

Renaud Guezennec

<http://renaudguezennec.homelinux.org/>

SGNGD: SvgGd is Not GD

User Manual

Table of Content

SGNGD: SvgGd is Not GD.....	1
User Manual.....	1
1.Description.....	3
2.Installation.....	3
3.How to use (Quick start).....	3
How to add the generated SVG chart into a html/xhtml page?.....	4
4.BarChart.....	4
5.PieChar.....	6
How to group many little values in once ?.....	9
6.PointCloud Chart.....	12
7.Colour.....	14
8.Gallery.....	15
9.Conclusion	17

Table of figures

Figure 1: the Pie chart with no exploded part.....	8
Figure 2: the Pie chart with one exploded part.....	8
Figure 3: the Pie chart with all exploded parts.....	9
Figure 4: Pie Chart with grouped little values.....	10
Figure 5: Pie Chart without grouped little values.....	11
Figure 6: Point Cloud Chart.....	13
Figure 7: Pie Chart.....	15
Figure 8: Pie chart with one exploded part.....	15
Figure 9: Pir Chart with all exploded parts.....	16
Figure 10: Bar Chart.....	17

1. Description

SGNGD is a free software to help you for generating bar chart, pie chart and few other chart kinds .

Based-on SVG technology, charts are more efficient. With few code lines, you can create nice charts.

Its name means SvgGd is Not GD. GD is a php library which manages and creates image (such as jpg, .png and .gif). I used it for generating chart as a image. But this usage is not really relevant. Then, I decided to write few classes to draw chart using SVG technology. I think, it is a more clever choice.

2. Installation

Extract the archive in a directory of your web site. The archive contains these files:

- Chart.php: Abstract class.
- PieChart.php: The class required to use a pie chart.
- BarChart.php : The class required to use a bar chart.
- PointCloudChart.php : The class required to use a PointCloudChart.
- Color.php : The color class. Useful
- test.php : read this file if you want an example of BarChart use.
- test2.php : read this file if you want an example of BarChart use.
- test3.php : read this file if you want an example of PointCloudChart use.
- COPYING : The text of General Public Licence
- README.txt

3. How to use (Quick start)

- Create a new file.
- Include SGNGD: `require_once (path_to_SGNGD/Chart.php);`
- Make a new instance of which chart you want: `PieChart` or `BarChart` (others are coming soon). `$class = new BarChart ();`
- Add items by calling `$class->AddData (name ,value);`
- Call `$class->compute () .`
- Define the type of page by call header: `header ('Content-Type: image/svg+xml ');`

and display the chart by calling `echo $class->display();`

To change the size of the view box, you can define the width and the height by calling these functions: `setWidth($w), setHeight($h)`. Do it before calling "compute" method.

You can manage colours of your chart. Please see the appropriate chapter for more informations.

How to add the generated SVG chart into a html/xhtml page?

There are many way to do that. So, I use this one:

```
<object type="image/svg+xml" data="url_to_myfile/piechart.php">
    
</object >
```

Add an object. Be careful about the type. Define the data field to your php file. The image defines by the `img` balise will be displayed if the visitor's web browser does not support the SVG.

4. BarChart

I will describe the given example and I will try to explain "how does it work".

```
<?php
/**!
 *
 * Copyright 2008 Renaud Guezenec
 *
 * This file is part of SGNGD.
 *
 * SGNGD is free software: you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation, either version 3 of the License, or
 * (at your option) any later version.
 *
 * SGNGD distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 * GNU General Public License for more details.
 *
 * You should have received a copy of the GNU General Public License
 * along with SGNGD. If not, see <http://www.gnu.org/licenses/>.
```

```

*
* @Author:Renaud Guezennec
* Contact me: http://renaudguezennec.homelinux.org/index,1,3.html
*
* @desc: test of BarChart class.
*
*/
require_once('Chart.php');

$chartsvg = new BarChart("BarChart SVG");
$chartsvg->setStartColor(new Color(0,0,255));
$chartsvg->setEndColor(new Color(0,255,0));

$chartsvg->AddData('cv',18);
$chartsvg->AddData('Mega',-5);
$chartsvg->AddData('com',8);
$chartsvg->AddData('com2',9);
$chartsvg->AddData('com3',20);
$chartsvg->AddData('com4',1);
$chartsvg->AddData('comder',6);

$chartsvg->SetScale(5);
$chartsvg->Compute();
header('Content-Type: image/svg+xml');
echo $chartsvg->display();
?>

