

## Bachelorthesis - Multi-/Remotemaus

Generated by Doxygen 1.8.13



# Contents

<b>1</b>	<b>Class Index</b>	<b>1</b>
1.1	Class List . . . . .	1
<b>2</b>	<b>File Index</b>	<b>3</b>
2.1	File List . . . . .	3
<b>3</b>	<b>Class Documentation</b>	<b>5</b>
3.1	input_event Struct Reference . . . . .	5
3.1.1	Detailed Description . . . . .	5
3.1.2	Member Data Documentation . . . . .	5
3.1.2.1	code . . . . .	5
3.1.2.2	time . . . . .	5
3.1.2.3	type . . . . .	6
3.1.2.4	value . . . . .	6
3.2	mrmCfgParser Class Reference . . . . .	6
3.2.1	Constructor & Destructor Documentation . . . . .	6
3.2.1.1	mrmCfgParser() . . . . .	6
3.2.2	Member Function Documentation . . . . .	7
3.2.2.1	createLocalCursors() . . . . .	7
3.2.2.2	createLocalMice() . . . . .	7
3.2.2.3	createRemoteCursors() . . . . .	7
3.2.2.4	createScreens() . . . . .	8
3.2.2.5	getHostIP() . . . . .	8
3.2.2.6	getUDPPort() . . . . .	8

3.2.2.7	<a href="#">parse()</a>	9
3.2.3	<a href="#">Member Data Documentation</a>	9
3.2.3.1	<a href="#">jsonobj</a>	9
3.2.3.2	<a href="#">jsonstr</a>	9
3.3	<a href="#">mrmCursor Class Reference</a>	9
3.3.1	<a href="#">Constructor &amp; Destructor Documentation</a>	10
3.3.1.1	<a href="#">mrmCursor()</a>	11
3.3.2	<a href="#">Member Function Documentation</a>	12
3.3.2.1	<a href="#">activate()</a>	12
3.3.2.2	<a href="#">createTheRenderContext()</a>	12
3.3.2.3	<a href="#">createTheWindow()</a>	12
3.3.2.4	<a href="#">describe_fbconfig()</a>	12
3.3.2.5	<a href="#">getDisplay()</a>	13
3.3.2.6	<a href="#">getid()</a>	13
3.3.2.7	<a href="#">getPosx()</a>	13
3.3.2.8	<a href="#">getPosy()</a>	13
3.3.2.9	<a href="#">getrelxfromabs()</a>	13
3.3.2.10	<a href="#">getrelyfromabs()</a>	14
3.3.2.11	<a href="#">getXMax()</a>	14
3.3.2.12	<a href="#">getYMax()</a>	14
3.3.2.13	<a href="#">isactive()</a>	14
3.3.2.14	<a href="#">redrawTheWindow()</a>	15
3.3.2.15	<a href="#">run()</a>	15
3.3.2.16	<a href="#">setlastabsx()</a>	15
3.3.2.17	<a href="#">setlastabsy()</a>	15
3.3.2.18	<a href="#">setPosx()</a>	15
3.3.2.19	<a href="#">setPosy()</a>	16
3.3.2.20	<a href="#">start()</a>	16
3.3.2.21	<a href="#">updateTheMessageQueue()</a>	16
3.3.3	<a href="#">Member Data Documentation</a>	16

3.3.3.1	active	16
3.3.3.2	click	16
3.3.3.3	clickrunning	17
3.3.3.4	cmap	17
3.3.3.5	color_b	17
3.3.3.6	color_g	17
3.3.3.7	color_r	17
3.3.3.8	datatf	17
3.3.3.9	del_atom	17
3.3.3.10	fbconfig	17
3.3.3.11	fbconfigs	18
3.3.3.12	glX_window_handle	18
3.3.3.13	height	18
3.3.3.14	id	18
3.3.3.15	lastabsx	18
3.3.3.16	lastabsy	18
3.3.3.17	numfbconfigs	18
3.3.3.18	pict_format	18
3.3.3.19	posx	19
3.3.3.20	posy	19
3.3.3.21	relx	19
3.3.3.22	rely	19
3.3.3.23	render_context	19
3.3.3.24	show	19
3.3.3.25	VisData	19
3.3.3.26	visual	20
3.3.3.27	width	20
3.3.3.28	window_handle	20
3.3.3.29	Xdisplay	20
3.3.3.30	Xroot	20

3.3.3.31	Xscreen	20
3.4	mrmCursorControl Class Reference	20
3.4.1	Constructor & Destructor Documentation	21
3.4.1.1	mrmCursorControl()	21
3.4.2	Member Function Documentation	21
3.4.2.1	getlist()	21
3.4.2.2	start()	21
3.4.2.3	work_event()	21
3.4.3	Member Data Documentation	22
3.4.3.1	amount	22
3.4.3.2	amountlocal	22
3.4.3.3	cursors	22
3.4.3.4	oscursor	22
3.5	mrmMiceControl Class Reference	22
3.5.1	Constructor & Destructor Documentation	23
3.5.1.1	mrmMiceControl()	23
3.5.2	Member Function Documentation	23
3.5.2.1	start()	23
3.5.3	Member Data Documentation	23
3.5.3.1	amount	23
3.5.3.2	mice	24
3.6	mrmMouse Class Reference	24
3.6.1	Constructor & Destructor Documentation	24
3.6.1.1	mrmMouse()	24
3.6.2	Member Function Documentation	25
3.6.2.1	run()	25
3.6.2.2	start()	25
3.6.3	Member Data Documentation	25
3.6.3.1	eventfile	25
3.6.3.2	id	25

3.6.3.3	<a href="#">ieMouse</a>	25
3.6.3.4	<a href="#">mousefile</a>	26
3.7	<a href="#">mrmMultiRemoteMouse Class Reference</a>	26
3.7.1	<a href="#">Constructor &amp; Destructor Documentation</a>	26
3.7.1.1	<a href="#">mrmMultiRemoteMouse()</a>	27
3.7.2	<a href="#">Member Function Documentation</a>	27
3.7.2.1	<a href="#">gethostip()</a>	27
3.7.2.2	<a href="#">getscreen()</a>	27
3.7.2.3	<a href="#">recvMouseEvent()</a>	28
3.7.2.4	<a href="#">sendmouseevent()</a>	28
3.7.2.5	<a href="#">sendscreenchange()</a>	28
3.7.2.6	<a href="#">start()</a>	29
3.7.3	<a href="#">Member Data Documentation</a>	29
3.7.3.1	<a href="#">amountscreen</a>	29
3.7.3.2	<a href="#">client</a>	29
3.7.3.3	<a href="#">cursorctrl</a>	29
3.7.3.4	<a href="#">hostip</a>	29
3.7.3.5	<a href="#">micectl</a>	29
3.7.3.6	<a href="#">ocursor</a>	30
3.7.3.7	<a href="#">screens</a>	30
3.7.3.8	<a href="#">server</a>	30
3.7.3.9	<a href="#">tempcount</a>	30
3.8	<a href="#">mrmOSCursor Class Reference</a>	30
3.8.1	<a href="#">Constructor &amp; Destructor Documentation</a>	31
3.8.1.1	<a href="#">mrmOSCursor()</a>	31
3.8.2	<a href="#">Member Function Documentation</a>	31
3.8.2.1	<a href="#">clickPress()</a>	31
3.8.2.2	<a href="#">clickRelease()</a>	31
3.8.2.3	<a href="#">getMutex()</a>	31
3.8.2.4	<a href="#">releaseMutex()</a>	32

