

Grafana plugin modification

Blendstat Panel

First steps:

- Install Ubuntu in a Virtual Machine.
- Install Grafana and a Data source, in this case Prometheus with node exporter.
- Install Blendstat Panel plugin.
- Install nodejs, npm and grunt.
- Research about how to modify a plugin in Grafana.
- Research about how to create a plugin in Grafana.
- Research about how it works a plugin in Grafana.

Plugin Directory Structure

“dist” = distribution.

“src” = source. In this part we do the plugin modification.

C. Plugin directory structure

Folder Name	Definition
dist	Grunt compilation
src	> plugin.json, basic information about the plugin > css, style file > img, image file > partials, HTML documents > js documents
Gruntfile.js	Grunt compilation script
Package.json	Plugin item configuration information and required dependency package information
README.md	Introduction of the plugin's functions

Blendstat Plugin Structure

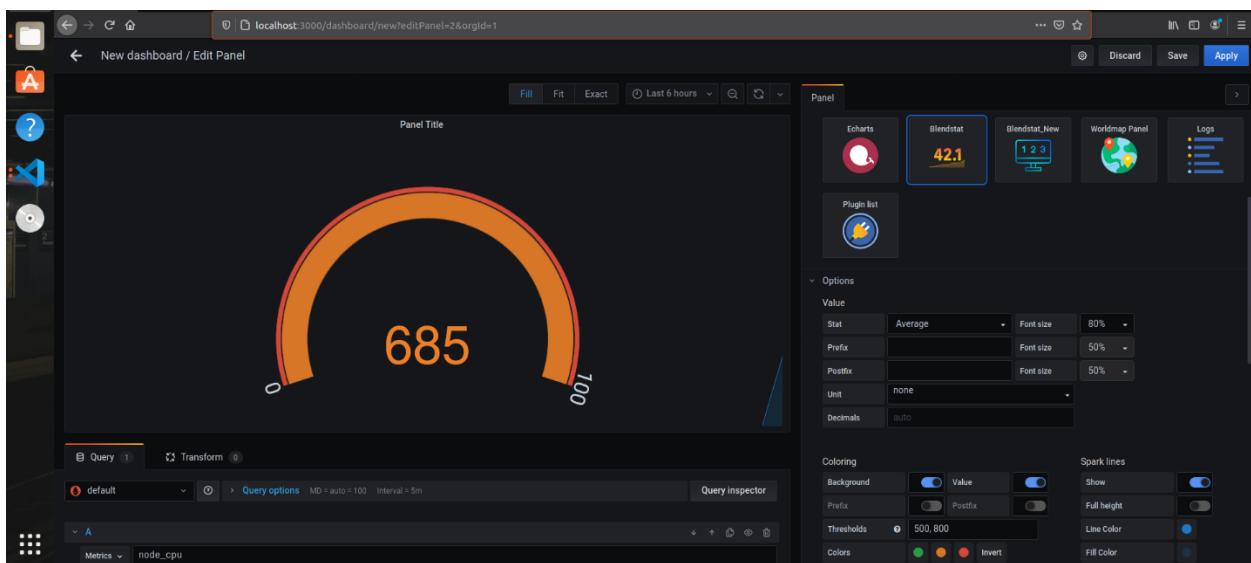
```

oct 6 10:51
editor.html module.ts module.html
blendstat-grafana > src > module.ts > BlendTimeCtrl > link > addGauge > gaugeOptions > centralLabel
12 // furnished to do so, subject to the following conditions:
13
14 // The above copyright notice and this permission notice shall be included in all
15 // copies or substantial portions of the Software.
16
17 // THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
18 // IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
19 // FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
20 // AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
21 // LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
22 // OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
23 // SOFTWARE.
24
25 import "jquery.flot";
26 import "./lib/flot/jquery.flot.gauge";
27 import "jquery.flot.time";
28 import "jquery.flot.crosshair";
29
30 import _ from "lodash";
31 import $ from "jquery";
32 // import 'vendor/flot/jquery.flot';
33 // import 'vendor/flot/jquery.flot.gauge';
34 // import 'app/features/dashboard/panellinks/link_srv';
35
36 import kbn from "app/core/utils/kbn";
37 import config from "app/core/config";
38 import TimeSeries from "app/core/time_series";
39 import { MetricsPanelCtrl } from "app/plugins/sdk";
40 import { GaugeChart } from "./lib/flot/gauge-chart";
41
42 const BASE_FONT_SIZE = 38;
43
44 class BlendTimeCtrl extends MetricsPanelCtrl {
    ...
}

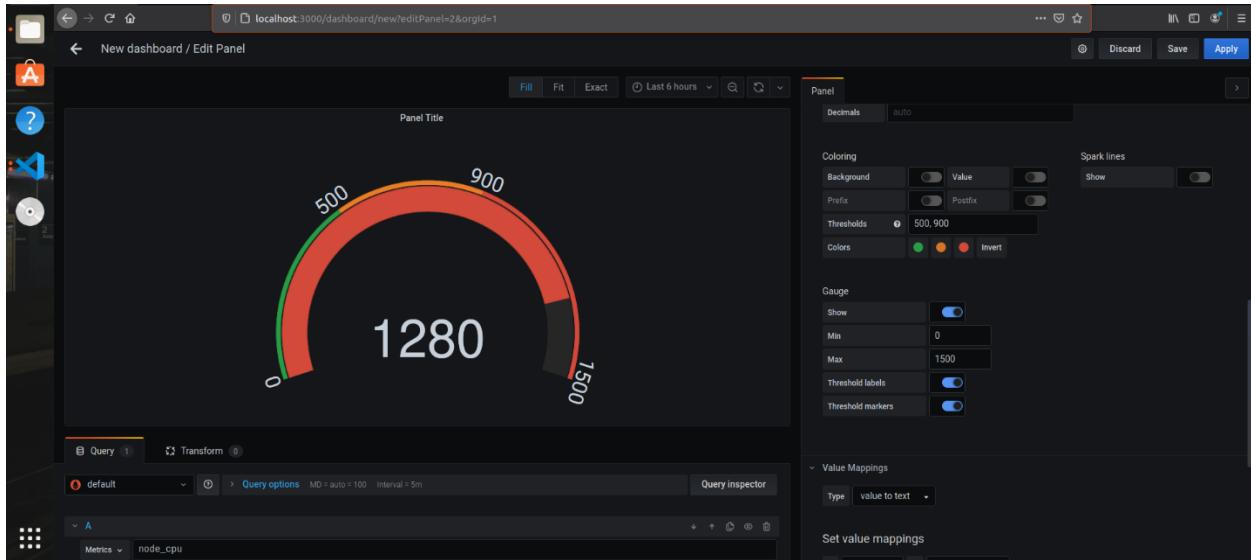
```