```

As I said in the quick start chapter. The first thing is: include the file Chart.php. The next step is to create an instance of BarChart. The constructor can take few parameters:

- A string as the title of the chart.
- A colour behaviour: You have the choice between the NORMAL mode and the FULLDEFINED mode. The default value is the normal mode.(see more informations in the colour chapter)

After that, if you select the normal mode, you can choose two colours. All items will be colourised by a colour between these both. (The default value is white and black.)

Now, you can add items. The AddData method can takes three parameters:

- A string: the name of Item. This string appears when the cursor is hover the item.
- The value, It can be a signed int/double/float.
- The third one is the colour. It is required if you choose FULLDEFINED mode.

The method “compute” is compulsory. In this method, SGNGD calculates all values required for drawing the chart.

Thank to the definition of the header, your web browser can understand that the following code is SVG code.

The display method returns a string of the whole svg. So, you have to print or display the string to draw the SVG.

5. PieChar

```
<?php
/**!
 *
 * Copyright 2008 Renaud Guezenec
 *
 * This file is part of SGNGD.
 *
 * SGNGD is free software: you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation, either version 3 of the License, or
 * (at your option) any later version.
 *
 * SDNGD distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 * GNU General Public License for more details.
 *
 * You should have received a copy of the GNU General Public License
 * along with SGNGD. If not, see <http://www.gnu.org/licenses/>.
 *
```

```
* @Author:Renaud Guezennec
* Contact me: http://renaudguezennec.homelinux.org/index,1,3.html
*
* @desc: test of PieChart class.
*
*/
require_once('Chart.php');
$chartsvg = new PieChart("PieChart SVG",FULLDEFINED);

$chartsvg->setStartColor(new Color(255,10,10));
$chartsvg->setEndColor(new Color(0,250,250));

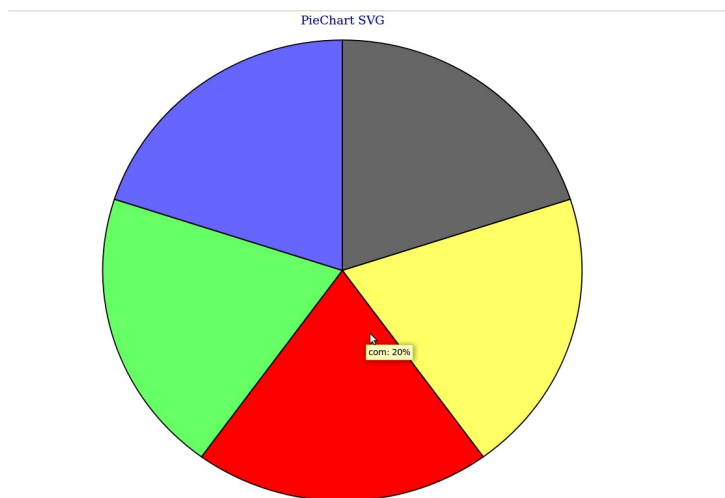
$chartsvg->AddData('cv',20,true,new Color(0,0,0));
$chartsvg->AddData('Mega',20,false,new Color(255,255,0));
$chartsvg->AddData('com',20,false,new Color(255,0,0));
$chartsvg->AddData('com2',20,false,new Color(0,255,0));
$chartsvg->AddData('com3',20,false,new Color(0,0,255));
//$chartsvg->AddData('com4',1);
//$chartsvg->AddData('comder',4);

$chartsvg->Setlimit(1,"others",false);
$chartsvg->Compute();
header('Content-Type: image/svg+xml');
echo $chartsvg->display();
?>
```

Here, you have an example of a FULLDEFINED mode use.