3.8.2.5	<a href="#">setCursor()</a>	32
3.8.3	<a href="#">Member Data Documentation</a>	32
3.8.3.1	<a href="#">dpy</a>	32
3.8.3.2	<a href="#">mutex</a>	32
3.8.3.3	<a href="#">posx</a>	32
3.8.3.4	<a href="#">posy</a>	32
3.9	<a href="#">mrmScreen Class Reference</a>	33
3.9.1	<a href="#">Constructor &amp; Destructor Documentation</a>	33
3.9.1.1	<a href="#">mrmScreen()</a>	33
3.9.2	<a href="#">Member Function Documentation</a>	33
3.9.2.1	<a href="#">getDirection()</a>	33
3.9.2.2	<a href="#">getipaddr()</a>	34
3.9.3	<a href="#">Member Data Documentation</a>	34
3.9.3.1	<a href="#">ip</a>	34
3.9.3.2	<a href="#">pos</a>	34
3.10	<a href="#">mrmUDPCClient Class Reference</a>	34
3.10.1	<a href="#">Constructor &amp; Destructor Documentation</a>	34
3.10.1.1	<a href="#">mrmUDPCClient()</a>	34
3.10.2	<a href="#">Member Function Documentation</a>	35
3.10.2.1	<a href="#">sendevent()</a>	35
3.10.3	<a href="#">Member Data Documentation</a>	35
3.10.3.1	<a href="#">addrout</a>	35
3.10.3.2	<a href="#">s</a>	35
3.11	<a href="#">mrmUDPServer Class Reference</a>	35
3.11.1	<a href="#">Constructor &amp; Destructor Documentation</a>	36
3.11.1.1	<a href="#">mrmUDPServer()</a>	36
3.11.2	<a href="#">Member Function Documentation</a>	36
3.11.2.1	<a href="#">run()</a>	36
3.11.2.2	<a href="#">start()</a>	36
3.11.3	<a href="#">Member Data Documentation</a>	37
3.11.3.1	<a href="#">addrin</a>	37
3.11.3.2	<a href="#">hostip</a>	37
3.11.3.3	<a href="#">s</a>	37



<b>4 File Documentation</b>	<b>39</b>
4.1 inc/mrmCfgParser.h File Reference	39
4.1.1 Typedef Documentation	39
4.1.1.1 json	39
4.2 inc/mrmCursor.h File Reference	40
4.3 inc/mrmCursorControl.h File Reference	40
4.3.1 Typedef Documentation	40
4.3.1.1 json	40
4.4 inc/mrmMiceControl.h File Reference	41
4.5 inc/mrmMouse.h File Reference	41
4.6 inc/mrmMultiRemoteMouse.h File Reference	41
4.6.1 Typedef Documentation	42
4.6.1.1 json	42
4.7 inc/mrmOSCursor.h File Reference	42
4.8 inc/mrmScreen.h File Reference	42
4.9 inc/mrmUDPClient.h File Reference	42
4.10 inc/mrmUDPServer.h File Reference	43
4.10.1 Typedef Documentation	43
4.10.1.1 json	43
4.11 inc/osinput.h File Reference	43
4.11.1 Macro Definition Documentation	44
4.11.1.1 ABS_WHEEL	44
4.11.1.2 ABS_X	44
4.11.1.3 ABS_Y	44
4.11.1.4 BTN_LEFT	44
4.11.1.5 BTN_MIDDLE	44
4.11.1.6 BTN_RIGHT	44
4.11.1.7 EV_ABS	44
4.11.1.8 EV_KEY	45
4.11.1.9 EV_REL	45

4.11.1.10 REL_WHEEL . . . . .	45
4.11.1.11 REL_X . . . . .	45
4.11.1.12 REL_Y . . . . .	45
4.12 src/main.cpp File Reference . . . . .	45
4.12.1 Function Documentation . . . . .	46
4.12.1.1 main() . . . . .	46
4.12.2 Variable Documentation . . . . .	46
4.12.2.1 cfgfilename . . . . .	46
4.12.2.2 mrm . . . . .	46
4.13 src/mrmCfgParser.cpp File Reference . . . . .	46
4.14 src/mrmCursor.cpp File Reference . . . . .	46
4.14.1 Function Documentation . . . . .	47
4.14.1.1 ThreadCursor() . . . . .	47
4.15 src/mrmCursorControl.cpp File Reference . . . . .	47
4.15.1 Variable Documentation . . . . .	47
4.15.1.1 mrm . . . . .	47
4.16 src/mrmMiceControl.cpp File Reference . . . . .	47
4.17 src/mrmMouse.cpp File Reference . . . . .	48
4.17.1 Function Documentation . . . . .	48
4.17.1.1 ThreadMouse() . . . . .	48
4.17.2 Variable Documentation . . . . .	48
4.17.2.1 mrm . . . . .	48
4.18 src/mrmMultiRemoteMouse.cpp File Reference . . . . .	48
4.19 src/mrmOSCursor.cpp File Reference . . . . .	49
4.20 src/mrmScreen.cpp File Reference . . . . .	49
4.21 src/mrmUDPClient.cpp File Reference . . . . .	49
4.22 src/mrmUDPServer.cpp File Reference . . . . .	49
4.22.1 Function Documentation . . . . .	49
4.22.1.1 ThreadUDPServer() . . . . .	49
4.22.2 Variable Documentation . . . . .	49
4.22.2.1 mrm . . . . .	49

# Chapter 1

## Class Index

### 1.1 Class List

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

<a href="#">input_event</a>	5
<a href="#">mrmCfgParser</a>	6
<a href="#">mrmCursor</a>	9
<a href="#">mrmCursorControl</a>	20
<a href="#">mrmMiceControl</a>	22
<a href="#">mrmMouse</a>	24
<a href="#">mrmMultiRemoteMouse</a>	26
<a href="#">mrmOSCursor</a>	30
<a href="#">mrmScreen</a>	33
<a href="#">mrmUDPClient</a>	34
<a href="#">mrmUDPServer</a>	35



## Chapter 2

# File Index

### 2.1 File List

Here is a list of all files with brief descriptions:

inc/ <a href="#">mrmCfgParser.h</a>	39
inc/ <a href="#">mrmCursor.h</a>	40
inc/ <a href="#">mrmCursorControl.h</a>	40
inc/ <a href="#">mrmMiceControl.h</a>	41
inc/ <a href="#">mrmMouse.h</a>	41
inc/ <a href="#">mrmMultiRemoteMouse.h</a>	41
inc/ <a href="#">mrmOSCursor.h</a>	42
inc/ <a href="#">mrmScreen.h</a>	42
inc/ <a href="#">mrmUDPClient.h</a>	42
inc/ <a href="#">mrmUDPServer.h</a>	43
inc/ <a href="#">osinput.h</a>	43
src/ <a href="#">main.cpp</a>	45
src/ <a href="#">mrmCfgParser.cpp</a>	46
src/ <a href="#">mrmCursor.cpp</a>	46
src/ <a href="#">mrmCursorControl.cpp</a>	47
src/ <a href="#">mrmMiceControl.cpp</a>	47
src/ <a href="#">mrmMouse.cpp</a>	48
src/ <a href="#">mrmMultiRemoteMouse.cpp</a>	48
src/ <a href="#">mrmOSCursor.cpp</a>	49
src/ <a href="#">mrmScreen.cpp</a>	49
src/ <a href="#">mrmUDPClient.cpp</a>	49
src/ <a href="#">mrmUDPServer.cpp</a>	49



## Chapter 3

# Class Documentation

### 3.1 `input_event` Struct Reference

```
#include <osinput.h>
```

#### Public Attributes

- `timeval` [time](#)
- `uint16_t` [type](#)
- `uint16_t` [code](#)
- `int32_t` [value](#)

#### 3.1.1 Detailed Description

dummy structer for OSX

#### 3.1.2 Member Data Documentation

##### 3.1.2.1 `code`

```
uint16_t input_event::code
```

##### 3.1.2.2 `time`

```
timeval input_event::time
```

### 3.1.2.3 type

```
uint16_t input_event::type
```

### 3.1.2.4 value

```
int32_t input_event::value
```

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

- [inc/osinput.h](#)

## 3.2 mrmCfgParser Class Reference

```
#include <mrmCfgParser.h>
```

### Public Member Functions

- [mrmCfgParser](#) (string jsonstring)
- [json parse](#) ()
- list< [mrmMouse](#) > [createLocalMice](#) (int \*count)
- list< [mrmCursor](#) > [createLocalCursors](#) (int \*count)
- list< [mrmCursor](#) > [createRemoteCursors](#) (int \*count)
- list< [mrmScreen](#) > [createScreens](#) (int \*count)
- int [getUDPPort](#) ()
- string [getHostIP](#) ()

