Line.cpp:



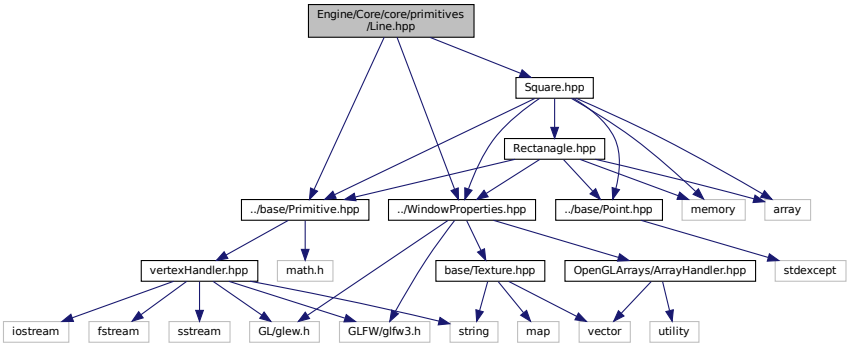
Namespaces

- Primitives

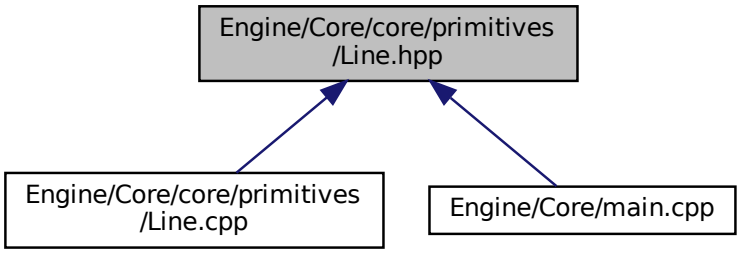
9.23 Engine/Core/core/primitives/Line.hpp File Reference

```
#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

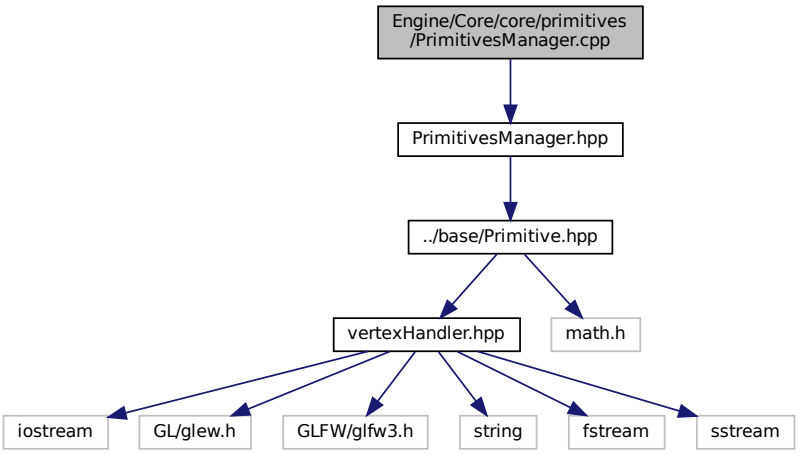
- class Primitives::Line

Namespaces

- Primitives

9.24 Engine/Core/core/primitives/PrimitivesManager.cpp File Reference

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



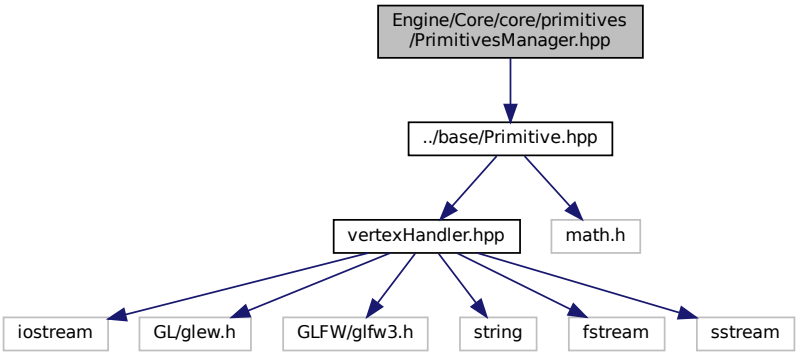
Namespaces

- Primitives

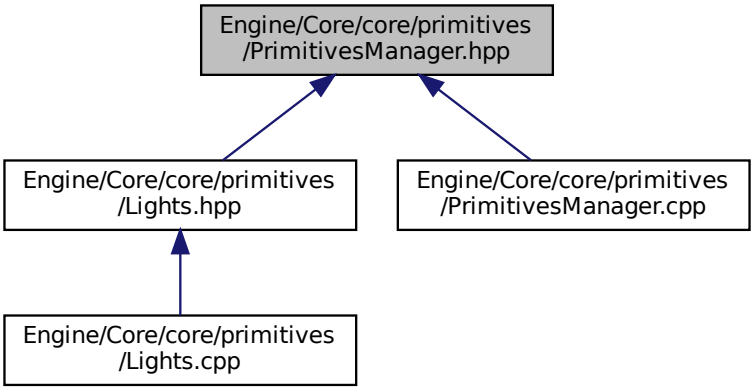
9.25 Engine/Core/core/primitives/PrimitivesManager.hpp File Reference