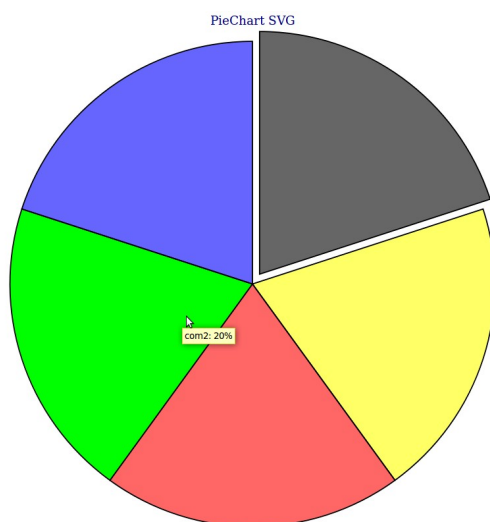
Make a PieChart start exactly like a BarChart. The first difference is in the addData call.

As you can see, there is a boolean parameter. This parameter enables or disables the explosion of this item.



Powered by SGNPD

Figure 1: the Pie chart with no exploded part



Powered by SGNPD

Figure 2: the Pie chart with one exploded part

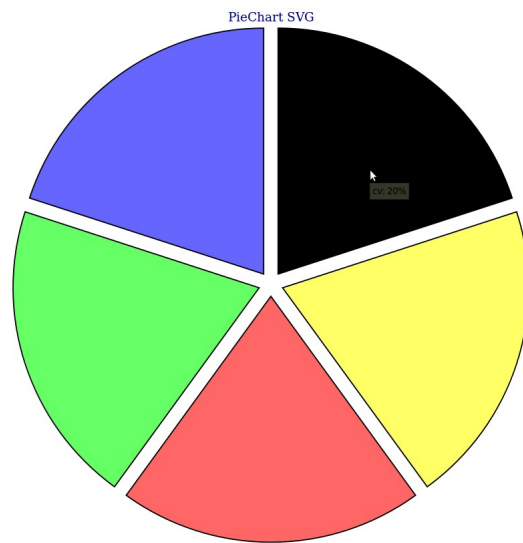


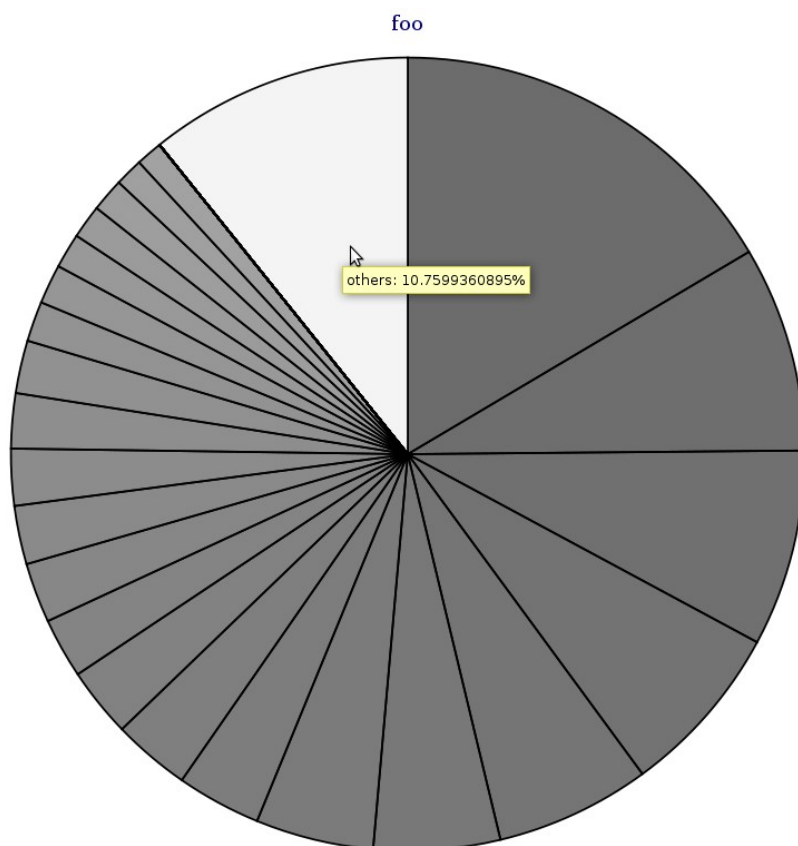
Figure 3: the Pie chart with all exploded parts

How to group many little values in once ?

PieChart provides one method called “Setlimit(limit ,title ,exploded [, Color])”

The Setlimit call must be done before the compute call.