### Public Attributes

- string [jsonstr](#)
- [json](#) jsonobj

### 3.2.1 Constructor & Destructor Documentation

#### 3.2.1.1 mrmCfgParser()

```
mrmCfgParser::mrmCfgParser (  
    string jsonstring )
```

creates a cfg parser. Add the text to be parsed as a parameter



## Parameters

<i>jsonstring</i>	
-------------------	--

## 3.2.2 Member Function Documentation

### 3.2.2.1 createLocalCursors()

```
list< mrmCursor > mrmCfgParser::createLocalCursors (
    int * count )
```

creates a list of all the configured local cursors

## Parameters

<i>count</i>	
--------------	--

## Returns

list of the cursor objects

### 3.2.2.2 createLocalMice()

```
list< mrmMouse > mrmCfgParser::createLocalMice (
    int * count )
```

creates a list of all the configured local mice

## Parameters

<i>count</i>	
--------------	--

## Returns

list of the mice objects

### 3.2.2.3 createRemoteCursors()

```
list< mrmCursor > mrmCfgParser::createRemoteCursors (
    int * count )
```

creates a list of all the configured remote cursors

**Parameters**

<i>count</i>	
--------------	--

**Returns**

list of the cursor objects

**3.2.2.4 createScreens()**

```
list< mrmScreen > mrmCfgParser::createScreens (
    int * count )
```

creates a list of all the configured remote screens

**Parameters**

<i>count</i>	
--------------	--

**Returns**

list of the screen objects

**3.2.2.5 getHostIP()**

```
string mrmCfgParser::getHostIP ( )
```

return the configured host ip as a string

**Returns****3.2.2.6 getUDPPort()**

```
int mrmCfgParser::getUDPPort ( )
```

return the configured UDP port

**Returns**

## 3.2.2.7 parse()

```
json mrmCfgParser::parse ( )
```

parse the received string to an json object

Returns

## 3.2.3 Member Data Documentation

## 3.2.3.1 jsonobj

```
json mrmCfgParser::jsonobj
```

## 3.2.3.2 jsonstr

```
string mrmCfgParser::jsonstr
```

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

- inc/[mrmCfgParser.h](#)
- src/[mrmCfgParser.cpp](#)

## 3.3 mrmCursor Class Reference

```
#include <mrmCursor.h>
```

## Public Member Functions

- int [getPosx](#) () const
- void [setPosx](#) (int [posx](#))
- int [getPosy](#) () const
- void [setPosy](#) (int [posy](#))
- int [getXMax](#) ()
- int [getYMax](#) ()
- Display \* [getDisplay](#) ()
- [mrmCursor](#) (string [id](#), int [color\\_r](#), int [color\\_g](#), int [color\\_b](#), int [show](#), int [click](#), int [datatf](#), bool [active](#), int devicenumbers[])
- string [getid](#) ()
- void [run](#) ([mrmCursor](#) \*cursor)
- void [start](#) ()
- int [updateTheMessageQueue](#) ()
- void [createTheWindow](#) ()
- void [createTheRenderContext](#) ()
- void [redrawTheWindow](#) ()
- void [describe\\_fbconfig](#) (GLXFBConfig [fbconfig](#))
- void [setlastabsx](#) (int lastabs)
- void [setlastabsy](#) (int lastabs)
- int [getrelxfromabs](#) ()
- int [getrelyfromabs](#) ()
- bool [isactive](#) ()
- void [activate](#) (bool activate)

## Public Attributes

- int [clickrunning](#)
- string [id](#)
- int [color\\_r](#)
- int [color\\_g](#)
- int [color\\_b](#)
- int [show](#)
- int [click](#)
- int [datatf](#)
- int [lastabsx](#)
- int [lastabsy](#)
- int [relx](#)
- int [rely](#)

## Private Attributes

- int [Xscreen](#)
- Atom [del\\_atom](#)
- Colormap [cmap](#)
- Display \* [Xdisplay](#)
- XVisualInfo \* [visual](#)
- XRenderPictFormat \* [pict\\_format](#)
- GLXFBConfig \* [fbconfigs](#)
- GLXFBConfig [fbconfig](#)
- int [numfbconfigs](#)
- GLXContext [render\\_context](#)
- Window [Xroot](#)
- Window [window\\_handle](#)
- GLXWindow [glX\\_window\\_handle](#)
- int [width](#)
- int [height](#)
- int [posx](#)
- int [posy](#)
- bool [active](#)

## Static Private Attributes

- static int [VisData](#) []

### 3.3.1 Constructor & Destructor Documentation

#### 3.3.1.1 mrmCursor()

```
mrmCursor::mrmCursor (
    string id,
    int color_r,
    int color_g,
    int color_b,
    int show,
    int click,
    int datatf,
    bool active,
    int devicenumbers[] )
```

create the current cursor with al the needed configuration values

**Parameters**

<i>id</i>	
<i>color_r</i>	
<i>color_g</i>	
<i>color_b</i>	
<i>show</i>	
<i>click</i>	
<i>datatf</i>	
<i>active</i>	
<i>devicenumbers</i>	

**3.3.2 Member Function Documentation****3.3.2.1 activate()**

```
void mrmCursor::activate (
    bool activate )
```

activate or deactivate the cursor. The parameter defines if deactivated or activated

**Parameters**

<i>activate</i>	
-----------------	--

**3.3.2.2 createTheRenderContext()**

```
void mrmCursor::createTheRenderContext ( )
```

creating the needed render content to be able to draw the window and the cursor

**3.3.2.3 createTheWindow()**

```
void mrmCursor::createTheWindow ( )
```

create the window needed to draw in it

**3.3.2.4 describe\_fbconfig()**

```
void mrmCursor::describe_fbconfig (
    GLXFBConfig fbconfig )
```

set/describe the current used fbconfig

**Returns**

#### 3.3.2.5 getDisplay()

```
Display * mrmCursor::getDisplay ( )
```

#### 3.3.2.6 getid()

```
string mrmCursor::getid ( )
```

get the id of this cursor as string

**Returns**

#### 3.3.2.7 getPosx()

```
int mrmCursor::getPosx ( ) const
```

get position

**Returns**

#### 3.3.2.8 getPosy()

```
int mrmCursor::getPosy ( ) const
```

get position

**Returns**

#### 3.3.2.9 getrelxfromabs()

```
int mrmCursor::getrelxfromabs ( )
```

calculate the relative value from the last two absolut values

**Returns**

rel value

#### 3.3.2.10 getrelyfromabs()

```
int mrmCursor::getrelyfromabs ( )
```

calculate the relative value from the last two absolut values

##### Returns

rel value

#### 3.3.2.11 getXMax()

```
int mrmCursor::getXMax ( )
```

get max position

##### Returns

#### 3.3.2.12 getYMax()

```
int mrmCursor::getYMax ( )
```

get max position

##### Returns

#### 3.3.2.13 isactive()

```
bool mrmCursor::isactive ( )
```

returns the current activ state

##### Returns



#### 3.3.2.14 redrawTheWindow()

```
void mrmCursor::redrawTheWindow ( )
```

redraw the window which holds our cursor. Redraw the cursor into this window.

#### 3.3.2.15 run()

```
void mrmCursor::run (
    mrmCursor * cursor )
```

#### 3.3.2.16 setlastabsx()

```
void mrmCursor::setlastabsx (
    int lastabs )
```

set the last absolut value

##### Parameters

<i>lastabs</i>	
----------------	--

#### 3.3.2.17 setlastabsy()

```
void mrmCursor::setlastabsy (
    int lastabs )
```

set the last absolut value

##### Parameters

<i>lastabs</i>	
----------------	--

#### 3.3.2.18 setPosx()

```
void mrmCursor::setPosx (
    int posx )
```

set position

##### Returns

#### 3.3.2.19 setPosy()

```
void mrmCursor::setPosy (
    int posy )
```

set position

**Returns**

#### 3.3.2.20 start()

```
void mrmCursor::start ( )
```

starts the Thread which is running the cursor. The run function will be called in the thread.

