

Multiplot

0.5.5

Generated by Doxygen 1.8.15

1 Hierarchical Index	1
1.1 Class Hierarchy	1
2 Class Index	3
2.1 Class List	3
3 Class Documentation	5
3.1 <code>multiplot::Multiplot::Color3f</code> Class Reference	5
3.2 <code>multiplot::Multiplot</code> Class Reference	5
3.2.1 Detailed Description	7
3.2.2 Member Function Documentation	7
3.2.2.1 <code>bg_color()</code>	7
3.2.2.2 <code>clear()</code>	7
3.2.2.3 <code>clear_all()</code>	7
3.2.2.4 <code>color3f()</code>	7
3.2.2.5 <code>grid()</code>	8
3.2.2.6 <code>grid_color()</code>	8
3.2.2.7 <code>linewidth()</code>	8
3.2.2.8 <code>operator()()</code>	8
3.2.2.9 <code>operator[]()</code>	8
3.2.2.10 <code>plot()</code> [1/3]	8
3.2.2.11 <code>plot()</code> [2/3]	9
3.2.2.12 <code>plot()</code> [3/3]	9
3.2.2.13 <code>pointsize()</code>	9
3.2.2.14 <code>scaling()</code>	9
3.2.2.15 <code>scrolling()</code>	9
3.2.2.16 <code>sleep()</code>	9
3.2.2.17 <code>title()</code> [1/2]	10
3.2.2.18 <code>title()</code> [2/2]	10
3.2.2.19 <code>trace()</code>	10
3.3 <code>multiplot::Multiplot_base</code> Class Reference	10
3.3.1 Constructor & Destructor Documentation	11
3.3.1.1 <code>Multiplot_base()</code>	11
3.4 <code>multiplot::Multiplot::Point2d</code> Class Reference	11
3.5 <code>multiplot::Multiplot::Trace</code> Class Reference	12
3.5.1 Detailed Description	12
3.5.2 Member Function Documentation	12
3.5.2.1 <code>clear()</code>	12
3.5.2.2 <code>color3f()</code>	13
3.5.2.3 <code>linewidth()</code>	13
3.5.2.4 <code>plot()</code>	13
3.5.2.5 <code>pointsize()</code>	13
3.5.2.6 <code>scrolling()</code>	13

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

multiplot::Multiplot::Color3f	5
FI_GL_Window	
multiplot::Multiplot_base	10
multiplot::Multiplot	5
multiplot::Multiplot::Point2d	11
vector	
multiplot::Multiplot::Trace	12

Chapter 2

Class Index

2.1 Class List

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

multiplot::Multiplot::Color3f	5
multiplot::Multiplot	5
multiplot::Multiplot_base	10
multiplot::Multiplot::Point2d	11
multiplot::Multiplot::Trace	12

Chapter 3

Class Documentation

3.1 multiplot::Multiplot::Color3f Class Reference

Public Member Functions

- **Color3f** (float r_, float g_, float b_)

Public Attributes

- float **r** = 0.0f
- float **g** = 0.0f
- float **b** = 0.0f

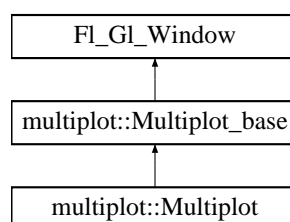
The documentation for this class was generated from the following file:

- multiplot.h

3.2 multiplot::Multiplot Class Reference

```
#include <multiplot.h>
```

Inheritance diagram for multiplot::Multiplot:



Classes

- class [Color3f](#)
- class [Point2d](#)
- class [Trace](#)

Public Member Functions

- **Multiplot** (const int x, const int y, const int w, const int h, const std::wstring &title_str=L"www.andre-krause.net/multiplot", bool fullscreen=false)
- [Trace](#) & [operator\[\]](#) (int _trace)
- [Trace](#) & [operator\(\)](#) (int _trace)
- [Trace](#) & [trace](#) (unsigned int _trace)
- void [plot](#) (const float x, const float y)
- template<class T >
void [plot](#) (const std::vector< T > &v)
- template<class T >
void [plot](#) (const std::vector< T > &vx, const std::vector< T > &vy)
- void [color3f](#) (float r, float g, float b)
- void [title](#) (const std::wstring &title_)
- void [title](#) (const std::string &title_)
- void [linewidth](#) (float width)
- void [pointsize](#) (float psize)
- void [scrolling](#) (int max_points_to_plot)
- void [scaling](#) (enum MP_SCALING sc, float x_min=-10, float x_max=10, float y_min=-10, float y_max=10)
- void [sleep](#) (unsigned int milliseconds_)
- void [grid](#) (enum MP_GRIDSTYLE ggridx=MP_LINEAR_GRID, enum MP_GRIDSTYLE ggridy=MP_LINEAR_GRID, float ggridx_step=-1.0, float ggridy_step=-1.0, float w=1.0)
- void [bg_color](#) (float r, float g, float b)
- void [grid_color](#) (float r, float g, float b)
- void [clear_all](#) ()
- void [clear](#) (int [trace](#))