SGNGD PieChart sums every values lower than “limit” to create a new Pie part.



Powered by SGNPD

Figure 4: Pie Chart with grouped little values

As you can see, the name of last part is “others”. Its value is the sum of every value lower than “1%”

Now the same PieChart without this option:

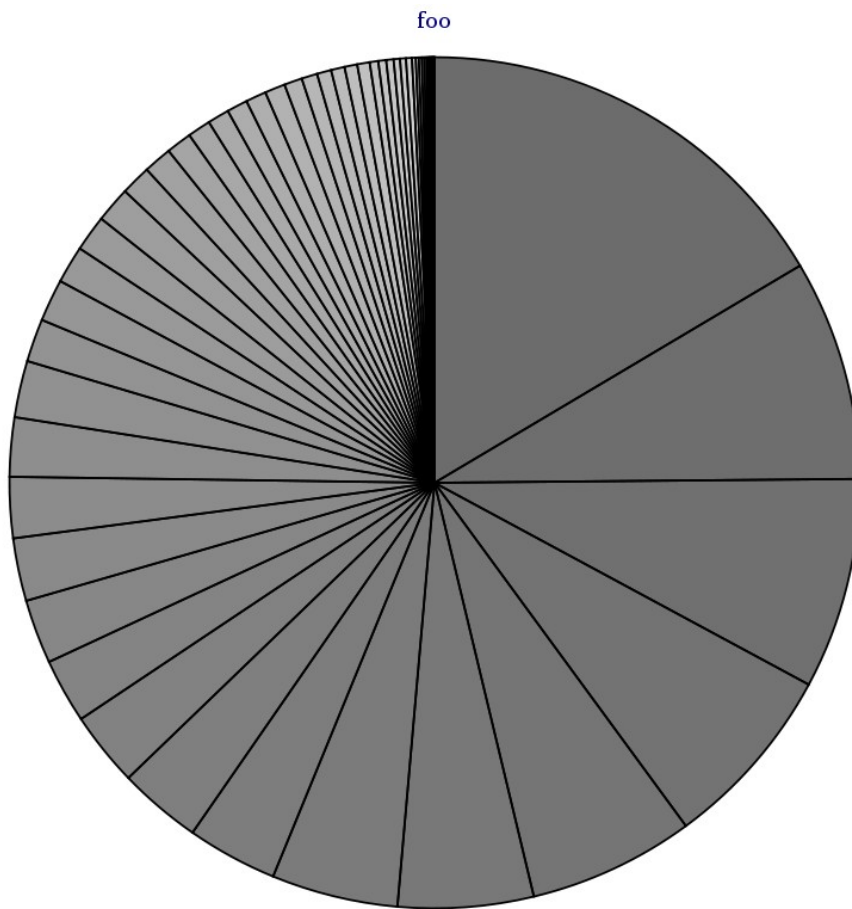


Figure 5: Pie Chart without grouped little values

Powered by SGNPD

6. PointCloud Chart

```
<?php
/**!
 *
 * Copyright 2008 Renaud Guezenec
 *
 * This file is part of SGNGD.
 *
 * SGNGD is free software: you can redistribute it and/or modify
 * it under the terms of the GNU General Public License as published by
 * the Free Software Foundation, either version 3 of the License, or
 * (at your option) any later version.
 *
 * SDNGD distributed in the hope that it will be useful,
 * but WITHOUT ANY WARRANTY; without even the implied warranty of
 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
 * GNU General Public License for more details.
 *
 * You should have received a copy of the GNU General Public License
 * along with SGNGD. If not, see <http://www.gnu.org/licenses/>.
 *
 * @Author:Renaud Guezenec
 * Contact me: http://renaudguezenec.homelinux.org/index,1,3.html
 *
 * @desc: test of PieChart class.
 *
 */
require_once('Chart.php');
$chartsvg = new PointCloudChart("PointCloudChart SVG");
$chartsvg->setStartColor(new Color(0,0,0));
$chartsvg->setEndColor(new Color(0,0,0));
for($i = -50 ; $i <= 50 ; $i++)
{
    $chartsvg->AddData($i,$i,$i );
}
$chartsvg->SetScale(5);
$chartsvg->Compute();
```

```
header('Content-Type: image/svg+xml');  
echo $chartsvg->display();  
?>
```

For the PointCloudChart, the only thing that worth to be noted is the AddData function. The first parameter is still the title. The second one is the X coordinate and the last one is the Y coordinate of the point.