#### 3.3.2.21 updateTheMessageQueue()

```
int mrmCursor::updateTheMessageQueue ( )
```

Check the MMessage queue of the windowserver. Act if needed

**Returns**

### 3.3.3 Member Data Documentation

#### 3.3.3.1 active

```
bool mrmCursor::active [private]
```

#### 3.3.3.2 click

```
int mrmCursor::click
```

#### 3.3.3.3 clickrunning

```
int mrmCursor::clickrunning
```

#### 3.3.3.4 cmap

```
Colormap mrmCursor::cmap [private]
```

#### 3.3.3.5 color\_b

```
int mrmCursor::color_b
```

#### 3.3.3.6 color\_g

```
int mrmCursor::color_g
```

#### 3.3.3.7 color\_r

```
int mrmCursor::color_r
```

#### 3.3.3.8 datatf

```
int mrmCursor::datatf
```

#### 3.3.3.9 del\_atom

```
Atom mrmCursor::del_atom [private]
```

#### 3.3.3.10 fbconfig

```
GLXFBConfig mrmCursor::fbconfig [private]
```

**3.3.3.11 fbconfigs**

```
GLXFBConfig* mrmCursor::fbconfigs [private]
```

**3.3.3.12 glX\_window\_handle**

```
GLXWindow mrmCursor::glX_window_handle [private]
```

**3.3.3.13 height**

```
int mrmCursor::height [private]
```

**3.3.3.14 id**

```
string mrmCursor::id
```

**3.3.3.15 lastabsx**

```
int mrmCursor::lastabsx
```

**3.3.3.16 lastabsy**

```
int mrmCursor::lastabsy
```

**3.3.3.17 numfbconfigs**

```
int mrmCursor::numfbconfigs [private]
```

**3.3.3.18 pict\_format**

```
XRenderPictFormat* mrmCursor::pict_format [private]
```

**3.3.3.19 posx**

```
int mrmCursor::posx [private]
```

**3.3.3.20 posy**

```
int mrmCursor::posy [private]
```

**3.3.3.21 relx**

```
int mrmCursor::relx
```

**3.3.3.22 rely**

```
int mrmCursor::rely
```

**3.3.3.23 render\_context**

```
GLXContext mrmCursor::render_context [private]
```

**3.3.3.24 show**

```
int mrmCursor::show
```

**3.3.3.25 VisData**

```
int mrmCursor::VisData [static], [private]
```

**Initial value:**

```
= {  
    GLX_RENDER_TYPE, GLX_RGBA_BIT,  
    GLX_DRAWABLE_TYPE, GLX_WINDOW_BIT,  
    GLX_DOUBLEBUFFER, True,  
    GLX_RED_SIZE, 8,  
    GLX_GREEN_SIZE, 8,  
    GLX_BLUE_SIZE, 8,  
    GLX_ALPHA_SIZE, 8,  
    GLX_DEPTH_SIZE, 16,  
    None  
}
```

### 3.3.3.26 visual

```
XVisualInfo* mrmCursor::visual [private]
```

### 3.3.3.27 width

```
int mrmCursor::width [private]
```

### 3.3.3.28 window\_handle

```
Window mrmCursor::window_handle [private]
```

### 3.3.3.29 Xdisplay

```
Display* mrmCursor::Xdisplay [private]
```

### 3.3.3.30 Xroot

```
Window mrmCursor::Xroot [private]
```

### 3.3.3.31 Xscreen

```
int mrmCursor::Xscreen [private]
```

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

- [inc/mrmCursor.h](#)
- [src/mrmCursor.cpp](#)

## 3.4 mrmCursorControl Class Reference

```
#include <mrmCursorControl.h>
```

## Public Member Functions

- `mrmCursorControl` (list< `mrmCursor` > localcursors, list< `mrmCursor` > remotecursors, int amountlc, int amountrc)
- void `start` (`mrmOSCursor` \*)
- list< `mrmCursor` > `getlist` ()
- int `work_event` (`json` work\_event)

## Public Attributes

- list< `mrmCursor` > `cursors`
- int `amount`
- `mrmOSCursor` \* `oscursor`
- unsigned long `amountlocal`

## 3.4.1 Constructor & Destructor Documentation

### 3.4.1.1 mrmCursorControl()

```
mrmCursorControl::mrmCursorControl (
    list< mrmCursor > localcursors,
    list< mrmCursor > remotecursors,
    int amountlc,
    int amountrc )
```

## 3.4.2 Member Function Documentation

### 3.4.2.1 getlist()

```
list<mrmCursor> mrmCursorControl::getlist ( )
```

### 3.4.2.2 start()

```
void mrmCursorControl::start (
    mrmOSCursor * oscursor )
```

starts all the Thread of the known mice

### 3.4.2.3 work\_event()

```
int mrmCursorControl::work_event (
    json work_event )
```

receives an event to work with. The event will be processed and given to the correct object

**Parameters**

<i>work_event</i>	
-------------------	--

**Returns****3.4.3 Member Data Documentation****3.4.3.1 amount**

```
int mrmCursorControl::amount
```

**3.4.3.2 amountlocal**

```
unsigned long mrmCursorControl::amountlocal
```

**3.4.3.3 cursors**

```
list<mrmCursor> mrmCursorControl::cursors
```

**3.4.3.4 oscursor**

```
mrmOSCursor* mrmCursorControl::oscursor
```

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

- [inc/mrmCursorControl.h](#)
- [src/mrmCursorControl.cpp](#)

**3.5 mrmMiceControl Class Reference**

```
#include <mrmMiceControl.h>
```



## Public Member Functions

- `mrmMiceControl` (list< `mrmMouse` > localmice, int `amount`)
- void `start` ()

## Public Attributes

- list< `mrmMouse` > `mice`
- int `amount`

## 3.5.1 Constructor & Destructor Documentation

### 3.5.1.1 `mrmMiceControl()`

```
mrmMiceControl::mrmMiceControl (
    list< mrmMouse > localmice,
    int amount )
```

## 3.5.2 Member Function Documentation

### 3.5.2.1 `start()`

```
void mrmMiceControl::start ( )
```

starts all the Thread of the known mice

## 3.5.3 Member Data Documentation

### 3.5.3.1 `amount`

```
int mrmMiceControl::amount
```

### 3.5.3.2 mice

```
list<mrmMouse> mrmMiceControl::mice
```

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

- inc/mrmMiceControl.h
- src/mrmMiceControl.cpp

## 3.6 mrmMouse Class Reference

```
#include <mrmMouse.h>
```

### Public Member Functions

- [mrmMouse](#) (string [id](#), string evenfile)
- void [start](#) ()
- void [run](#) ([mrmMouse](#) \*mouse)

### Public Attributes

- struct [input\\_event](#) \* [ieMouse](#)
- int [mousefile](#)
- const char \* [id](#)
- const char \* [eventfile](#)

### 3.6.1 Constructor & Destructor Documentation

#### 3.6.1.1 mrmMouse()

```
mrmMouse::mrmMouse (
    string id,
    string evenfile )
```

create the mouse object. It will know which Devicefile to read with the eventfile string parameter.

#### Parameters

<i>id</i>	
<i>evenfile</i>	

## 3.6.2 Member Function Documentation

### 3.6.2.1 run()

```
void mrmMouse::run (
    mrmMouse * mouse )
```

the run function. This function will be called from the mouse thread. The run function receives Mouseevent packages and send them to our controller

#### Parameters

<i>mouse</i>	
--------------	--

### 3.6.2.2 start()

```
void mrmMouse::start ( )
```

starts the Thread which is running the mousobject. The run function will be called in the thread.