Compile Hero: Off Ln 702, Col 37 Spaces: 2 UTF-8 LF TypeScript 4.0.2 Prettier

This is how it looks like right now.



This is the part of the Gauge.



As you can see it hasn't an indicator in the gauge and the colors aren't the best also contains a prefix and postfix section but the problem is that all the information can't fit in the middle of the gauge.



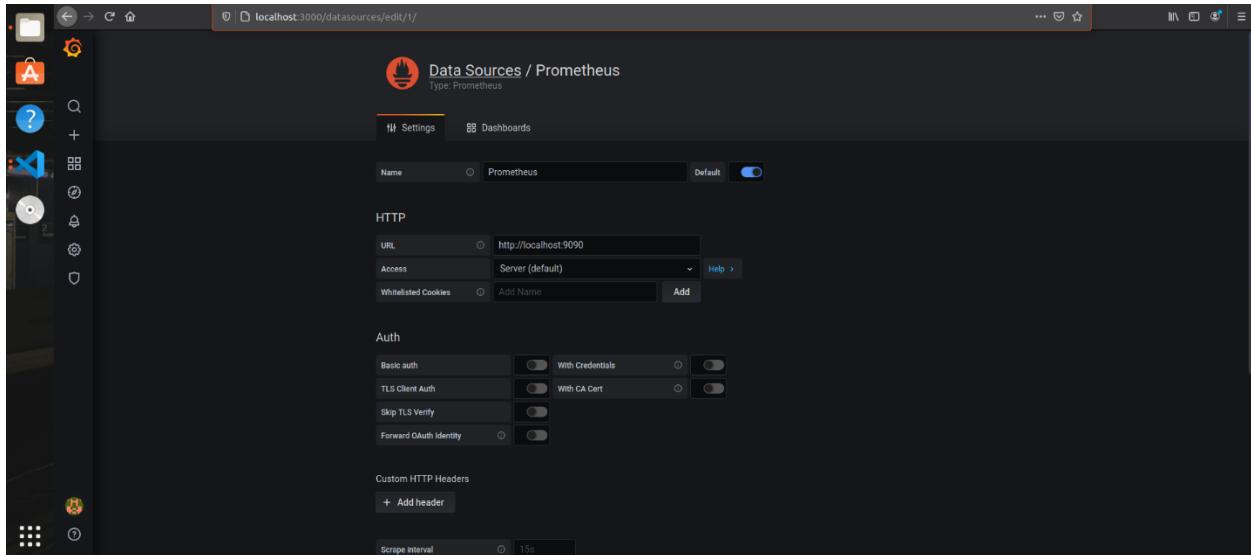
So, we need to fix that and adjust the colors and add an indicator for the value.