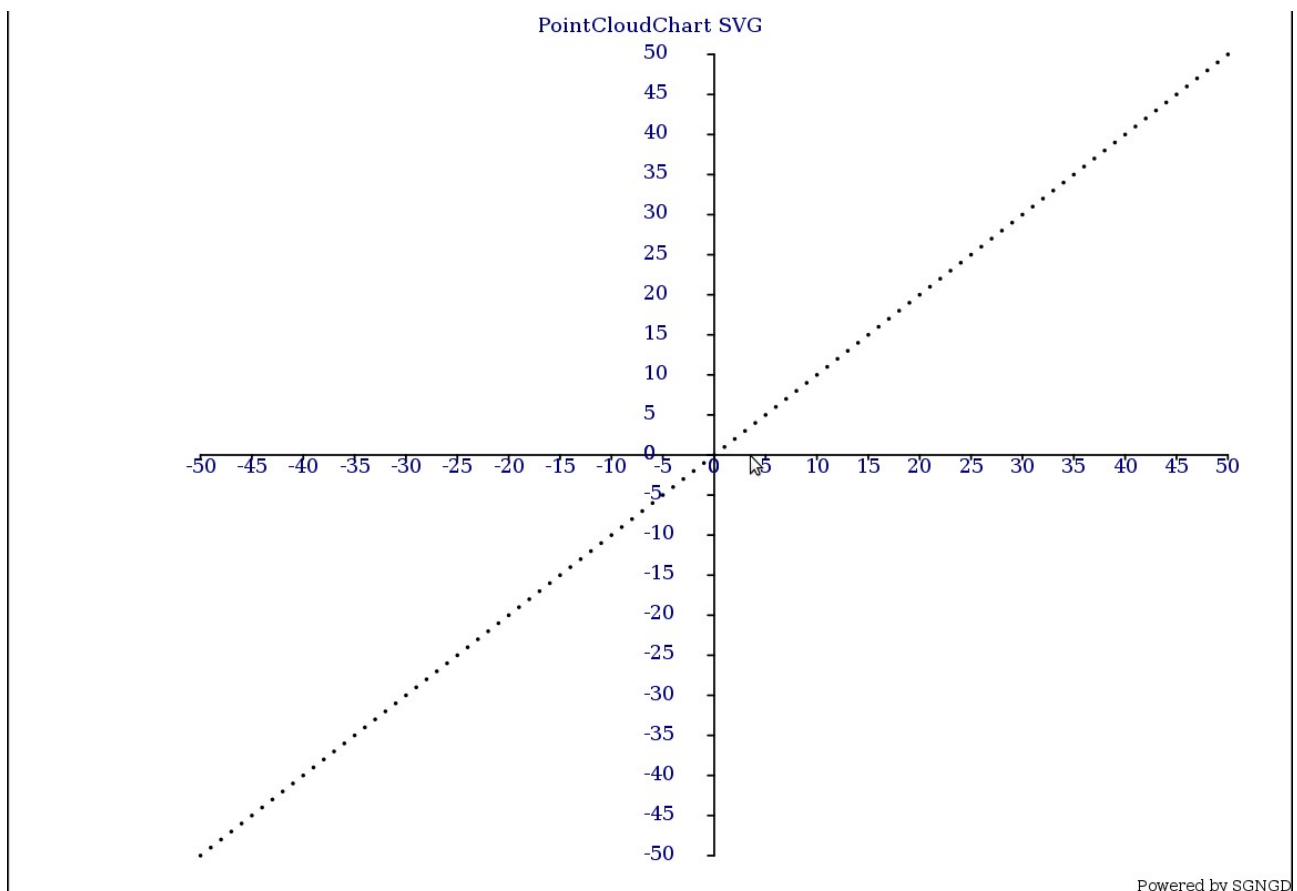


Figure 6: Point Cloud Chart

Powered by SGNCD

7. Colour

In SGNGD, the colours are represented into the “Color” class. Create an instance of this class is very easy. The constructor takes threes parameters: the value of each channel: red, green and blue. You need to build instance of this class when you use the FULLDEFINED mode or when you redefine the colours of the normal mode.