## 3.6.3 Member Data Documentation

### 3.6.3.1 eventfile

```
const char* mrmMouse::eventfile
```

### 3.6.3.2 id

```
const char* mrmMouse::id
```

### 3.6.3.3 ieMouse

```
struct input_event* mrmMouse::ieMouse
```

### 3.6.3.4 mousefile

```
int mrmMouse::mousefile
```

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

- [inc/mrmMouse.h](#)
- [src/mrmMouse.cpp](#)

## 3.7 mrmMultiRemoteMouse Class Reference

```
#include <mrmMultiRemoteMouse.h>
```

### Public Member Functions

- [mrmMultiRemoteMouse](#) (list< [mrmMouse](#) > localmice, int amountlm, list< [mrmCursor](#) > localcursors, int amountlc, list< [mrmCursor](#) > remotecursors, int amountrc, list< [mrmScreen](#) > remotescreens, int amountrs, [mrmUDPClient](#) \*udpclient, [mrmUDPServer](#) \*udpserver)
- void [start](#) ()
- void [recvMouseEvent](#) ([json](#) mouseEvent)
- void [sendscreenchange](#) ([json](#) scevent)
- void [sendmouseevent](#) ([json](#) mevent)
- string [getscreen](#) (int i)
- string [gethostip](#) ()

### Public Attributes

- list< [mrmScreen](#) > [screens](#)
- [mrmMiceControl](#) \* [micectl](#)
- [mrmCursorControl](#) \* [cursorctrl](#)
- [mrmOSCursr](#) \* [oscursor](#)
- int [amountscreen](#)
- int [tempcount](#)
- string [hostip](#)

### Private Attributes

- [mrmUDPClient](#) \* [client](#)
- [mrmUDPServer](#) \* [server](#)

### 3.7.1 Constructor & Destructor Documentation

### 3.7.1.1 mrmMultiRemoteMouse()

```
mrmMultiRemoteMouse::mrmMultiRemoteMouse (
    list< mrmMouse > localmice,
    int amountlm,
    list< mrmCursor > localcursors,
    int amountlc,
    list< mrmCursor > remotecursors,
    int amountrc,
    list< mrmScreen > remotescreens,
    int amountrs,
    mrmUDPClient * udpclient,
    mrmUDPServer * udpserver )
```

creates the multimouse controller. The multimouse controller needs to know all the objects. All known objects are given with the constructor

#### Parameters

<i>localmice</i>	
<i>amountlm</i>	
<i>localcursors</i>	
<i>amountlc</i>	
<i>remotecursors</i>	
<i>amountrc</i>	
<i>remotescreens</i>	
<i>amountrs</i>	
<i>udpclient</i>	
<i>udpserver</i>	

## 3.7.2 Member Function Documentation

### 3.7.2.1 gethostip()

```
string mrmMultiRemoteMouse::gethostip ( )
```

this function returns the host IP as string value

#### Returns

### 3.7.2.2 getscreen()

```
string mrmMultiRemoteMouse::getscreen (
    int i )
```

Sometimes the cursor needs to get the next screen in a specific direction. The cursor will call this function. The controller knows the screen objects and will answer with an IP as a string

**Parameters**

<i>i</i>	
----------	--

**Returns**

screen IP as string value

**3.7.2.3 recvMouseEvent()**

```
void mrmMultiRemoteMouse::recvMouseEvent (
    json mouseEvent )
```

If a mouse or a server receives a mouseevent, it will call this function. This function will send the event to the correct controller and work the event.

**Parameters**

<i>mouseEvent</i>	
-------------------	--

**3.7.2.4 sendmouseevent()**

```
void mrmMultiRemoteMouse::sendmouseevent (
    json mevent )
```

if a cursor needs to send a mouseevent it will call this function. The controller knows the client and will forward the request. The request is given with the parameter json object

**Parameters**

<i>mevent</i>	
---------------	--

**3.7.2.5 sendscreenchange()**

```
void mrmMultiRemoteMouse::sendscreenchange (
    json scevent )
```

if a cursor needs to send a screenchange it will call this function. The controller knows the client and will forward the request. The request is given with the parameter json object

## Parameters

<i>scevent</i>	
----------------	--

### 3.7.2.6 start()

```
void mrmMultiRemoteMouse::start ( )
```

starting all threads of the integrated objects

## 3.7.3 Member Data Documentation

### 3.7.3.1 amountscreen

```
int mrmMultiRemoteMouse::amountscreen
```

### 3.7.3.2 client

```
mrmUDPClient* mrmMultiRemoteMouse::client [private]
```

### 3.7.3.3 cursorctrl

```
mrmCursorControl* mrmMultiRemoteMouse::cursorctrl
```

### 3.7.3.4 hostip

```
string mrmMultiRemoteMouse::hostip
```

### 3.7.3.5 micctrl

```
mrmMiceControl* mrmMultiRemoteMouse::micctrl
```

### 3.7.3.6 oscursor

```
mrmOSCursOr* mrmMultiRemoteMouse::oscursor
```

### 3.7.3.7 screens

```
list<mrmScreen> mrmMultiRemoteMouse::screens
```

### 3.7.3.8 server

```
mrmUDPServer* mrmMultiRemoteMouse::server [private]
```

### 3.7.3.9 tempcount

```
int mrmMultiRemoteMouse::tempcount
```

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

- inc/mrmMultiRemoteMouse.h
- src/mrmMultiRemoteMouse.cpp

## 3.8 mrmOSCursOr Class Reference

```
#include <mrmOSCursOr.h>
```

### Public Member Functions

- [mrmOSCursOr](#) ()
- void [getMutex](#) ()
- void [releaseMutex](#) ()
- void [setCursor](#) (int [posx](#), int [posy](#))
- void [clickPress](#) (int key)
- void [clickRelease](#) (int key)

### Public Attributes

- int [posx](#)
- int [posy](#)
- Display \* [dpy](#)
- pthread\_mutex\_t [mutex](#)



### 3.8.1 Constructor & Destructor Documentation

#### 3.8.1.1 mrmOSCursor()

```
mrmOSCursor::mrmOSCursor ( )
```

create the OS cursor object

### 3.8.2 Member Function Documentation

#### 3.8.2.1 clickPress()

```
void mrmOSCursor::clickPress (
    int key )
```

simulates a button click press on the os cursor. The param gives the used key for the click

##### Parameters

<i>key</i>	
------------	--

#### 3.8.2.2 clickRelease()

```
void mrmOSCursor::clickRelease (
    int key )
```

simulates a button click release on the os cursor. The param gives the used key for the click

##### Parameters

<i>key</i>	
------------	--

#### 3.8.2.3 getMutex()

```
void mrmOSCursor::getMutex ( )
```

get the mutex. This function will make a blocking wait until you have acces to the mutex

### 3.8.2.4 releaseMutex()

```
void mrmOSCursOr::releaseMutex ( )
```

release the mutex. This function frees the mutex if it is in your control

### 3.8.2.5 setCursor()

```
void mrmOSCursOr::setCursor (
    int posx,
    int posy )
```

sets the cursor position to the parameter values for *posx* and *y*.

#### Parameters

<i>posx</i>	
<i>posy</i>	

## 3.8.3 Member Data Documentation

### 3.8.3.1 dpy

```
Display* mrmOSCursOr::dpy
```

### 3.8.3.2 mutex

```
pthread_mutex_t mrmOSCursOr::mutex
```

### 3.8.3.3 posx

```
int mrmOSCursOr::posx
```

### 3.8.3.4 posy

```
int mrmOSCursOr::posy
```

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

- [inc/mrmOSCursOr.h](#)
- [src/mrmOSCursOr.cpp](#)

## 3.9 mrmScreen Class Reference

```
#include <mrmScreen.h>
```

### Public Member Functions

- [mrmScreen](#) (string [ip](#), int [pos](#))
- int [getDirection](#) ()
- string [getipaddr](#) ()

### Public Attributes

- string [ip](#)
- int [pos](#)

### 3.9.1 Constructor & Destructor Documentation

#### 3.9.1.1 mrmScreen()

```
mrmScreen::mrmScreen (  
    string ip,  
    int pos )
```

creates the screen with all needed information, like position and IP

#### Parameters

<i>ip</i>	
<i>pos</i>	

### 3.9.2 Member Function Documentation

#### 3.9.2.1 getDirection()

```
int mrmScreen::getDirection ( )
```

getter function to receive the Direction of the screen

#### Returns

the direction as integer -1 and 1

### 3.9.2.2 getipaddr()

```
string mrmScreen::getipaddr ( )
```

getter function to receive the IP of the screen

#### Returns

returns the IP as string value

## 3.9.3 Member Data Documentation

### 3.9.3.1 ip

```
string mrmScreen::ip
```

### 3.9.3.2 pos

```
int mrmScreen::pos
```

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

- [inc/mrmScreen.h](#)
- [src/mrmScreen.cpp](#)

