

## R-Type architecture

Generated by Doxygen 1.13.2



---

<b>1 Hierarchical Index</b>	<b>1</b>
1.1 Class Hierarchy . . . . .	1
<b>2 Class Index</b>	<b>3</b>
2.1 Class List . . . . .	3
<b>3 File Index</b>	<b>5</b>
3.1 File List . . . . .	5
<b>4 Class Documentation</b>	<b>7</b>
4.1 color_t Struct Reference . . . . .	7
4.2 DLLoader< T > Class Template Reference . . . . .	7
4.2.1 Member Function Documentation . . . . .	8
4.2.1.1 Close() . . . . .	8
4.2.1.2 Error() . . . . .	8
4.2.1.3 getHandler() . . . . .	8
4.2.1.4 Open() . . . . .	8
4.2.1.5 Symbol() . . . . .	8
4.3 IEvent Class Reference . . . . .	9
4.4 ILoader Class Reference . . . . .	9
4.5 IWindow Class Reference . . . . .	10
<b>5 File Documentation</b>	<b>11</b>
5.1 IEvent.hpp . . . . .	11
5.2 IWindow.hpp . . . . .	11
5.3 DLLoader.hpp . . . . .	12
5.4 ILoader.hpp . . . . .	13
5.5 LoaderType.hpp . . . . .	13
<b>Index</b>	<b>15</b>



# Chapter 1

## Hierarchical Index

### 1.1 Class Hierarchy

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

color_t . . . . .	7
IEvent . . . . .	9
ILoader . . . . .	9
DLLoader< T > . . . . .	7
IWindow . . . . .	10



## Chapter 2

# Class Index

### 2.1 Class List

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

<a href="#">color_t</a> . . . . .	7
<a href="#">DLLoader&lt; T &gt;</a> . . . . .	7
<a href="#">IEvent</a> . . . . .	9
<a href="#">ILoader</a> . . . . .	9
<a href="#">IWindow</a> . . . . .	10





## Chapter 3

# File Index

### 3.1 File List

Here is a list of all documented files with brief descriptions:

/home/albane/epitech/tech3/r-type/copyDocu/ryanR-type/client/graphics/ <a href="#">IEvent.hpp</a> . . . . .	11
/home/albane/epitech/tech3/r-type/copyDocu/ryanR-type/client/graphics/ <a href="#">IWindow.hpp</a> . . . . .	11
/home/albane/epitech/tech3/r-type/copyDocu/ryanR-type/common/ <a href="#">DLLoader.hpp</a> . . . . .	12
/home/albane/epitech/tech3/r-type/copyDocu/ryanR-type/common/ <a href="#">ILoader.hpp</a> . . . . .	13
/home/albane/epitech/tech3/r-type/copyDocu/ryanR-type/common/ <a href="#">LoaderType.hpp</a> . . . . .	13



## Chapter 4

# Class Documentation

### 4.1 color\_t Struct Reference

#### Public Attributes

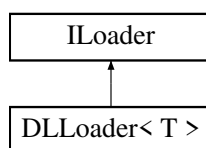
- int **r**
- int **g**
- int **b**

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

- /home/albane/epitech/tech3/r-type/copyDocu/ryanR-type/client/graphics/IWindow.hpp

### 4.2 DLLoader< T > Class Template Reference

Inheritance diagram for DLLoader< T >:



#### Public Member Functions

- void \* [getHandler](#) () const override
- void \* [Open](#) (const char \*path, int flag=RTLD\_LAZY) override
- void \* [Symbol](#) (const char \*symbolName) override
- T [getSymbol](#) (const char \*symbolName)
- int [Close](#) () override
- const char \* [Error](#) () override

## Private Attributes

- void \* `_handler` = nullptr

## 4.2.1 Member Function Documentation

### 4.2.1.1 Close()

```
template<typename T>
int DLLoader< T >::Close () [inline], [override], [virtual]
```

Implements [ILoader](#).

### 4.2.1.2 Error()

```
template<typename T>
const char * DLLoader< T >::Error () [inline], [override], [virtual]
```

Implements [ILoader](#).

### 4.2.1.3 getHandler()

```
template<typename T>
void * DLLoader< T >::getHandler () const [inline], [override], [virtual]
```

Implements [ILoader](#).

### 4.2.1.4 Open()

```
template<typename T>
void * DLLoader< T >::Open (
    const char * path,
    int flag = RTLD_LAZY) [inline], [override], [virtual]
```

Implements [ILoader](#).

### 4.2.1.5 Symbol()

```
template<typename T>
void * DLLoader< T >::Symbol (
    const char * symbolName) [inline], [override], [virtual]
```

Implements [ILoader](#).