The normal mode calculates the colour between two others. The mathematical tool to do that is the linear interpolation([wiki](#)). The default values are black and white.

The FULLDEFINED mode means that SGNGD takes the given values in the AddData calls.

In the future, SGNGD will manage other colour modes: such as random mode and only one colour. If you need other modes please contact me.

8. Gallery

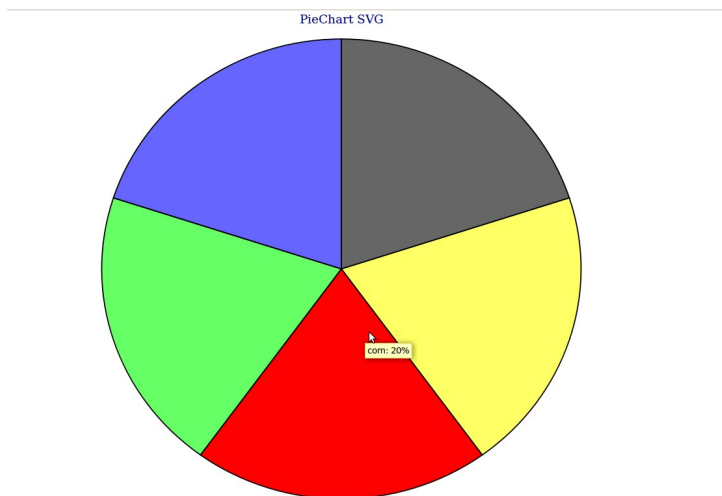


Figure 7: Pie Chart

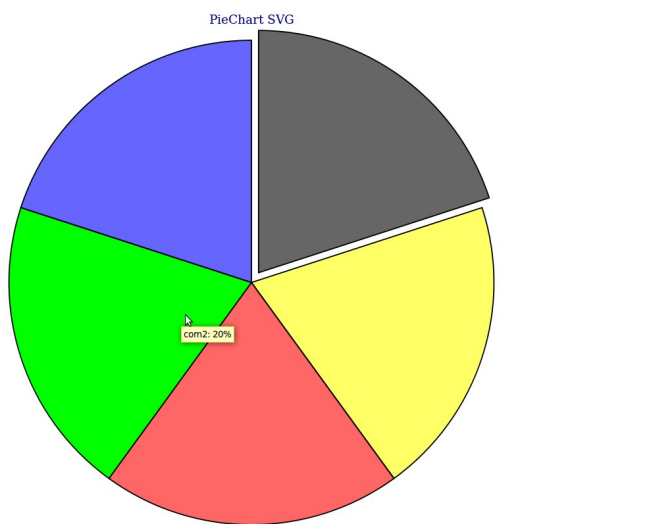
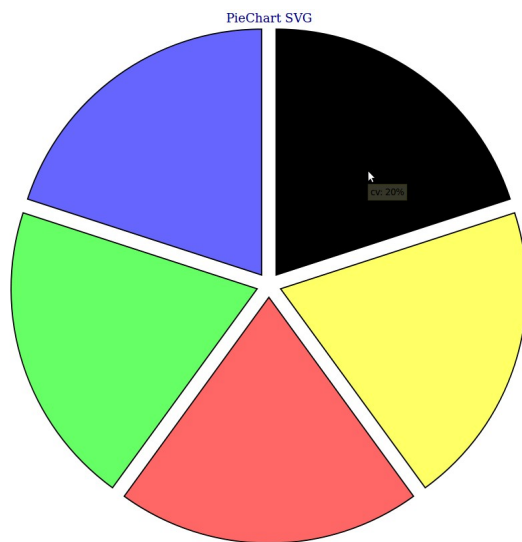