## 3.10 mrmUDPClient Class Reference

```
#include <mrmUDPClient.h>
```

### Public Member Functions

- [mrmUDPClient](#) (int port)
- void [sendevent](#) ([json](#) sendevent)

### Private Attributes

- int [s](#)
- struct sockaddr\_in [addrout](#)

### 3.10.1 Constructor & Destructor Documentation

#### 3.10.1.1 mrmUDPClient()

```
mrmUDPClient::mrmUDPClient (
    int port )
```

create the UDP client on the parameter port

## Parameters

<i>port</i>	
-------------	--

## 3.10.2 Member Function Documentation

## 3.10.2.1 sendevent()

```
void mrmUDPClient::sendevent (
    json sendevent )
```

send a event over the network. The event must be given as a json object

## Parameters

<i>sendevent</i>	
------------------	--

## 3.10.3 Member Data Documentation

## 3.10.3.1 addrout

```
struct sockaddr_in mrmUDPClient::addrout [private]
```

## 3.10.3.2 s

```
int mrmUDPClient::s [private]
```

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

- inc/[mrmUDPClient.h](#)
- src/[mrmUDPClient.cpp](#)

## 3.11 mrmUDPServer Class Reference

```
#include <mrmUDPServer.h>
```

## Public Member Functions

- [mrmUDPServer](#) (int port, string [hostip](#))
- void [start](#) ()
- void [run](#) ()

## Public Attributes

- int [s](#)
- struct sockaddr\_in [addrin](#)
- string [hostip](#)

### 3.11.1 Constructor & Destructor Documentation

#### 3.11.1.1 mrmUDPServer()

```
mrmUDPServer::mrmUDPServer (
    int port,
    string hostip )
```

creating the USP server and will be opened with the parameter port. The hostip is needed to chekc which messages are import for our server

#### Parameters

<i>port</i>	
<i>hostip</i>	

### 3.11.2 Member Function Documentation

#### 3.11.2.1 run()

```
void mrmUDPServer::run ( )
```

the run function. This function will bi called from the server thread. The run function receives UDP packages and send them to our controller

#### 3.11.2.2 start()

```
void mrmUDPServer::start ( )
```

starts the Thread which is running the server. The run function wil be called in the thread.

### 3.11.3 Member Data Documentation

#### 3.11.3.1 addrin

```
struct sockaddr_in mrmUDPServer::addrin
```

#### 3.11.3.2 hostip

```
string mrmUDPServer::hostip
```

#### 3.11.3.3 s

```
int mrmUDPServer::s
```

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

- [inc/mrmUDPServer.h](#)
- [src/mrmUDPServer.cpp](#)





## Chapter 4

# File Documentation

### 4.1 inc/mrmCfgParser.h File Reference

```
#include <string>
#include <list>
#include "mrmMouse.h"
#include "mrmCursor.h"
#include "mrmScreen.h"
#include <nlohmann/json.hpp>
```

#### Classes

- class [mrmCfgParser](#)

#### Typedefs

- using [json](#) = nlohmann::json

#### 4.1.1 Typedef Documentation

##### 4.1.1.1 json

```
using json = nlohmann::json
```

## 4.2 inc/mrmCursor.h File Reference

```
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <sys/types.h>
#include <GL/gl.h>
#include <GL/glx.h>
#include <GL/glxext.h>
#include <X11/Xatom.h>
#include <X11/extensions/Xrender.h>
#include <X11/Xutil.h>
#include <string>
```

### Classes

- class [mrmCursor](#)

## 4.3 inc/mrmCursorControl.h File Reference

```
#include <list>
#include "mrmCursor.h"
#include "mrmOSCursor.h"
#include <nlohmann/json.hpp>
```

### Classes

- class [mrmCursorControl](#)

### Typedefs

- using [json](#) = nlohmann::json

### 4.3.1 Typedef Documentation

#### 4.3.1.1 json

```
using json = nlohmann::json
```

## 4.4 inc/mrmMiceControl.h File Reference

```
#include <list>
#include "../inc/mrmMouse.h"
```

### Classes

- class [mrmMiceControl](#)

## 4.5 inc/mrmMouse.h File Reference

```
#include <string>
#include <unistd.h>
#include <iostream>
#include "../inc/osinput.h"
#include <fcntl.h>
```

### Classes

- class [mrmMouse](#)

## 4.6 inc/mrmMultiRemoteMouse.h File Reference

```
#include <list>
#include "mrmUDPClient.h"
#include "mrmMouse.h"
#include "mrmCursor.h"
#include "mrmScreen.h"
#include "mrmUDPServer.h"
#include "mrmMiceControl.h"
#include "mrmCursorControl.h"
#include "mrmOSCursors.h"
#include <nlohmann/json.hpp>
```

### Classes

- class [mrmMultiRemoteMouse](#)

### Typedefs

- using [json](#) = nlohmann::json

### 4.6.1 Typedef Documentation

#### 4.6.1.1 json

```
using json = nlohmann::json
```

## 4.7 inc/mrmOSCursor.h File Reference

```
#include <X11/Xatom.h>
#include <X11/extensions/Xrender.h>
#include <X11/extensions/Xtest.h>
#include <X11/Xutil.h>
```

### Classes

- class [mrmOSCursor](#)

## 4.8 inc/mrmScreen.h File Reference

```
#include <string>
```

### Classes

- class [mrmScreen](#)

## 4.9 inc/mrmUDPClient.h File Reference

```
#include <netinet/in.h>
#include "../inc/mrmCfgParser.h"
```

### Classes

- class [mrmUDPClient](#)

## 4.10 inc/mrmUDPServer.h File Reference

```
#include <netinet/in.h>
#include <nlohmann/json.hpp>
```

### Classes

- class [mrmUDPServer](#)

### Typedefs

- using [json](#) = nlohmann::json

#### 4.10.1 Typedef Documentation

##### 4.10.1.1 json

```
using json = nlohmann::json
```

## 4.11 inc/osinput.h File Reference

### Classes

- struct [input\\_event](#)

### Macros

- #define [EV\\_KEY](#) 0x01
- #define [EV\\_REL](#) 0x02
- #define [EV\\_ABS](#) 0x03
- #define [REL\\_X](#) 0x00
- #define [REL\\_Y](#) 0x01
- #define [REL\\_WHEEL](#) 0x08
- #define [ABS\\_X](#) 0x00
- #define [ABS\\_Y](#) 0x01
- #define [ABS\\_WHEEL](#) 0x08
- #define [BTN\\_LEFT](#) 0x110
- #define [BTN\\_RIGHT](#) 0x111
- #define [BTN\\_MIDDLE](#) 0x112

### 4.11.1 Macro Definition Documentation

#### 4.11.1.1 ABS\_WHEEL

```
#define ABS_WHEEL 0x08
```

#### 4.11.1.2 ABS\_X

```
#define ABS_X 0x00
```

#### 4.11.1.3 ABS\_Y

```
#define ABS_Y 0x01
```

#### 4.11.1.4 BTN\_LEFT

```
#define BTN_LEFT 0x110
```

#### 4.11.1.5 BTN\_MIDDLE

```
#define BTN_MIDDLE 0x112
```

#### 4.11.1.6 BTN\_RIGHT

```
#define BTN_RIGHT 0x111
```

#### 4.11.1.7 EV\_ABS

```
#define EV_ABS 0x03
```

#### 4.11.1.8 EV\_KEY

```
#define EV_KEY 0x01
```

#### 4.11.1.9 EV\_REL

```
#define EV_REL 0x02
```

#### 4.11.1.10 REL\_WHEEL

```
#define REL_WHEEL 0x08
```

#### 4.11.1.11 REL\_X

```
#define REL_X 0x00
```

#### 4.11.1.12 REL\_Y

```
#define REL_Y 0x01
```

## 4.12 src/main.cpp File Reference

```
#include <iostream>
#include <fstream>
#include <sstream>
#include "../inc/mrmCfgParser.h"
#include "../inc/mrmMultiRemoteMouse.h"
```