```
#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

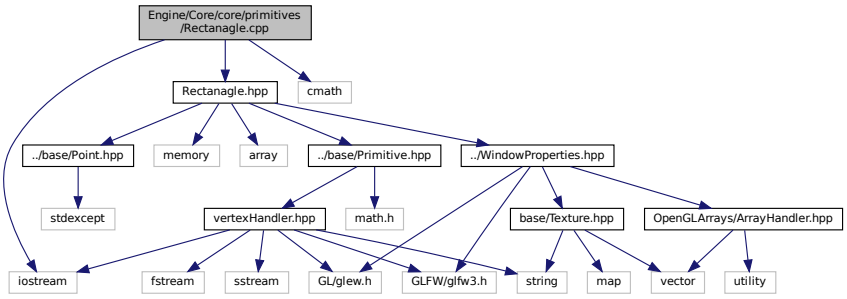
Typedefs

- typedef Primitive * Primitives::PrimitivePtr
- typedef std::vector< PrimitivePtr > Primitives::Primitives
- typedef Primitives & Primitives::PrimitivesRef

9.26 Engine/Core/core/primitives/Rectanagle.cpp File Reference

```
#include "Rectanagle.hpp"
#include <iostream>
#include <cmath>
```

Include dependency graph for Rectanagle.cpp:



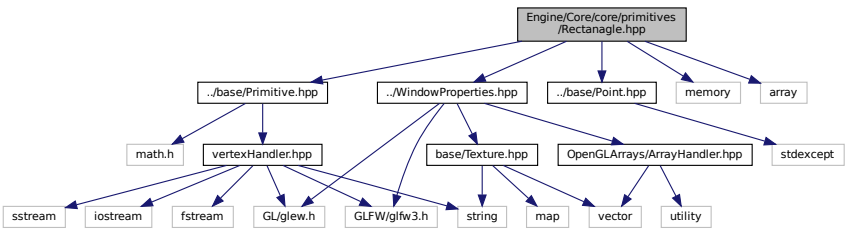
Namespaces

- Primitives

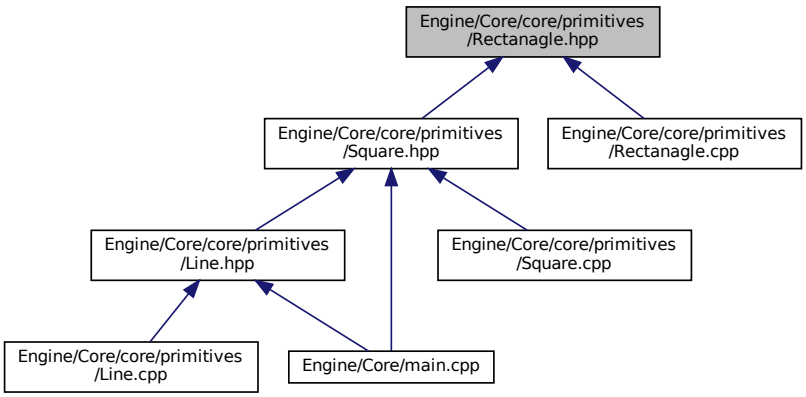
9.27 Engine/Core/core/primitives/Rectanagle.hpp File Reference