First, we need to make it work.

For that I use Prometheus with Node exporter as a Datasource so I installed Prometheus. So with Prometheus you can collect data and in this case I use Node exporter to collect my laptop data. And in that way with Grafana I can explore the data of my machine with Prometheus.

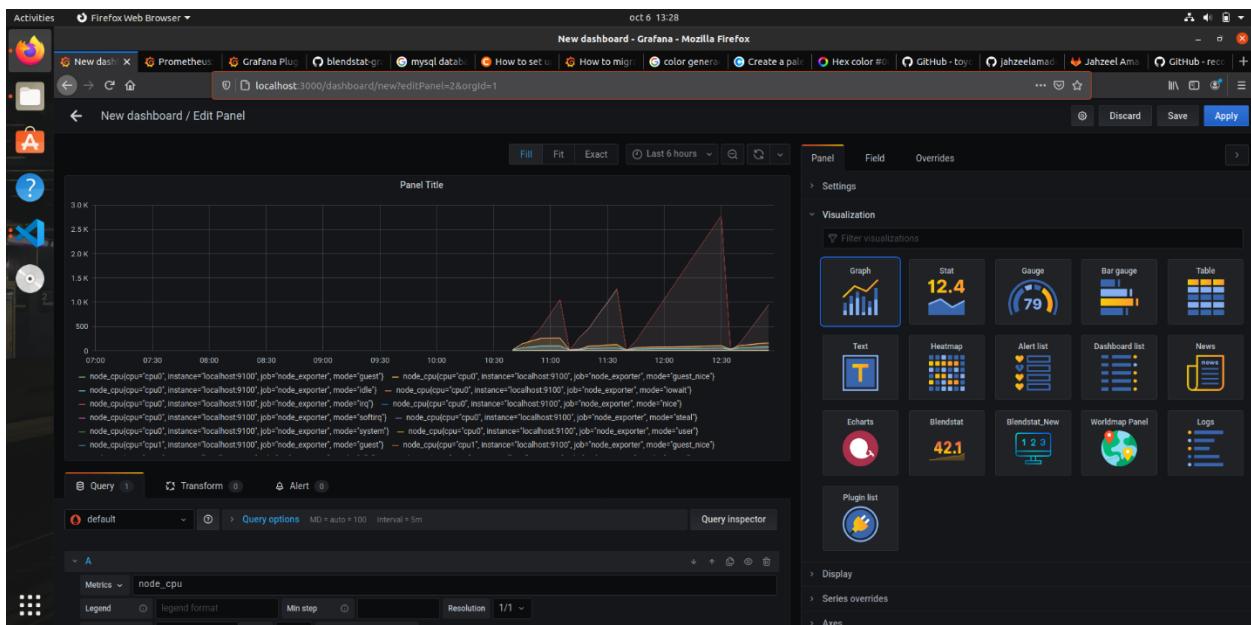
Do the Prometheus installation and add the Datasource in Grafana.

As you can see once you installed Prometheus the only thing that you need to do when you are adding the Prometheus as Datasource is adding the url where Prometheus is running and that's it.



The screenshot shows the 'Data Sources / Prometheus' configuration page in Grafana. The 'Name' field is set to 'Prometheus'. The 'URL' field contains 'http://localhost:9090'. The 'Scrape interval' is set to '15s'. The 'Auth' section is configured with 'Basic auth' and 'With Credentials' turned on. Other options like 'TLS Client Auth' and 'Skip TLS Verify' are turned off. The 'HTTP' section also includes fields for 'Access' (set to 'Server (default)') and 'Whitelisted Cookies'.

Now verify that is actually working.



The screenshot shows the 'New dashboard / Edit Panel' in Grafana. A line chart titled 'Panel Title' displays the 'node_cpu' metric over the last 6 hours. The Y-axis ranges from 0 to 3,000. The chart shows several sharp peaks, notably around 11:00 and 11:30. The legend lists various node exporter metrics for CPU usage across different modes (idle, guest, nice, steal, system, user).