### Functions

- int [main](#) (int argc, char \*\*argv)

### Variables

- [mrmMultiRemoteMouse](#) \* [mrm](#) = nullptr
- string [cfgfilename](#) = "../cfg/config.json"

## 4.12.1 Function Documentation

### 4.12.1.1 main()

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

## 4.12.2 Variable Documentation

### 4.12.2.1 cfgfilename

```
string cfgfilename = "../cfg/config.json"
```

### 4.12.2.2 mrm

```
mrmMultiRemoteMouse* mrm = nullptr
```

## 4.13 src/mrmCfgParser.cpp File Reference

```
#include <nlohmann/json.hpp>
#include <iostream>
#include <list>
#include "../inc/mrmCfgParser.h"
```

## 4.14 src/mrmCursor.cpp File Reference

```
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <GL/glew.h>
#include <GL/glx.h>
#include <X11/Xatom.h>
#include <X11/extensions/shape.h>
#include <X11/extensions/Xrender.h>
#include <X11/extensions/Xfixes.h>
#include <X11/extensions/Xcomposite.h>
#include <iostream>
#include <sstream>
#include "../inc/mrmCursor.h"
```



## Functions

- void \* [ThreadCursor](#) (void \*pVoid)

### 4.14.1 Function Documentation

#### 4.14.1.1 ThreadCursor()

```
void* ThreadCursor (  
    void * pVoid )
```

## 4.15 src/mrmCursorControl.cpp File Reference

```
#include <iostream>  
#include "../inc/osinput.h"  
#include "../inc/mrmOSCursor.h"  
#include "../inc/mrmCursorControl.h"  
#include "../inc/mrmMultiRemoteMouse.h"
```

## Variables

- [mrmMultiRemoteMouse](#) \* mrm

### 4.15.1 Variable Documentation

#### 4.15.1.1 mrm

```
mrmMultiRemoteMouse* mrm
```

## 4.16 src/mrmMiceControl.cpp File Reference

```
#include <utility>  
#include "../inc/mrmMiceControl.h"
```

## 4.17 src/mrmMouse.cpp File Reference

```
#include <unistd.h>
#include <iostream>
#include "../inc/osinput.h"
#include <fcntl.h>
#include <cstring>
#include "../inc/mrmMouse.h"
#include "../inc/mrmMultiRemoteMouse.h"
```

### Functions

- void \* [ThreadMouse](#) (void \*pVoid)

### Variables

- [mrmMultiRemoteMouse](#) \* [mrm](#)

### 4.17.1 Function Documentation

#### 4.17.1.1 ThreadMouse()

```
void* ThreadMouse (
    void * pVoid )
```

### 4.17.2 Variable Documentation

#### 4.17.2.1 mrm

```
mrmMultiRemoteMouse* mrm
```

## 4.18 src/mrmMultiRemoteMouse.cpp File Reference

```
#include "../inc/mrmMultiRemoteMouse.h"
```

## 4.19 src/mrmOSCursor.cpp File Reference

```
#include <cstring>
#include <iostream>
#include "../inc/mrmOSCursor.h"
```

## 4.20 src/mrmScreen.cpp File Reference

```
#include "../inc/mrmScreen.h"
```

## 4.21 src/mrmUDPClient.cpp File Reference

```
#include <iostream>
#include <cstring>
#include "../inc/mrmUDPClient.h"
```

## 4.22 src/mrmUDPServer.cpp File Reference

```
#include <iostream>
#include <arpa/inet.h>
#include "../inc/mrmUDPServer.h"
#include "../inc/mrmMultiRemoteMouse.h"
```

### Functions

- void \* [ThreadUDPServer](#) (void \*pVoid)

### Variables

- [mrmMultiRemoteMouse](#) \* [mrm](#)

### 4.22.1 Function Documentation

#### 4.22.1.1 ThreadUDPServer()

```
void* ThreadUDPServer (
    void * pVoid )
```

### 4.22.2 Variable Documentation

#### 4.22.2.1 mrm

```
mrmMultiRemoteMouse* mrm
```



# Index

ABS\_WHEEL  
    osinput.h, 44  
ABS\_X  
    osinput.h, 44  
ABS\_Y  
    osinput.h, 44  
activate  
    mrmCursor, 12  
active  
    mrmCursor, 16  
addrin  
    mrmUDPServer, 37  
addrout  
    mrmUDPClient, 35  
amount  
    mrmCursorControl, 22  
    mrmMiceControl, 23  
amountlocal  
    mrmCursorControl, 22  
amountscreen  
    mrmMultiRemoteMouse, 29  
  
BTN\_LEFT  
    osinput.h, 44  
BTN\_MIDDLE  
    osinput.h, 44  
BTN\_RIGHT  
    osinput.h, 44  
  
cfgfilename  
    main.cpp, 46  
click  
    mrmCursor, 16  
clickPress  
    mrmOSCursor, 31  
clickRelease  
    mrmOSCursor, 31  
clickrunning  
    mrmCursor, 16  
client  
    mrmMultiRemoteMouse, 29  
cmap  
    mrmCursor, 17  
code  
    input\_event, 5  
color\_b  
    mrmCursor, 17  
color\_g  
    mrmCursor, 17  
color\_r

    mrmCursor, 17  
createLocalCursors  
    mrmCfgParser, 7  
createLocalMice  
    mrmCfgParser, 7  
createRemoteCursors  
    mrmCfgParser, 7  
createScreens  
    mrmCfgParser, 8  
createTheRenderContext  
    mrmCursor, 12  
createTheWindow  
    mrmCursor, 12  
cursorctrl  
    mrmMultiRemoteMouse, 29  
cursors  
    mrmCursorControl, 22  
  
datatf  
    mrmCursor, 17  
del\_atom  
    mrmCursor, 17  
describe\_fbconfig  
    mrmCursor, 12  
dpy  
    mrmOSCursor, 32  
  
EV\_ABS  
    osinput.h, 44  
EV\_KEY  
    osinput.h, 44  
EV\_REL  
    osinput.h, 45  
eventfile  
    mrmMouse, 25  
  
fbconfig  
    mrmCursor, 17  
fbconfigs  
    mrmCursor, 17  
  
getDirection  
    mrmScreen, 33  
getDisplay  
    mrmCursor, 12  
getHostIP  
    mrmCfgParser, 8  
getMutex  
    mrmOSCursor, 31  
getPosx

- mrmCursor, 13
- getPosy
  - mrmCursor, 13
- getUDPPort
  - mrmCfgParser, 8
- getXMax
  - mrmCursor, 14
- getYMax
  - mrmCursor, 14
- gethostip
  - mrmMultiRemoteMouse, 27
- getid
  - mrmCursor, 13
- getipaddr
  - mrmScreen, 33
- getlist
  - mrmCursorControl, 21
- getrelxfromabs
  - mrmCursor, 13
- getrelyfromabs
  - mrmCursor, 13
- getscreen
  - mrmMultiRemoteMouse, 27
- glX\_window\_handle
  - mrmCursor, 18
- height
  - mrmCursor, 18
- hostip
  - mrmMultiRemoteMouse, 29
  - mrmUDPServer, 37
- id
  - mrmCursor, 18
  - mrmMouse, 25
- ieMouse
  - mrmMouse, 25
- inc/mrmCfgParser.h, 39
- inc/mrmCursor.h, 40
- inc/mrmCursorControl.h, 40
- inc/mrmMiceControl.h, 41
- inc/mrmMouse.h, 41
- inc/mrmMultiRemoteMouse.h, 41
- inc/mrmOSCursors.h, 42
- inc/mrmScreen.h, 42
- inc/mrmUDPCClient.h, 42
- inc/mrmUDPServer.h, 43
- inc/osinput.h, 43
- input\_event, 5
  - code, 5
  - time, 5
  - type, 5
  - value, 6
- ip
  - mrmScreen, 34
- isactive
  - mrmCursor, 14
- json
  - mrmCfgParser.h, 39
  - mrmCursorControl.h, 40
  - mrmMultiRemoteMouse.h, 42
  - mrmUDPServer.h, 43
- jsonobj
  - mrmCfgParser, 9
- jsonstr
  - mrmCfgParser, 9
- lastabsx
  - mrmCursor, 18
- lastabsy
  - mrmCursor, 18
- main
  - main.cpp, 46
- main.cpp
  - cfgfilename, 46
  - main, 46
  - mrm, 46
- mice
  - mrmMiceControl, 23
- micectrl
  - mrmMultiRemoteMouse, 29
- mousefile
  - mrmMouse, 25
- mrm
  - main.cpp, 46
  - mrmCursorControl.cpp, 47
  - mrmMouse.cpp, 48
  - mrmUDPServer.cpp, 49
- mrmCfgParser, 6
  - createLocalCursors, 7
  - createLocalMice, 7
  - createRemoteCursors, 7
  - createScreens, 8
  - getHostIP, 8
  - getUDPPort, 8
  - jsonobj, 9
  - jsonstr, 9
  - mrmCfgParser, 6
  - parse, 8
- mrmCfgParser.h
  - json, 39
- mrmCursor, 9
  - activate, 12
  - active, 16
  - click, 16
  - clickrunning, 16
  - cmap, 17
  - color\_b, 17
  - color\_g, 17
  - color\_r, 17
  - createTheRenderContext, 12
  - createTheWindow, 12
  - dataf, 17
  - del\_atom, 17
  - describe\_fbconfig, 12
  - fbconfig, 17