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

Include dependency graph for Rectanagle.hpp:



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



Classes

- class Primitives::Rectanagle

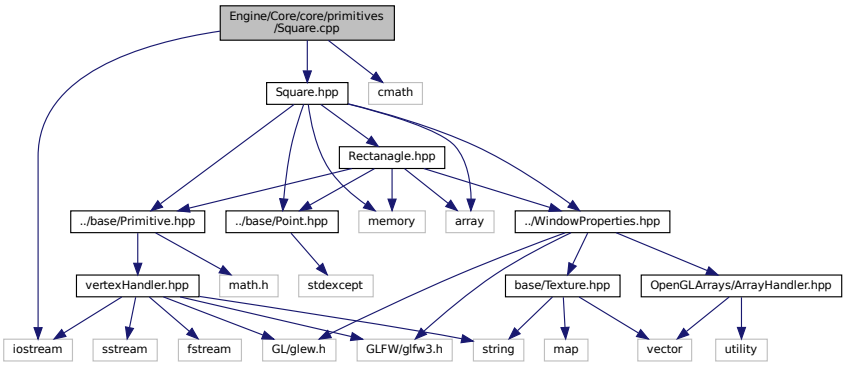
Namespaces

- Primitives

9.28 Engine/Core/core/primitives/Square.cpp File Reference

```
#include "Square.hpp"
#include <iostream>
#include <cmath>
```

Include dependency graph for Square.cpp:



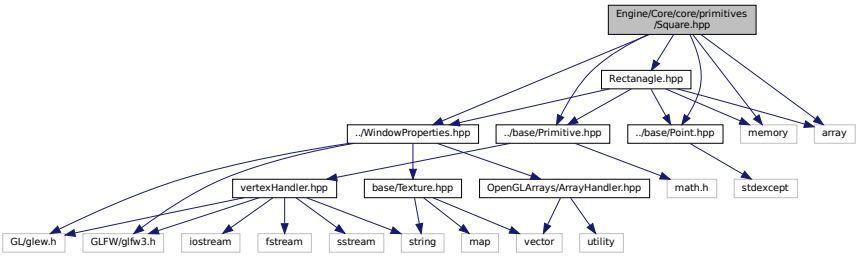
Namespaces

- Primitives

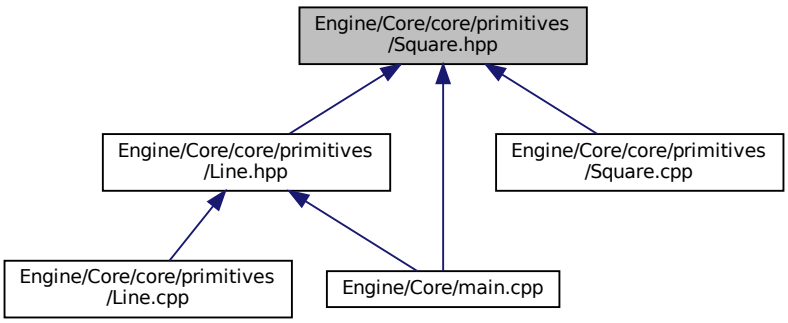
9.29 Engine/Core/core/primitives/Square.hpp File Reference

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

Include dependency graph for Square.hpp:



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



Classes

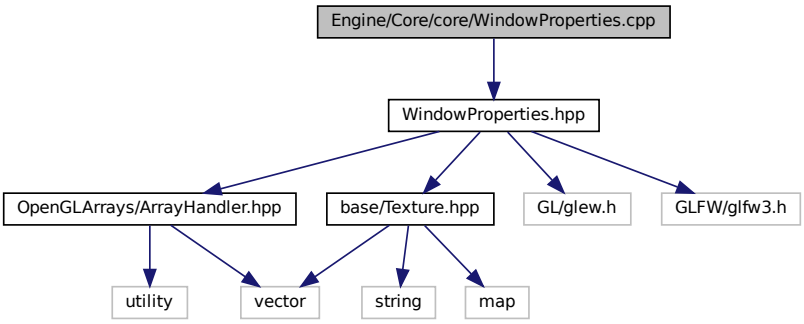
- class Primitives::Square

Namespaces

- Primitives

9.30 Engine/Core/core/WindowProperties.cpp File Reference

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

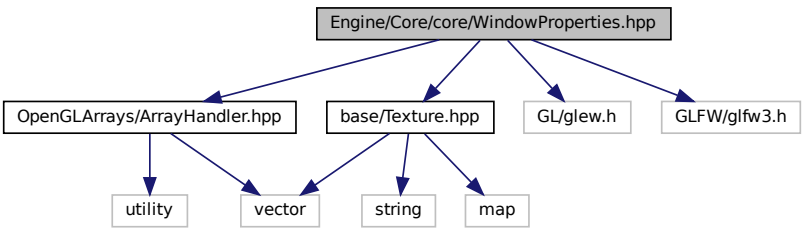


Namespaces

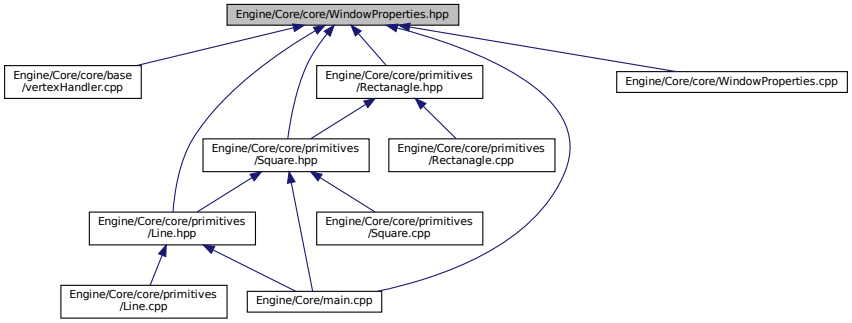
- Global

9.31 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:



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



Classes

- class Global::WindowProperties

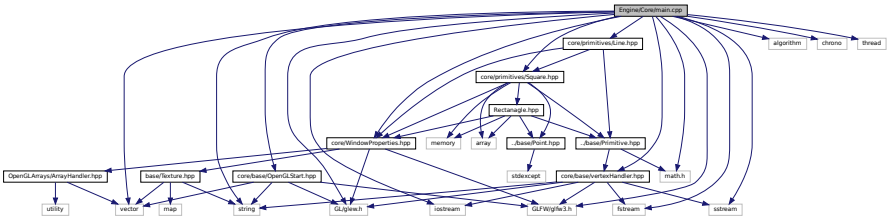
Namespaces

- Global

9.32 Engine/Core/main.cpp File Reference

```
#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>
```

Include dependency graph for main.cpp:



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

9.32.1 Macro Definition Documentation

9.32.1.1 STB_IMAGE_IMPLEMENTATION

```
#define STB_IMAGE_IMPLEMENTATION
```

9.32.2 Function Documentation

9.32.2.1 main()

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

9.32.2.2 onKeyCallback()

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

9.32.2.3 onKeyPress()

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

9.32.2.4 readFile()

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

9.32.3 Variable Documentation

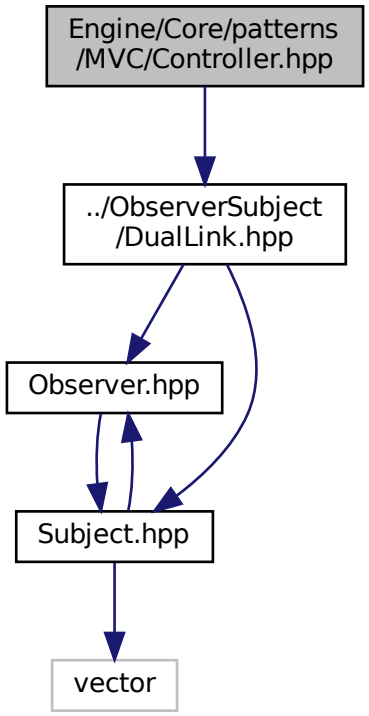
9.32.3.1 player

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

9.33 Engine/Core/patterns/MVC/Controller.hpp File Reference

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

Include dependency graph for Controller.hpp:

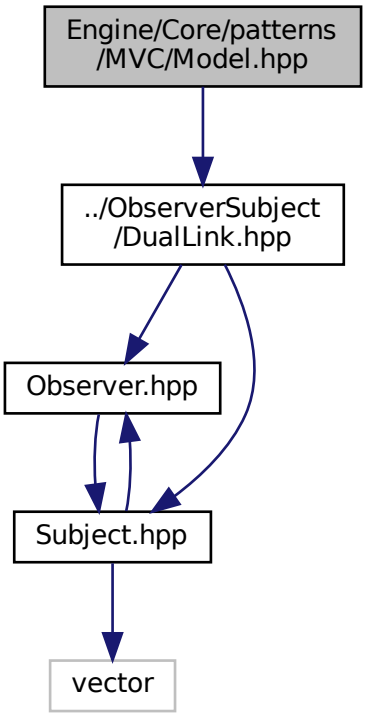


Classes

- class Controller

9.34 Engine/Core/patterns/MVC/Model.hpp File Reference

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



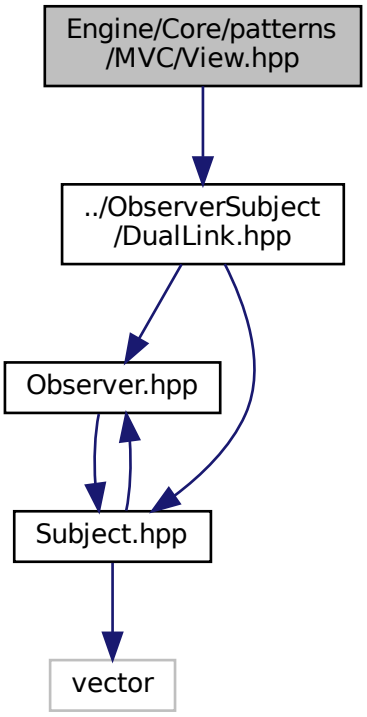
Classes

- class Model

9.35 Engine/Core/patterns/MVC/View.hpp File Reference

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

Include dependency graph for View.hpp:



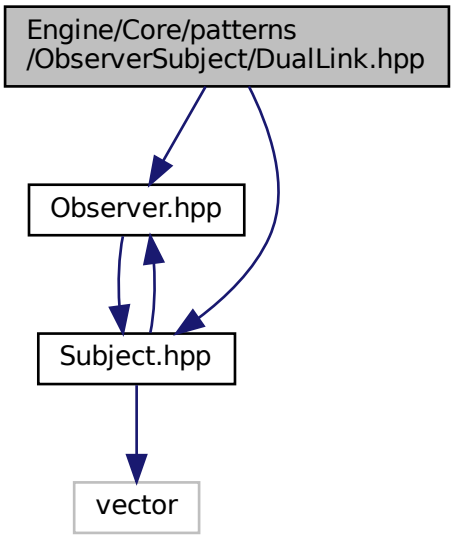
Classes

- class View

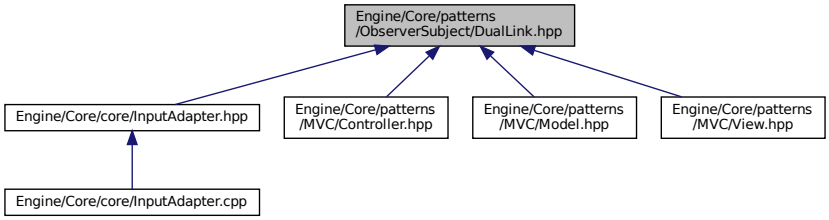
9.36 Engine/Core/patterns/ObserverSubject/DualLink.hpp File Reference

```
#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

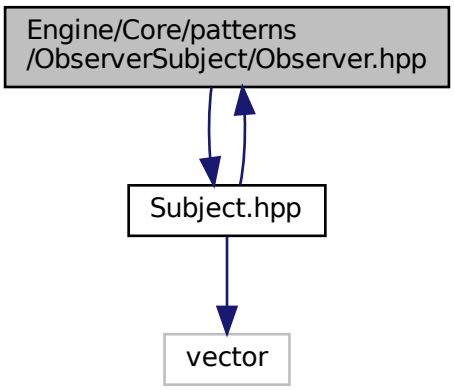
Namespaces

- OSDL

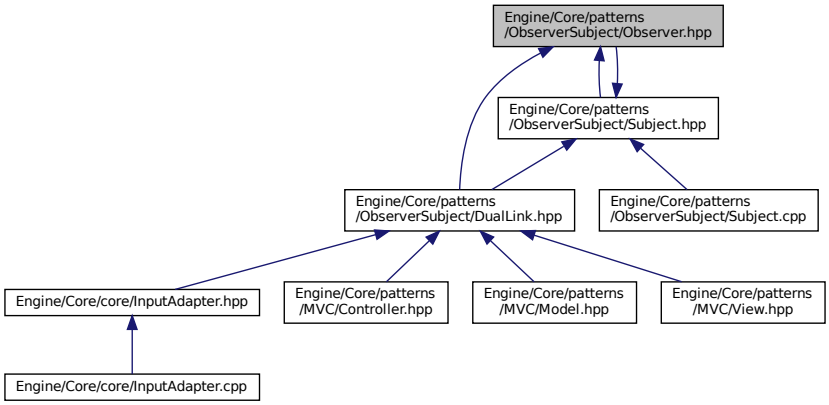
9.37 Engine/Core/patterns/ObserverSubject/Observer.hpp File Reference

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

Include dependency graph for Observer.hpp:



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



Classes

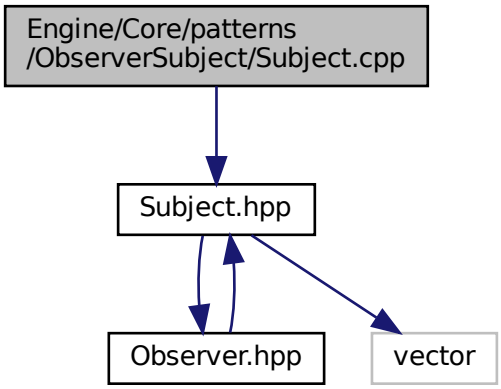
- class OSDL::Observer

Namespaces

- OSDL

9.38 Engine/Core/patterns/ObserverSubject/Subject.cpp File Reference

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

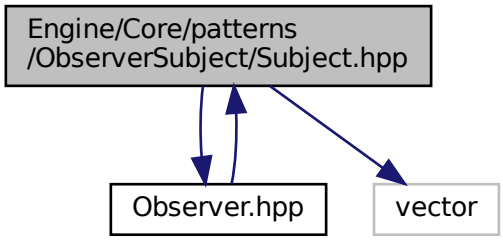


Namespaces

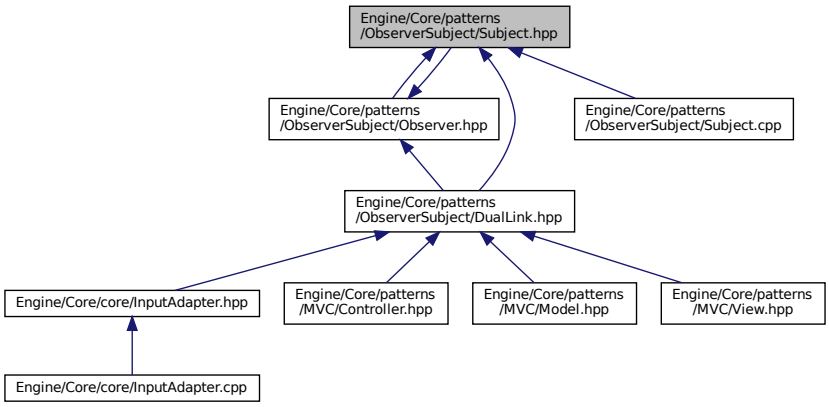
- OSDL

9.39 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

9.40 Engine/Core/README.md File Reference

9.41 README.md File Reference

Index