
SVGraph

WebCom, s.r.o.

Roman Fordinál <fordinal@webcom.sk>

Matej Gregor <gregor@webcom.sk>

Michal Ondrovič <ondrovic@webcom.sk>

This program is free software; you can redistribute it and/or modify it under LGPLv2.

YYYY-MM-DD

Prehľad revízií

Revízia 1	2005-01-07	RomanFordinál
	Creation of this documentation.	
Revízia 2	2005-07-07	MatejGregor
	Change of structure of this document, added new information.	
Revízia 3	2006-11-07	MichalOndrovič
	Added new information.	
Revízia 4	2008-02-20	RomanFordinál
	Legal notice update.	

Abstrakt

SVGraph Library

This library is used for generation of graphs in SVG format. With the help of the SVGraph library you should easily and intelligently design your own graph and its range.

SVGraph::2D::columns

Class for column graphs.

Dictionary

New object of a column graph

Creation of a new object of a column graph :

```
my $graph=SVGraph::2D::column->new(%params);
```

%param:

- title

Title of the graph. It will be displayed in the body of the graph.

- type

Type of drawing of the graph:

- normal
- normal/percentage
- overlap

- x
Width of the graph.
- y
Height of the graph.
- reload
Time in seconds for automatical refresh of the graph. It is assured by the javascript function reload().
- show_legend
1/0
Displays the legend on the right site of the graph.
- show_legend_reverse
1/0
Displays the legend in reverse order.
- show_areas
1/0
Displays the areas in graph.
- show_label_textsize
Textsize in labels on the x axis.
- show_areas_opacity
0.0 -> 1.1
Shows areas with defined alpha.
- show_data
1/0
Shows the values above each column.
- show_data_background
1/0
Shows the values above columns with background.
- show_grid_x
1/0
Shows the grid on x axis and the labels.
- grid_y_scale_minimum
Defines the starting value of the graph.
- grid_y_scale_maximum

Defines the maximum value displayed on the y axis

- `grid_y_main_spacing`

Defines spacing in the grid on the y axis.

Adding columns into graph

Columns are data accumulators of the same type.

For adding new columns in the graph is used following method:

```
my $column = $graph->addColumn(%params);
```

%params :

- `title`

Name of the column.

- `show_area`

0/1

Shows the columns as areas.

- `show_line`

0/1

Allows to display the column as a line.

- `color`

Color of the column.

Adding row labels

For adding labels to rows is used the following method:

```
$graph->addRowLabel($label_text);
```

Parameters:

- `$label_text`

Text, which identifies the row.

Adding data to graph

```
$column->addData($label_text, value);
```

Parameters:

- `$label_text`

Text in row label.

Adding value marks

```
$graph->addValueMark(value,%params);
```

%params:

- front

1/0

The markline is in the front of the graph.

- size

Size of the line.

- color

Color of the line.

- right

0/1

Is the label on the right (1) or left (0) side?

- show_label

Shows the label. Default is it the value of the mark.

- show_label_text

Shows a text as the label.

Example

Code:

```
#!/bin/perl
use SVGraph::2D::columns;
use strict;

my $graph=SVGraph::2D::columns->new(
    title => "My graph",
    type => "normal",
    x => 500,
    y => 300,
    show_legend => 1,
    show_data => 1,
    show_data_background =>1,
    grid_y_scale_maximum => 70.00,
    show_grid_x => 1
);

# adding two columns
my %columns;
$columns{first}=$graph->addColumn(title=>"first");
$columns{second}=$graph->addColumn(title=>"second");

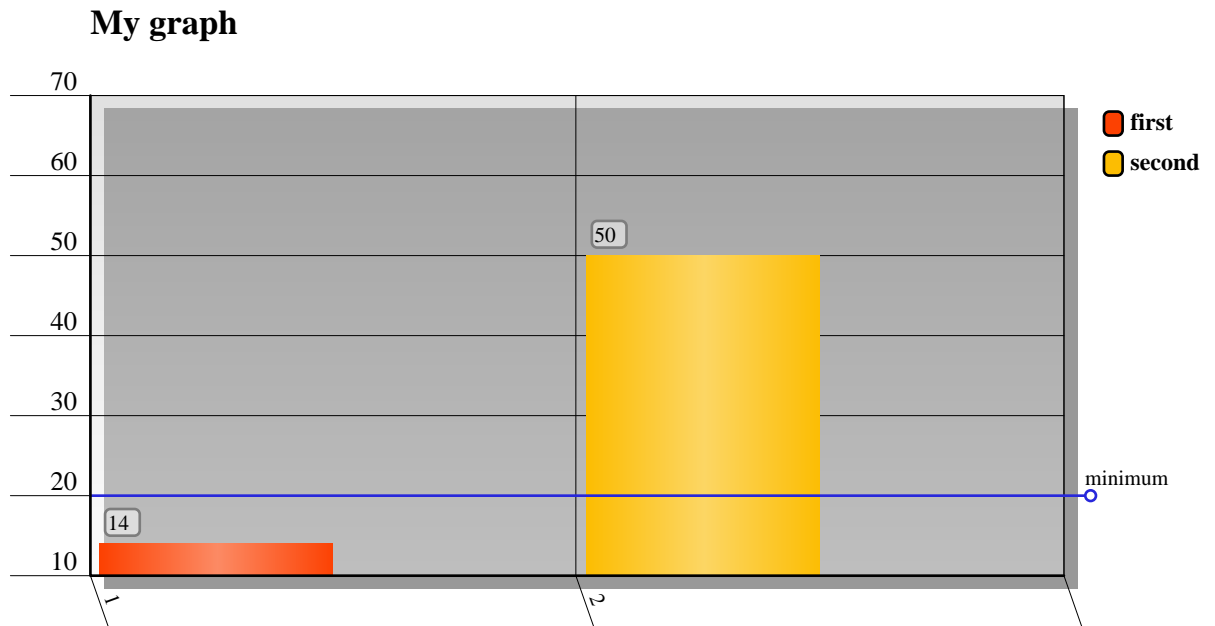
# adding rows and labels
$graph->addRowLabel('1');
$graph->addRowLabel('2');

# data for graph
$columns{first}->addData('1',14);
```

```
$columns{second}->addData('2',50);

$graph->addValueMark(20,front => 1, color => 'blue',
                    right => 1, show_label_text => 'minimum', show_label => 1);

# create svg file
open (my $fh, "+>graph.svg");
print $fh $graph->prepare;
```