The next step is installed the plugin and for that you need to go to the official page of Grafana plugins, search for the plugin and follow the instructions.

The installation its simple, you just need to copy and paste one command in the Grafana plugin directory.

Installing on a local Grafana:
For local instances, plugins are installed and updated via a simple CLI command.
Plugins are not updated automatically, however you will be notified when updates are available right within your Grafana.

① Install the Panel
Use the grafana-cli tool to install Blendstat from the commandline:

```
grafana-cli plugins install farski-blendstat-panel
```

The plugin will be installed into your grafana plugins directory; the default is /var/lib/grafana/plugins. More information on the cli tool.

Note: Grafana 3.0 or greater is required to install and use plugins. Download Grafana latest.

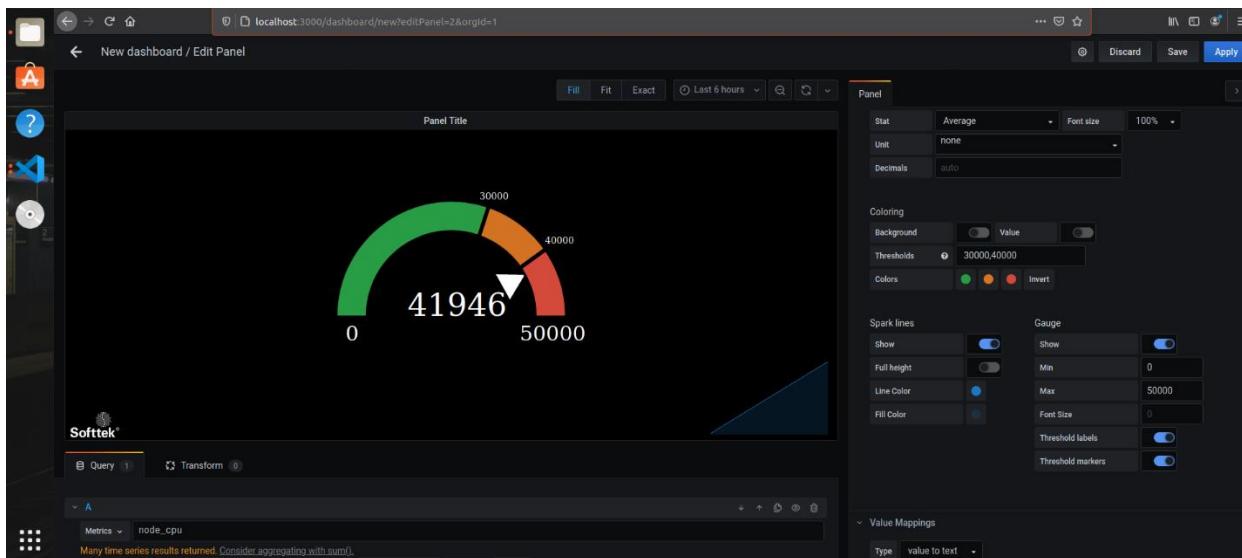
Alternatively, you can manually download the .zip file and unpack it into your grafana plugins directory.

② Add the Panel to a Dashboard
Installed panels are available immediately in the **Dashboards** section in your Grafana main menu, and can be added like any other core panel in Grafana.

To see a list of installed panels, click the Plugins item in the main menu. Both core panels and installed panels will appear.

And with that you installed the plugin and you can see in the previous pages how this plugin works.

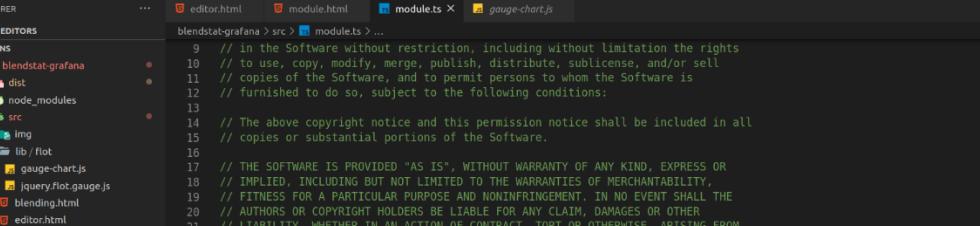
So now this is the final result after all the changes.