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

- /home/albane/epitech/tech3/r-type/copyDocu/ryanR-type/common/DLLoader.hpp

## 4.3 IEvent Class Reference

### Public Types

- enum **event\_e** {  
**UP** , **DOWN** , **LEFT** , **RIGHT** ,  
**SPACE** , **ENTER** , **ESCAPE** , **CLOSE** ,  
**NEXTGAME** , **NEXTGRAPHIC** , **REFRESH** , **MOUSECLICK** ,  
**MOUSERIGHTCLICK** , **MOUSELEFTCLICK** , **MENU** , **NOTHING** ,  
**TYIPING** , **NEXTDIFFICULTY** }
- typedef enum IEvent::event\_e **event\_t**

### Public Member Functions

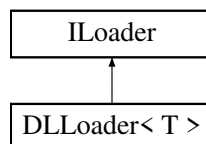
- virtual void **init** ()=0
- virtual event\_t **pollEvents** (std::pair< int, int > gridSize)=0
- virtual void **cleanup** ()=0
- virtual std::pair< int, int > **getMousePos** ()=0
- virtual void **setMapSize** (std::pair< int, int > size)=0

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

- /home/albane/epitech/tech3/r-type/copyDocu/ryanR-type/client/graphics/IEvent.hpp

## 4.4 ILoader Class Reference

Inheritance diagram for ILoader:



### Public Member Functions

- virtual void \* **Open** (const char \*path, int flag)=0
- virtual void \* **Symbol** (const char \*symbolName)=0
- virtual int **Close** ()=0
- virtual const char \* **Error** ()=0
- virtual void \* **getHandler** () const =0

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

- /home/albane/epitech/tech3/r-type/copyDocu/ryanR-type/common/ILoader.hpp

## 4.5 IWindow Class Reference

### Public Member Functions

- virtual void **display** ()=0
- virtual void **closeWindow** ()=0
- virtual bool **isOpen** ()=0
- virtual void **clear** ()=0
- virtual void **setMapSize** (std::pair< size\_t, size\_t > size)=0
- virtual void **resizeWindow** (size\_t x, size\_t y)=0
- virtual void **drawSprite** (std::string asset, int color, std::string text, std::pair< size\_t, size\_t > position)=0
- virtual void **drawText** (std::string text, int color, std::pair< size\_t, size\_t > position)=0
- virtual void **drawRectangle** (int color, std::pair< size\_t, size\_t > position)=0
- virtual bool **isMouseOver** (std::pair< size\_t, size\_t > position, std::pair< size\_t, size\_t > size)=0
- virtual std::pair< int, int > **getWindowSize** ()=0

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

- /home/albane/epitech/tech3/r-type/copyDocu/ryanR-type/client/graphics/IWindow.hpp

# Chapter 5

## File Documentation

### 5.1 IEvent.hpp

```
00001 /*
00002 ** EPITECH PROJECT, 2025
00003 ** r-type
00004 ** File description:
00005 ** IEvent
00006 */
00007
00008 #ifndef IEVENT_HPP_
00009 #define IEVENT_HPP_
00010
00011 #include <utility>
00012 #include <memory>
00013
00014 class IEvent {
00015     public:
00016         typedef enum event_e {
00017             UP,
00018             DOWN,
00019             LEFT,
00020             RIGHT,
00021             SPACE,
00022             ENTER,
00023             ESCAPE,
00024             CLOSE,
00025             NEXTGAME,
00026             NEXTGRAPHIC,
00027             REFRESH,
00028             MOUSECLICK,
00029             MOUSERIGHTCLICK,
00030             MOUSELEFTCLICK,
00031             MENU,
00032             NOTHING,
00033             TYPING,
00034             NEXTDIFFICULTY
00035         } event_t;
00036         virtual ~IEvent() = default;
00037         virtual void init() = 0;
00038         virtual event_t pollEvents(std::pair<int, int> gridSize) = 0;
00039         virtual void cleanup() = 0;
00040         virtual std::pair<int, int> getMousePos() = 0;
00041         virtual void setMapSize(std::pair<int, int> size) = 0;
00042     };
00043 };
00044
00045 #endif /* !IEVENT_HPP_ */
```

### 5.2 IWindow.hpp

```
00001 /*
00002 ** EPITECH PROJECT, 2025
00003 ** r-type
00004 ** File description:
00005 ** IWindow
```

```

00006 */
00007
00008
00009 #ifndef IWINDOW_HPP_
00010 #define IWINDOW_HPP_
00011
00012 #include <string>
00013 #include <utility>
00014 #include <memory>
00015
00016 struct color_t {
00017     int r;
00018     int g;
00019     int b;
00020 };
00021
00022 class IWindow {
00023     public:
00024         virtual ~IWindow() = default;
00025         virtual void display() = 0;
00026         virtual void closeWindow() = 0;
00027         virtual bool isOpen() = 0;
00028         virtual void clear() = 0;
00029
00030         virtual void setMapSize(std::pair<size_t, size_t> size) = 0;
00031         virtual void resizeWindow(size_t x, size_t y) = 0;
00032
00033         virtual void drawSprite(std::string asset, int color, std::string text, std::pair<size_t,
size_t>
position) = 0;
00034         virtual void drawText(std::string text, int color, std::pair<size_t, size_t> position) = 0;
00035         virtual void drawRectangle(int color, std::pair<size_t, size_t> position) = 0;
00036
00037         virtual bool isMouseOver(std::pair<size_t, size_t> position, std::pair<size_t, size_t> size) =
0;
00038         virtual std::pair<int, int> getWindowSize() = 0;
00039 };
00040
00041 #endif /* !IWINDOW_HPP_ */