- fbconfigs, 17
- getDisplay, 12
- getPosx, 13
- getPosy, 13
- getXMax, 14
- getYMax, 14
- getid, 13
- getrelxfromabs, 13
- getrelyfromabs, 13
- glX\_window\_handle, 18
- height, 18
- id, 18
- isactive, 14
- lastabsx, 18
- lastabsy, 18
- mrmCursor, 10
- numfbconfigs, 18
- pict\_format, 18
- posx, 18
- posy, 19
- redrawTheWindow, 14
- relx, 19
- rely, 19
- render\_context, 19
- run, 15
- setPosx, 15
- setPosy, 15
- setlastabsx, 15
- setlastabsy, 15
- show, 19
- start, 16
- updateTheMessageQueue, 16
- VisData, 19
- visual, 19
- width, 20
- window\_handle, 20
- Xdisplay, 20
- Xroot, 20
- Xscreen, 20
- mrmCursor.cpp
  - ThreadCursor, 47
- mrmCursorControl, 20
  - amount, 22
  - amountlocal, 22
  - cursors, 22
  - getlist, 21
  - mrmCursorControl, 21
  - oscursor, 22
  - start, 21
  - work\_event, 21
- mrmCursorControl.cpp
  - mrm, 47
- mrmCursorControl.h
  - json, 40
- mrmMiceControl, 22
  - amount, 23
  - mice, 23
  - mrmMiceControl, 23
  - start, 23
- mrmMouse, 24
  - eventfile, 25
  - id, 25
  - ieMouse, 25
  - mousefile, 25
  - mrmMouse, 24
  - run, 25
  - start, 25
- mrmMouse.cpp
  - mrm, 48
  - ThreadMouse, 48
- mrmMultiRemoteMouse, 26
  - amountscreen, 29
  - client, 29
  - cursorctrl, 29
  - gethostip, 27
  - getscreen, 27
  - hostip, 29
  - micectrl, 29
  - mrmMultiRemoteMouse, 26
  - oscursor, 29
  - recvMouseEvent, 28
  - screens, 30
  - sendmouseevent, 28
  - sendscreenchange, 28
  - server, 30
  - start, 29
  - tempcount, 30
- mrmMultiRemoteMouse.h
  - json, 42
- mrmOSCursor, 30
  - clickPress, 31
  - clickRelease, 31
  - dpy, 32
  - getMutex, 31
  - mrmOSCursor, 31
  - mutex, 32
  - posx, 32
  - posy, 32
  - releaseMutex, 31
  - setCursor, 32
- mrmScreen, 33
  - getDirection, 33
  - getipaddr, 33
  - ip, 34
  - mrmScreen, 33
  - pos, 34
- mrmUDPClient, 34
  - addrout, 35
  - mrmUDPClient, 34
  - s, 35
  - sendevent, 35
- mrmUDPServer, 35
  - addrin, 37
  - hostip, 37
  - mrmUDPServer, 36
  - run, 36

- s, 37
- start, 36
- mrmUDPServer.cpp
  - mrm, 49
  - ThreadUDPServer, 49
- mrmUDPServer.h
  - json, 43
- mutex
  - mrmOSCursor, 32
- numfbconfigs
  - mrmCursor, 18
- oscursor
  - mrmCursorControl, 22
  - mrmMultiRemoteMouse, 29
- osinput.h
  - ABS\_WHEEL, 44
  - ABS\_X, 44
  - ABS\_Y, 44
  - BTN\_LEFT, 44
  - BTN\_MIDDLE, 44
  - BTN\_RIGHT, 44
  - EV\_ABS, 44
  - EV\_KEY, 44
  - EV\_REL, 45
  - REL\_WHEEL, 45
  - REL\_X, 45
  - REL\_Y, 45
- parse
  - mrmCfgParser, 8
- pict\_format
  - mrmCursor, 18
- pos
  - mrmScreen, 34
- posix
  - mrmCursor, 18
  - mrmOSCursor, 32
- posy
  - mrmCursor, 19
  - mrmOSCursor, 32
- REL\_WHEEL
  - osinput.h, 45
- REL\_X
  - osinput.h, 45
- REL\_Y
  - osinput.h, 45
- recvMouseEvent
  - mrmMultiRemoteMouse, 28
- redrawTheWindow
  - mrmCursor, 14
- releaseMutex
  - mrmOSCursor, 31
- relx
  - mrmCursor, 19
- rely
  - mrmCursor, 19
- render\_context
  - mrmCursor, 19
- run
  - mrmCursor, 15
  - mrmMouse, 25
  - mrmUDPServer, 36
- s
  - mrmUDPClient, 35
  - mrmUDPServer, 37
- screens
  - mrmMultiRemoteMouse, 30
- sendevent
  - mrmUDPClient, 35
- sendmouseevent
  - mrmMultiRemoteMouse, 28
- sendscreenchange
  - mrmMultiRemoteMouse, 28
- server
  - mrmMultiRemoteMouse, 30
- setCursor
  - mrmOSCursor, 32
- setPosx
  - mrmCursor, 15
- setPosy
  - mrmCursor, 15
- setlastabsx
  - mrmCursor, 15
- setlastabsy
  - mrmCursor, 15
- show
  - mrmCursor, 19
- src/main.cpp, 45
- src/mrmCfgParser.cpp, 46
- src/mrmCursor.cpp, 46
- src/mrmCursorControl.cpp, 47
- src/mrmMiceControl.cpp, 47
- src/mrmMouse.cpp, 48
- src/mrmMultiRemoteMouse.cpp, 48
- src/mrmOSCursor.cpp, 49
- src/mrmScreen.cpp, 49
- src/mrmUDPClient.cpp, 49
- src/mrmUDPServer.cpp, 49
- start
  - mrmCursor, 16
  - mrmCursorControl, 21
  - mrmMiceControl, 23
  - mrmMouse, 25
  - mrmMultiRemoteMouse, 29
  - mrmUDPServer, 36
- tempcount
  - mrmMultiRemoteMouse, 30
- ThreadCursor
  - mrmCursor.cpp, 47
- ThreadMouse
  - mrmMouse.cpp, 48
- ThreadUDPServer
  - mrmUDPServer.cpp, 49



- time
  - input\_event, [5](#)
- type
  - input\_event, [5](#)
- updateTheMessageQueue
  - mrmCursor, [16](#)
- value
  - input\_event, [6](#)
- VisData
  - mrmCursor, [19](#)
- visual
  - mrmCursor, [19](#)
- width
  - mrmCursor, [20](#)
- window\_handle
  - mrmCursor, [20](#)
- work\_event
  - mrmCursorControl, [21](#)
- Xdisplay
  - mrmCursor, [20](#)
- Xroot
  - mrmCursor, [20](#)
- Xscreen
  - mrmCursor, [20](#)