Plugin Source Code

If you want to modify this plugin, I did the modification in the next files:

- module.ts
 - module.html
 - editor.html
 - plugin.json
 - gauge-chart.js library



```
9 // in the Software without restriction, including without limitation the rights
10 // to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
11 // copies of the Software, and to permit persons to whom the Software is
12 // furnished to do so, subject to the following conditions:
13
14 // The above copyright notice and this permission notice shall be included in all
15 // copies or substantial portions of the Software.
16
17 // THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
18 // IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
19 // FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
20 // AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
21 // LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
22 // OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
23 // SOFTWARE.
24
25 import "jquery.flot";
26 import "./lib/flot/jquery.flot.gauge";
27 import "jquery.flot.time";
28 import "jquery.flot.crosshair";
29
30 import _ from "lodash";
31 import $ from "jquery";
32 // import 'vendor/flot/jquery.flot';
33 // import 'vendor/flot/jquery.flot.gauge';
34 // import 'app/features/dashboard/panellinks/link_srv';
35
36 import kbn from "app/core/utils/kbn";
37 import config from "app/core/config";
38 import TimeSeries from "app/core/time_series2";
39 import { MetricsPanelCtrl } from "app/plugins/sdk";
40 import * as GaugeChart from "./lib/flot/gauge-chart";
41
42 export BASE_FONT_SIZE = 28;
```

Code changes.

After I made changes in the original code with the original gauge library, I did changes but I didn't solve the problems and I realized that the current library has limitations so for that I decided to use and import a new library. This new library is the gauge-chart.js from this repo <https://github.com/recogizer/gauge-chart> so imported the library.

Now the first change that I made was in the plugin.json so in that file I changed the name of the plugin “Blendstat_New” and imported the icon of the plugin.

```

{
  "type": "panel",
  "name": "Blendstat_New",
  "id": "farski-blendstat-panel_new",
  "info": {
    "description": "Blendstat Panel for Grafana",
    "author": {
      "name": "Christopher Kalafarski",
      "url": "https://github.com/farski/blendstat-grafana"
    },
    "keywords": ["blendstat", "grafana", "panel"],
    "logos": {
      "small": "img/numbers(1).svg",
      "large": "img/numbers(1).svg"
    },
    "version": "1.0.3",
    "updated": "2020-9-21"
  },
  "dependencies": {
    "grafanaVersion": "5.0.0",
    "plugins": []
  }
}

```

To change the icon, first you need to add the icon/image in the “img” folder and then add it in the “logos” section.

The result of this changes.



Then the next thing is imported the library and for that I did a git clone in the blendstat plugin directory and ran the next command “npm install gauge-chart”. And now I have the new library added it in the node_modules directory of the plugin.

At this point I realized that somehow I can't make changes of the gauge with the plugin inside the node_modules and I didn't know why but even when I compiled the code I can't made changes so after a research I found a solution for this problem and in that way I can made changes of the gauge.

To solve this problem, I download the gauge library source out of the plugins Grafana directory.

This is de code structure and here I did the gauge changes, to be more specific in the gauge.ts file and as you can see the file names are intuitive so if you want to change another thing you can do it in the properly file.

The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** Activities ➔ Visual Studio Code ➔ gauge.ts - gauge-chart-master - Visual Studio Code oct 8 11:25
- File Explorer (Left):** Shows the project structure under "GAUGE-CHART-MASTER". The "src" folder contains "gauge" which includes "gauge-interface.ts", "gauge.css", "gauge.specs", and "gauge.ts". Other files like "index.d.ts", "logger.ts", etc., are also listed.
- Editor Area (Center):** Displays the "gauge.ts" file content. The code handles arc colors and needle value modifiers.
- Right Panel:** Shows the "Problems" and "Output" tabs, along with other status indicators.

```
gauge.ts x bundle.js
src > gauge > gauge.ts > ↗ arcColorsModifier
19
20     if (arcDelimiters.length > arcColors.length - 1) {
21         let colorDiff = arcDelimiters.length - arcColors.length + 1
22         for (let i = 0; i < colorDiff; i++) {
23             arcColors.push(schemePaired[i % schemePaired.length])
24         }
25     } else if (arcDelimiters.length < arcColors.length - 1) {
26         arcColors = arcColors.slice(0, arcDelimiters.length + 1)
27     }
28
29     return arcColors
30
31
32 /**
33 * Function that checks whether value that needle points at is between 0 and 100.
34 * If it is less than 0 or larger than 100, value is equated to 0 and 100 respectively.
35 * @param needleValue - value at which needle points.
36 * @returns modified needleValue.
37 */
38 export function needleValueModifier(needleValue: number) {
39     return needleValue
40 }
41
42 /**
43 * Function that converts percentage into radians.
44 * @param perc - percentage.
45 * @returns value in radians.
46 */
47 export function perc2RadWithShift(perc: number) {
48     return (perc / 100 - 0.5) * Math.PI
49 }
50
51 /**
52 * Function for drawing gauge arc.
```

Once I did the changes that I wanted, I did the build of the library to compile the changes so for that I checked the package.json

Here says what commands run to do the build, the initialization and the dev.