Protected Member Functions

- void [initgl](#) ()
- [Point2d](#) [draw_grid](#) ()
- virtual void [draw](#) ()

Protected Attributes

- float [cur_point_size](#) = 0.0f
- unsigned int [cur_trace](#) = 0
- std::wstring [title_str](#)
- std::wstring [caption_str](#)
- [Color3f](#) [bg_col](#) { 0.0f, 0.0f, 0.0f }
- [Color3f](#) [grid_col](#) { 0.8f, 0.8f, 0.8f }
- MP_SCALING [scaling_](#) = MP_AUTO_SCALE
- [Point2d](#) [range_min](#)
- [Point2d](#) [range_max](#)
- [Point2d](#) [minimum](#) { -std::numeric_limits<float>::max() , -std::numeric_limits<float>::max() }
- [Point2d](#) [maximum](#) { std::numeric_limits<float>::max() , std::numeric_limits<float>::max() }

- [Point2d](#) **scale**
- [Point2d](#) **offset**
- std::vector< [Trace](#) > **traces**
- int **gridx** = MP_NO_GRID
- int **gridy** = MP_NO_GRID
- float **gridx_step** = -1
- float **gridy_step** = -1
- float **grid_linewidth** = 1.0f
- [Point2d](#) **grid_spacing**

3.2.1 Detailed Description

this class creates a window to wich you can add an arbitrary number of autoscaling traces.

3.2.2 Member Function Documentation

3.2.2.1 `bg_color()`

```
void multiplot::Multiplot::bg_color (
    float r,
    float g,
    float b ) [inline]
```

sets the background color

3.2.2.2 `clear()`

```
void multiplot::Multiplot::clear (
    int trace ) [inline]
```

this function call clears trace number t

3.2.2.3 `clear_all()`

```
void multiplot::Multiplot::clear_all ( ) [inline]
```

this function call simply clears all traces

3.2.2.4 `color3f()`

```
void multiplot::Multiplot::color3f (
    float r,
    float g,
    float b ) [inline]
```

change current drawing color for current trace.

3.2.2.5 grid()

```
void multiplot::Multiplot::grid (
    enum MP_GRIDSTYLE ggridx = MP_LINEAR_GRID,
    enum MP_GRIDSTYLE ggridy = MP_LINEAR_GRID,
    float ggridx_step = -1.0,
    float ggridy_step = -1.0,
    float w = 1.0 ) [inline]
```

call this function if you wish a grid to be plotted in your graph. by default, no grids are plotted. call this function with the first two arguments set to either MP_NO_GRID, MP_LINEAR_GRID or MP_LOG_GRID. the next two arguments gridx_step and gridy_step specify the grid spacing. Zero or a negative value like -1 enables auto - spacing. The last parameter w sets the grid-linewidth. the default is 1 pixel.

3.2.2.6 grid_color()

```
void multiplot::Multiplot::grid_color (
    float r,
    float g,
    float b ) [inline]
```

sets the grid color

3.2.2.7 linewidth()

```
void multiplot::Multiplot::linewidth (
    float width ) [inline]
```

changes current line width.

3.2.2.8 operator()

```
Trace& multiplot::Multiplot::operator() (
    int _trace ) [inline]
```

Access function. allows direct access to a trace.

3.2.2.9 operator[]()

```
Trace& multiplot::Multiplot::operator[] (
    int _trace ) [inline]
```

Access function. allows direct access to a trace.