SVGraph::2D::lines

Class for the line graphs.

Dictionary

New object of a line graph

Creation of a new object of a line graph :

```
my $graf=SVGraph::2D::lines->new(%params)
```

%params:

- title

Title of the graph. It will be displayed in the body of the graph.

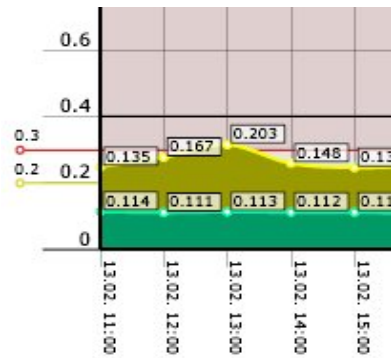
- type

Type of drawing of the graph:

- normal

- normal/percentage
- stacked

The values are counted together.



- stacked/percentage
- overlap
- x

Width of the graph.

- y

Height of the graph.

- reload

Time in seconds for automatical refresh of the graph. It is assured by the javascript function reload().

- show_legend

1/0

Displays the legend on the right site of the graph.

- show_legend_reverse

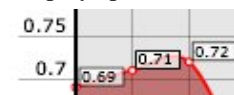
0/1

Displays the legend in reverse order.

- show_points

0/1

Displays points in the graph.



- show_points_middle

0/1

Displays point in the middle of defined values.

- show_lines

1/0

Shows lines in the line graph.

- show_lines_smooth

0/1

Rounds the lines in the graph.

- show_areas

1/0

Displays the areas in graph.

- show_label_textsize

Textsize in labels on the x axis.

- show_areas

1/0

Dislpays the areas.

- show_areas_opacity

0.0 -> 1.1

Shows areas with defined alpha.

- show_lines_smooth_range

1/0

Smooths ranges around the lines.

- show_data

1/0

Shows the values above each column.

- show_data_background

1/0

Shows the values above columns with background.

- show_grid_x

1/0

Shows the grid on x axis and the labels.

- grid_y_scale_minimum

Defines the starting value of the graph.

- grid_y_scale_maximum

Defines the maximum value displayed on the y axis

- `grid_y_main_spacing`

Defines spacing in the grid on the y axis.

Adding columns into graph

Columns are data accumulators of the same type.

For adding new columns in the graph is used following method:

```
my $column = $graph->addColumn(%params);
```

%params :

- `title`

Name of the column.

- `show_area`

0/1

Shows the columns as areas.

- `show_line`

0/1

Allows to display the column as a line.

- `color`

Color of the column.

Adding row labels

For adding labels to rows is used the following method:

```
$graph->addRowLabel($label_text);
```

Parameters:

- `$label_text`

Text, which identifies the row.

Adding data to graph

```
$column->addData($label_text, value);
```

Parameters:

- `$label_text`

Text in row label.

Adding value marks

```
$graph->addValueMark(value,%params);
```

%params:

- front

1/0

The markline is in the front of the graph.

- size

Size of the line.

- color

Color of the line.

- right

0/1

Is the label on the right (1) or left (0) side?

- show_label

Shows the label. Default is it the value of the mark.

- show_label_text

Shows a text as the label.