So I did the build.

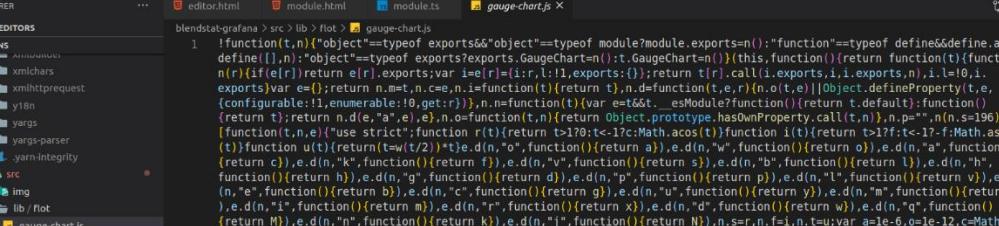
The screenshot shows the Visual Studio Code interface with the title "package.json - gauge-chart-master - Visual Studio Code". The code editor displays the contents of the package.json file, which includes details about the library, build configurations, and development scripts. The Explorer sidebar on the left shows the project structure, including files like gauge.ts, package.json, and bundle.js.

```
package.json - gauge-chart-master - Visual Studio Code

{
  "name": "gauge-chart",
  "version": "0.5.3",
  "description": "**A library for creating nice and flexible gauge charts.**",
  "main": "dist/bundle.js",
  "types": "dist/index.d.ts",
  "scripts": {
    "test": "karma start",
    "watch:test": "npm run test -- --auto-watch --no-single-run",
    "dev": "webpack-dev-server --config webpack/dev.config.js --hot",
    "build": "ENV=production webpack --config webpack/prod.config.js",
    "build:prod_warn": "ENV=prod_warn webpack --config webpack/prod.config.js",
    "typedoc": "typedoc --typedoc ./docs --exclude '**/*.spec.ts' ./src",
    "precommit": "lint-staged",
    "fix-all-ts": "prettier --write src/**/*.{ts,js} && tslint --fix -c ./tslint.json 'src/**/*.{ts,js}'",
    "start": "npm run dev"
  },
  "lint-staged": {
    "*.ts": [
      "prettier --write",
      "tslint --fix -c ./tslint.json 'src/**/*.{ts,js}'",
      "git add"
    ]
  },
  "author": "Maxim Maltsev <maltsevmn@ya.ru>, Alexey Karpov <karpovad@yandex.ru>",
  "license": "MIT",
  "devDependencies": {
    "@types/d3": "^4.13.2",
    "@types/jasmine": "^2.8.16",
    "@types/node": "^7.10.7",
    "@types/source-map": "^0.9.5",
    "@types/webpack": "^2.2.16",
    "awesome-typescript-loader": "^3.5.0"
  }
}
```

Once I ran the build, it generated the dist folder and inside the dist folder there is another folder where is all the gauge files but the one file that I need is the “bundle.js” file. This file contains all the code compiled.

What I did was copy this bundle.js file and pasted it in the lib folder of the Grafana plugin and changed the name for “gauge-chart.js”.



The screenshot shows the Visual Studio Code interface with the following details:

- Activity Bar:** Shows "Activities" and "Visual Studio Code".
- Header:** "oct 18 13:06" and "gauge-chart.js - plugins - Visual Studio Code".
- File Explorer:** Shows the project structure:
 - OPEN EDITORS:** "editor.html", "module.html", "module.ts", "gauge-chart.js" (highlighted).
 - PLUGINS:** "all-in-one", "xmlchar", "xmlhttprequest", "y18n", "yargs", "yargs-parser", "yaml-integrity".
 - src:** "img", "lib/flat".
 - lib/flat:** "gauge-chart.js", "jquery.flat-google.js", "blending.html", "editor.html", "mappings.html", "module.html", "module.ts", "plugin.json", "README.Emd", "styles.css", ".gitignore", "Gruntfile.js", "LICENSE", "package-lock.json", "package.json", "README.md".
- Editor:** The "gauge-chart.js" file is open, displaying a large block of JavaScript code. The code is annotated with numerous red highlights and underlines, primarily on the first few lines and throughout the "function(t,n)" block, indicating specific areas of interest or modification.