Figure 8: Pie chart with one exploded part



Powered by SGNGD

Figure 9: Pie Chart with all exploded parts

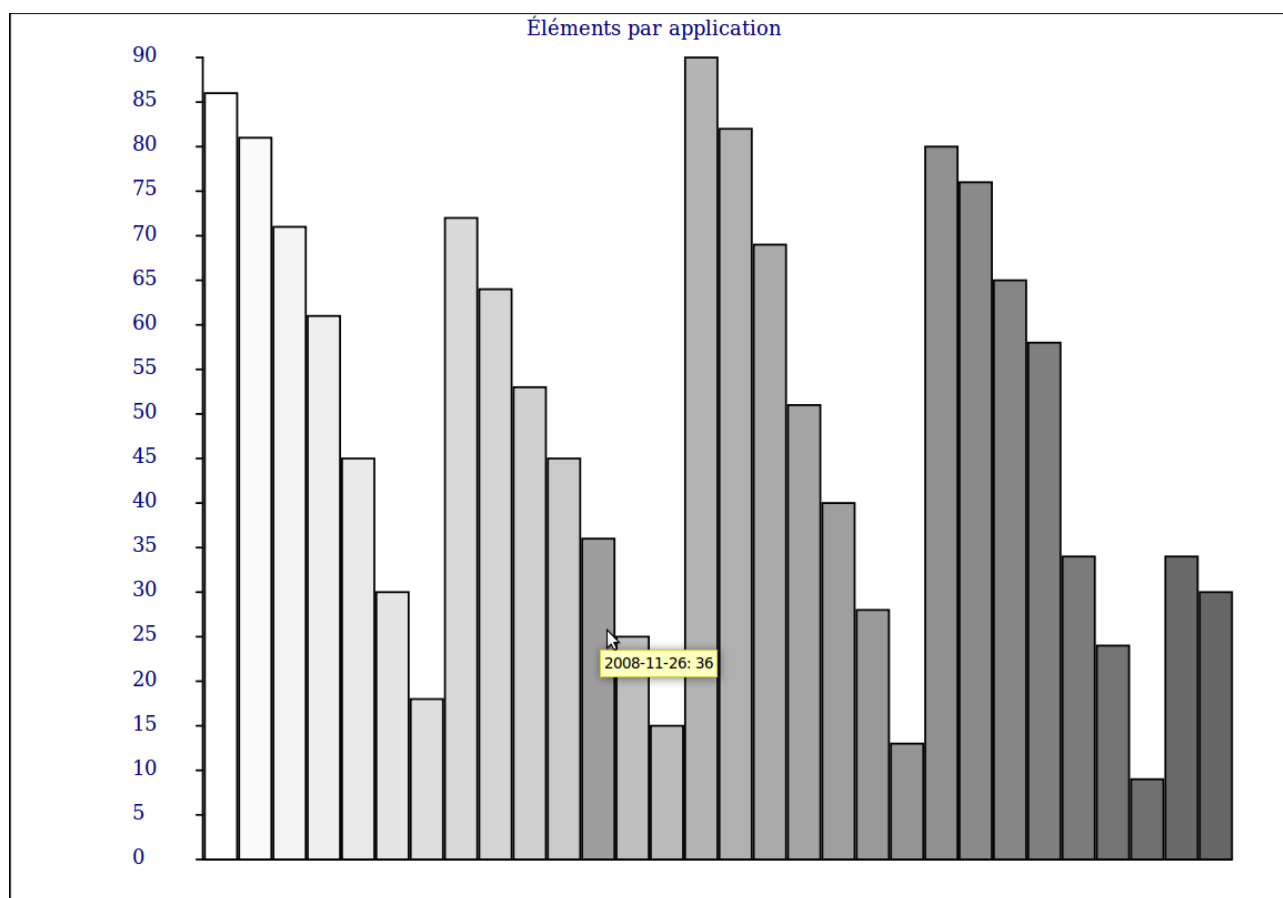


Figure 10: Bar Chart

9. Conclusion

It is just a beginning but SGNGD provides powerful tools to build chart easily. If you have any questions or any improvement requests or if you need help with SGNGD. Contact me through <http://renaudguezennec.homelinux.org/index,1.3.html>

See also:

<http://www.php.net/> : Php manual.

<http://renaudguezennec.homelinux.org/> : my web site (French)

<http://renaudguezennec.homelinux.org/index,2,1-2.html> : the SGNGD's web page.

http://en.wikipedia.org/wiki/Scalable_Vector_Graphics : the SVG wikipedia page.

<http://www.w3.org/TR/SVG/> : Specification of SVG v1.1.