```

## 5.3 DLLoader.hpp

```

00001 /*
00002 ** EPITECH PROJECT, 2025
00003 ** r-type
00004 ** File description:
00005 ** DLLoader
00006 */
00007
00008 #ifndef DLOADER_HPP_
00009 #define DLOADER_HPP_
00010
00011 #include <dlfcn.h>
00012 #include <iostream>
00013 #include <ostream>
00014 #include <memory>
00015 #include "ILoader.hpp"
00016
00017 template <typename T>
00018
00019 class DLLoader : public ILoader {
00020     private:
00021         void *_handler = nullptr;
00022
00023     public:
00024         ~DLLoader() = default;
00025
00026         void *getHandler() const override {
00027             return _handler;
00028         };
00029         void *Open(const char *path, int flag = RTLD_LAZY) override {
00030             _handler = dlopen(path, flag);
00031             return _handler;
00032         };
00033         void *Symbol(const char *symbolName) override {
00034             void *symbol = dlsym(_handler, symbolName);
00035             const char *error = dlerror();
00036             if (error) {
00037                 std::cerr << "dlerror: " << error << std::endl;
00038                 return nullptr;
00039             }
00040             return symbol;
00041         };
00042         T getSymbol(const char *symbolName) {

```



```

00043         return reinterpret_cast<T>(dlsym(_handler, symbolName));
00044     };
00045     int Close() override{
00046         if (_handler == nullptr)
00047             return -1;
00048         return dlclose(_handler);
00049     };
00050     const char *Error() override {
00051         return dlerror();
00052     };
00053 };
00054
00055 #endif /* !DLLOADER_HPP_ */

```

## 5.4 ILoader.hpp

```

00001 /*
00002 ** EPITECH PROJECT, 2025
00003 ** r-type
00004 ** File description:
00005 ** ILoader
00006 */
00007
00008 #ifndef ILoader_HPP_
00009 #define ILoader_HPP_
00010
00011
00012 class ILoader {
00013     public:
00014         ~ILoader() = default;
00015
00016         virtual void *Open(const char *path, int flag) = 0;
00017         virtual void *Symbol(const char *symbolName) = 0;
00018         virtual int Close() = 0;
00019         virtual const char *Error() = 0;
00020         virtual void *getHandler() const = 0;
00021
00022     protected:
00023     private:
00024 };
00025
00026 #endif /* !ILoader_HPP_ */

```

## 5.5 LoaderType.hpp

```

00001 /*
00002 ** EPITECH PROJECT, 2025
00003 ** r-type
00004 ** File description:
00005 ** LoaderType
00006 */
00007
00008 #ifndef LOADERTYPE_HPP_
00009 #define LOADERTYPE_HPP_
00010
00011 enum ModuleType_t{
00012     DISPLAY_MODULE,
00013     NONE
00014 };
00015
00016 typedef ModuleType_t (*getTypeFunc_t)();
00017
00018 #endif /* !LOADERTYPE_HPP_ */

```



# Index

[/home/albane/epitech/tech3/r-type/copyDocu/ryanR-type/client/graphics/IEvent.hpp](#), [11](#)  
[/home/albane/epitech/tech3/r-type/copyDocu/ryanR-type/client/graphics/IWindow.hpp](#), [11](#)  
[/home/albane/epitech/tech3/r-type/copyDocu/ryanR-type/common/DLLoader.hpp](#), [12](#)  
[/home/albane/epitech/tech3/r-type/copyDocu/ryanR-type/common/ILoader.hpp](#), [13](#)  
[/home/albane/epitech/tech3/r-type/copyDocu/ryanR-type/common/LoaderType.hpp](#), [13](#)

## Close

[DLLoader< T >](#), [8](#)  
[color\\_t](#), [7](#)

## [DLLoader< T >](#), [7](#)

[Close](#), [8](#)  
[Error](#), [8](#)  
[getHandler](#), [8](#)  
[Open](#), [8](#)  
[Symbol](#), [8](#)

## Error

[DLLoader< T >](#), [8](#)

## getHandler

[DLLoader< T >](#), [8](#)

## [IEvent](#), [9](#)

## [ILoader](#), [9](#)

## [IWindow](#), [10](#)

## Open

[DLLoader< T >](#), [8](#)

## Symbol

[DLLoader< T >](#), [8](#)