In that way I can make changes of the gauge but the only detail is that I need to do this procedure every time I need a change in the gauge.

Once I imported the new library I started with the changes.

First, I want to change the options that the plugin offers (right section).

I think the “Prefix” and “Postfix” aren’t work for me so I decide to remove them.



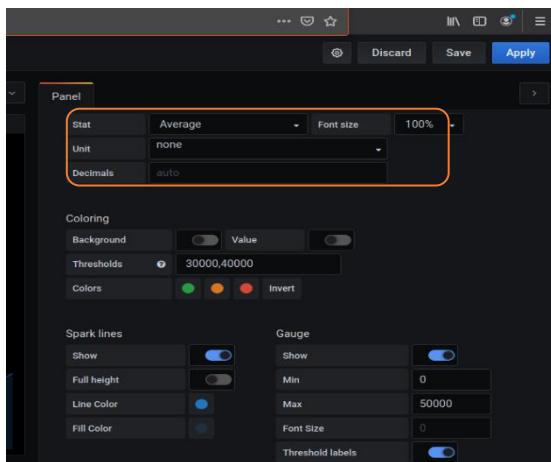
To remove these options, I need to modify the “editor.html” file.

The only thing that I’m doing is commenting the div where the options are and that’s it. And all the options can be removing it from this file.

A screenshot of Visual Studio Code showing the "editor.html" file open in the editor. The code contains several sections of HTML and Angular template code. A specific section of code related to "Prefix" and "Postfix" is highlighted with a comment symbol (//) added before it, indicating it has been commented out.

```
//<div class="gf-form">
//  <label class="gf-form-label width-6">Font size</label>
//  <input type="text" class="gf-form-input width-12" ng-model="ctrl.panel.prefix" ng-change="ctrl.render()" ng-model-options="ctrl.panel.ngModelOptions">
//  <div class="gf-form-select-wrapper">
//    <select class="gf-form-input" ng-model="ctrl.panel.valueFontSize" ng-options="f for f in ctrl.fontSizes" ng-change="ctrl.render()">
//      <option value="12">12</option>
//      <option value="14">14</option>
//      <option value="16">16</option>
//      <option value="18">18</option>
//      <option value="20">20</option>
//      <option value="22">22</option>
//      <option value="24">24</option>
//      <option value="26">26</option>
//      <option value="28">28</option>
//      <option value="30">30</option>
//      <option value="32">32</option>
//      <option value="34">34</option>
//      <option value="36">36</option>
//      <option value="38">38</option>
//      <option value="40">40</option>
//      <option value="42">42</option>
//      <option value="44">44</option>
//      <option value="46">46</option>
//      <option value="48">48</option>
//      <option value="50">50</option>
//      <option value="52">52</option>
//      <option value="54">54</option>
//      <option value="56">56</option>
//      <option value="58">58</option>
//      <option value="60">60</option>
//      <option value="62">62</option>
//      <option value="64">64</option>
//      <option value="66">66</option>
//      <option value="68">68</option>
//      <option value="70">70</option>
//    </select>
//  </div>
//</div>
```

With that I remove the options.



The next thing to do is in the “module.ts” file so first I imported the gauge-chart library.

```
oct 6 10:51
module.ts - Visual Studio Code

editor.html module.ts module.html
blendstat-grafana > src > module.ts > BlendStatCtrl > link > addGauge > gaugeOptions > centralLabel

12 // furnished to do so, subject to the following conditions:
13
14 // The above copyright notice and this permission notice shall be included in all
15 // copies or substantial portions of the Software.
16
17 // THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
18 // IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
19 // FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
20 // AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
21 // LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
22 // OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
23 // SOFTWARE.
24
25 import "jquery.flot";
26 import "./lib/flot/jquery.flot.gauge";
27 import "jquery.flot.time";
28 import "jquery.flot.crosshair";
29
30 import _ from "lodash";
31 import $ from "jquery";
32 // import 'vendor/flot/jquery.flot';
33 // import 'vendor/flot/jquery.flot.gauge';
34 // import 'app/features/dashboard/panellinks/link_srv';
35
36 import kbn from "app/core/utils/kbn";
37 import config from "app/core/config";
38 import TimeSeries from "app/core/time_series";
39 import { MetricsPanelCtrl } from "app/plugins/sdk";
40 import * as GaugeChart from "./lib/flot/gauge-chart";
41
42 const BASE_FONT_SIZE = 38;
43
44 class BlendStatCtrl extends MetricsPanelCtrl {
45   static templateFile = "module.html".
```

Now in this file I'm going to do all the plugin changes.