Example

Code:

```
#!/bin/perl
use strict;
use SVGraph::2D::lines;

my $graph=SVGraph::2D::lines->new(
    title => "My graph",
    type => "normal/percentage",
    x => 500,
    y => 300,
    show_lines => 1,
    show_points => 1,
    grid_y_main_lines => 0,
    show_label_textsize => 20,
    show_legend => 1,
    show_data => 1,
    show_data_background => 1,
    show_grid_x => 0
);

# two columns -> two lines in the graph
my %columns;
$columns{first}=$graph->addColumn(title=>"first", show_line=>1);
$columns{second}=$graph->addColumn(title=>"second", show_line=>1);

$graph->addRowLabel('1');
$graph->addRowLabel('2');
$graph->addRowLabel('3');
```

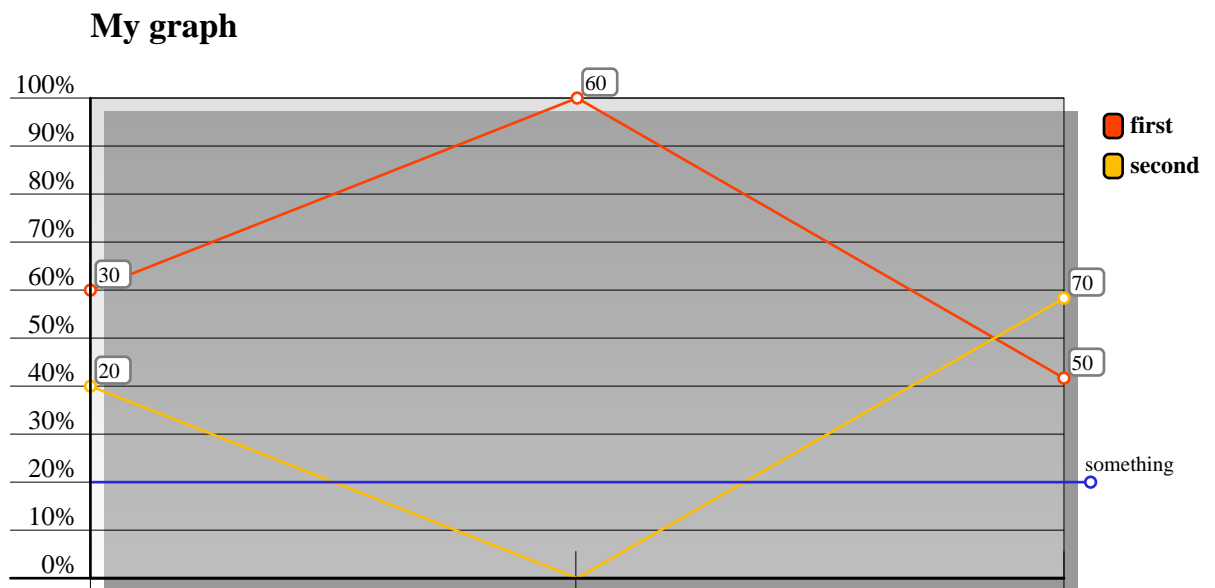
```

$graph->addValueMark(20,front => 1, color => 'blue', right => 1,
                    show_label_text => 'something', show_label => 1);

# data for graph
$columns{first}->addData('1',30);
$columns{first}->addData('2',60);
$columns{first}->addData('3',50);
$columns{second}->addData('1',20);
$columns{second}->addData('2',0);
$columns{second}->addData('3',70);

# creates svg file
open (my $fh, "+>graph.svg");
print $fh $graph->prepare;

```



SVGraph::2D::map

Class for graph maps.

Dictionary

New object of a map

Creation of a new object of a map graph :

```
my $map=SVGraph::2D::map->new(%params);
```

%param:

- title

Title of the graph.

- continent

Continent to be displayed. For whole world is it 'world', for slovakia is this parameter 'slovakia'.

- x

Width of the graph.

- y

Height of the graph.

- reload

Time in seconds for automatical refresh of the graph. It is assured by the javascript function reload().

- show_legend

1/0

Disables the legend.

Adding columns into graph

Columns are data accumulators of the same type. In the map is only one column.

Creating new column :

```
my $column = $map->addColumn(%params);
```

%params :

- title

Name of the coulumn.

- color

Color of the countries.

Adding row labels

Row labels identifies countries, which have defined value. For adding labels to rows is used the following method:

```
$map->addRowLabel($label_text);
```

Parameters:

- \$label_text

Text, which identifies the row.

Adding data to graph

```
$column->addData($label_text, value);
```

Parameters:

- \$label_text

Text in row label.

Example

Code:

```
#!/bin/perl

use SVGraph::2D::map;
use strict;

my $svgfh;

open($svgfh,">map.svg") or die $!;

my $map = SVGraph::2D::map->new(title => 'europe', continent => 'europe', x => .

my $column = $map->addColumn(title => 'europe', color => 'blue');

$map->addRowLabel('SK');
$column->addData('SK', 2520);

$map->addRowLabel('CZ');
$column->addData('CZ', 1000);

$map->addRowLabel('GB');
$column->addData('GB', 545);

print $svgfh $map->prepare();
```