3.2.2.10 plot() [1/3]

```
void multiplot::Multiplot::plot (
    const float x,
    const float y ) [inline]
```

plots a point at x,y to the currently active trace. select a trace with a call to trace(int _tracenum);

3.2.2.11 plot() [2/3]

```
template<class T >
void multiplot::Multiplot::plot (
    const std::vector< T > & v ) [inline]
```

plots a vector of values to the currently active trace. the x value is running from 0 .. vector.size()-1 select a trace with a call to trace(int _tracenum);

3.2.2.12 plot() [3/3]

```
template<class T >
void multiplot::Multiplot::plot (
    const std::vector< T > & vx,
    const std::vector< T > & vy ) [inline]
```

plots the values of vector vx and vy to the currently active trace. vx and vy must have the same length. select a trace with a call to trace(int _tracenum);

3.2.2.13 pointsize()

```
void multiplot::Multiplot::pointsiz (
    float psize ) [inline]
```

changes current point size.

3.2.2.14 scaling()

```
void multiplot::Multiplot::scaling (
    enum MP_SCALING sc,
    float x_min = -10,
    float x_max = 10,
    float y_min = -10,
    float y_max = 10 ) [inline]
```

changes the (auto-)scaling behaviour of the multiplot window. you can choose between MP_AUTO_SCALE MP_↵ AUTO_SCALE_EQUAL MP_FIXED_SCALE

3.2.2.15 scrolling()

```
void multiplot::Multiplot::scrolling (
    int max_points_to_plot ) [inline]
```

changes scrolling behaviour for current trace - see class [Trace](#) for details.

3.2.2.16 sleep()

```
void multiplot::Multiplot::sleep (
    unsigned int milliseconds_ ) [inline]
```

sleeps for the given amount of milliseconds useful to control the speed of animated graphs.

3.2.2.17 title() [1/2]

```
void multiplot::Multiplot::title (
    const std::wstring & title_ ) [inline]
```

sets the window title given a wide string.

3.2.2.18 title() [2/2]

```
void multiplot::Multiplot::title (
    const std::string & title_ ) [inline]
```

sets the window title given a string or char*.

3.2.2.19 trace()

```
Trace& multiplot::Multiplot::trace (
    unsigned int _trace ) [inline]
```

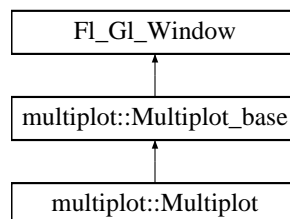
sets the current trace. traces are numbered from zero to N. memory for the traces is automatically allocated.

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

- multiplot.h

3.3 multiplot::Multiplot_base Class Reference

Inheritance diagram for multiplot::Multiplot_base:

**Public Member Functions**

- [Multiplot_base](#) (int x, int y, int w, int h, const std::wstring &title_, bool fullscreen_)
- bool **check** ()
- virtual void **draw** () override
- void **caption** (const std::string &t)
- void **caption** (const std::wstring &t)
- void **redraw** ()

Protected Attributes

- unsigned int **width** = 0
- unsigned int **height** = 0
- std::string **caption_str**

3.3.1 Constructor & Destructor Documentation

3.3.1.1 Multiplot_base()

```
multiplot::Multiplot_base::Multiplot_base (
    int x,
    int y,
    int w,
    int h,
    const std::wstring & title_,
    bool fullscreen_ ) [inline]
```

this constructor tells multiplot where to put the window on the desktop in pixel-coordinates(x,y) and with width and height (w,h)

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

- multiplot.h

3.4 multiplot::Multiplot::Point2d Class Reference

Public Member Functions

- **Point2d** (float xx, float yy, float rr=1, float gg=1, float bb=1, float _lwidth=1.0, float _point_size=0.0)

Public Attributes

- float **x** = 0.0f
- float **y** = 0.0f
- float **r** = 1.0f
- float **g** = 1.0f
- float **b** = 1.0f
- float **point_size** = 0.0f
- float **line_width** = 1.0f

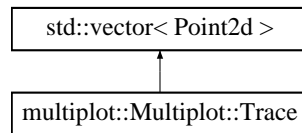
The documentation for this class was generated from the following file:

- multiplot.h

3.5 multiplot::Multiplot::Trace Class Reference

```
#include <multiplot.h>
```

Inheritance diagram for multiplot::Multiplot::Trace:



Public Member Functions

- void **draw** ([Point2d](#) &minimum, [Point2d](#) &maximum, const [Point2d](#) &scale, const [Point2d](#) &offset)
- void **plot** (const float x, const float y)
- void **color3f** (float r, float g, float b)
- void **linewidth** (float width)
- void **pointsize** (float psize)
- void **scrolling** (int number_of_points_to_plot_)
- void **clear** ()

Public Attributes

- unsigned int **max_points_to_plot** = std::numeric_limits<unsigned int>::max()
- bool **scroll** = false
- unsigned int **pos** = 0
- float **cur_col** [3] { 1.0f, 1.0f, 1.0f }
- float **cur_line_width** = 1.0f
- float **cur_point_size** = 0.0f

3.5.1 Detailed Description

class [Trace](#) describes a single [Trace](#). A Multiplot-Window can contain an unlimited number of Traces.

3.5.2 Member Function Documentation

3.5.2.1 clear()

```
void multiplot::Multiplot::Trace::clear ( ) [inline]
```

[clear\(\)](#) removes all points from the trace. the trace is empty afterwards and can be filled with plot(x,y) again.

3.5.2.2 color3f()

```
void multiplot::Multiplot::Trace::color3f (
    float r,
    float g,
    float b ) [inline]
```

sets the current drawing color in rgb format. r,g,b are in the range [0..1]

3.5.2.3 linewidth()

```
void multiplot::Multiplot::Trace::linewidth (
    float width ) [inline]
```

call linewidth to change the thickness of the traces. the default value is 1 pixel, if you set the linewidth to zero, no lines are drawn. this is usefull to create scatter-plots.

3.5.2.4 plot()

```
void multiplot::Multiplot::Trace::plot (
    const float x,
    const float y ) [inline]
```

plot a point at (x,y) to the currently active trace. you may switch the trace with a call to trace(int _trace)

3.5.2.5 pointsize()

```
void multiplot::Multiplot::Trace::pointsize (
    float psize ) [inline]
```

this function sets the size of the plot-points. the default value is zero, so no points are drawn at all. if you wish to create a scatter-plot, set the pointsize to a value bigger than zero and the linesize to zero.

3.5.2.6 scrolling()

```
void multiplot::Multiplot::Trace::scrolling (
    int number_of_points_to_plot_ ) [inline]
```

if you call scrolling with a positive number of points to be plotted, your graph will scroll left out of the plot-window as you add new plot-points beyond number_of_points_to_plot_. Zero or a negative number disables scrolling.

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

- multiplot.h

Index

- bg_color
 - multiplot::Multiplot, 7
- clear
 - multiplot::Multiplot, 7
 - multiplot::Multiplot::Trace, 12
- clear_all
 - multiplot::Multiplot, 7
- color3f
 - multiplot::Multiplot, 7
 - multiplot::Multiplot::Trace, 12
- grid
 - multiplot::Multiplot, 7
- grid_color
 - multiplot::Multiplot, 8
- linewidth
 - multiplot::Multiplot, 8
 - multiplot::Multiplot::Trace, 13
- multiplot::Multiplot, 5
 - bg_color, 7
 - clear, 7
 - clear_all, 7
 - color3f, 7
 - grid, 7
 - grid_color, 8
 - linewidth, 8
 - operator(), 8
 - operator[], 8
 - plot, 8, 9
 - pointsize, 9
 - scaling, 9
 - scrolling, 9
 - sleep, 9
 - title, 9, 10
 - trace, 10
- multiplot::Multiplot::Color3f, 5
- multiplot::Multiplot::Point2d, 11
- multiplot::Multiplot::Trace, 12
 - clear, 12
 - color3f, 12
 - linewidth, 13
 - plot, 13
 - pointsize, 13
 - scrolling, 13
- multiplot::Multiplot_base, 10
 - Multiplot_base, 11
- multiplot::Multiplot_base, 11
- operator()
 - multiplot::Multiplot, 8
- operator[]
 - multiplot::Multiplot, 8
- plot
 - multiplot::Multiplot, 8, 9
 - multiplot::Multiplot::Trace, 13
- pointsize
 - multiplot::Multiplot, 9
 - multiplot::Multiplot::Trace, 13
- scaling
 - multiplot::Multiplot, 9
- scrolling
 - multiplot::Multiplot, 9
 - multiplot::Multiplot::Trace, 13
- sleep
 - multiplot::Multiplot, 9
- title
 - multiplot::Multiplot, 9, 10
- trace
 - multiplot::Multiplot, 10