The next thing to do imported the Softtek logo and make it responsive and change the background color.

```

nov 5 13:06
File Edit Selection View Go Run Terminal Help
EXPLORER OPEN EDITORS
PLUGINS CINEPOLISLOGO
ts types.ts ts SimplePanel.tsx ts SimpleEditor.tsx module.html module.ts
pluginblendstat > src > ts modules > BlendStatCtrl > link > render > image.onmouseleave
804 //Create the element
805 let image = new Image(75);
806 // Import the image
807 image.src =
808 "https://www.softtek.com/images/content/design2015/LogoCompleto-Website-20.png";
809 // To fix the logo position in the panel
810 image.style.opacity = "0.7";
811 image.style.width = "7%";
812 image.style.position = "absolute";
813 image.style.display = "block";
814 image.style.left = "-1%";
815 image.style.bottom = "2%";
816
817 // Logo cinepolis.
818 image.id = "Logo";
819 image.onmouseover = () => {
820   image.src =
821     "https://static.cinepolis.com/img/lg.cinepolis-new.png";
822   image.style.opacity = "0.7";
823   image.style.width = "7%";
824   image.style.position = "absolute";
825   image.style.display = "block";
826   image.style.left = "1%";
827   image.style.bottom = "2%";
828 };
829 image.onmouseleave = () => [
830   image.src =
831     "https://www.softtek.com/images/content/design2015/LogoCompleto-Website-20.png";
832   image.style.opacity = "0.7";
833   image.style.width = "7%";
834   image.style.position = "absolute";
835   image.style.display = "block";
836   image.style.left = "-1%";
837   image.style.bottom = "2%";
838 ];
839
840 // Here we create the logo in the panel
841 const panelContents = document.getElementsByClassName("singlestat-panel");
842 for (let i = 0; i < panelContents.length; i++) {
843   if (document.getElementById("logo")) {
844     document
845       .getElementById("logo")
846       .parentElement.removeChild(document.getElementById("logo"));
847   }
848   // Here I change the background color and add the logo.
849   panelContents.item(i).style.backgroundColor = "#000";
850   panelContents.item(i).appendChild(image);
851 }

```

nov 5 13:26

```

File Edit Selection View Go Run Terminal Help
EXPLORER OPEN EDITORS
PLUGINS CINEPOLISLOGO
ts types.ts ts SimplePanel.tsx ts SimpleEditor.tsx module.html module.ts
pluginblendstat > src > ts modules > BlendStatCtrl > link > render > image.onmouseleave
828 );
829 image.onmouseleave = () => [
830   image.src =
831     "https://www.softtek.com/images/content/design2015/LogoCompleto-Website-20.png";
832   image.style.opacity = "0.7";
833   image.style.width = "7%";
834   image.style.position = "absolute";
835   image.style.display = "block"];
836   image.style.left = "1%";
837   image.style.bottom = "2%";
838 ];
839
840 // Here we create the logo in the panel
841 const panelContents = document.getElementsByClassName("singlestat-panel");
842 for (let i = 0; i < panelContents.length; i++) {
843   if (document.getElementById("logo")) {
844     document
845       .getElementById("logo")
846       .parentElement.removeChild(document.getElementById("logo"));
847   }
848   // Here I change the background color and add the logo.
849   panelContents.item(i).style.backgroundColor = "#000";
850   panelContents.item(i).appendChild(image);
851 }

```

The next thing to do is add the new gauge and for that I go to the existing function “addGauge()” and add the new gauge.

```

oct 21 11:06
File Edit Selection View Go Run Terminal Help
editor.html module.html module.ts
683 //Create the new gauge
684 let element = document.createElement('div');
685 element.id = '#gaugeArea';
686 // Element inside which you want to see the chart
687 let needleValue = parseFloat(getValueText());
688 // Properties of the gauge
689 let gaugeOptions = {
690     // needle options
691     hasNeedle: true,
692     outerNeedle: false,
693     needleColor: "#fff",
694     needleStartValue: (needleValue / panel.gauge maxValue) * 100,
695     needleUpdateSpeed: 100,
696     needleValue: (needleValue / panel.gauge maxValue) * 100,
697     // arc options
698     arcColors: [...panel.colors], // Here we match the threshold colors; green, red and orange.
699     arcDelimiters: [...panel.thresholds.trim().split(',').map(parseFloat).filter(item => item > 0)].map((threshold => {
700         console.log(threshold);
701         threshold = parseFloat(threshold);
702         let delimiter = (threshold / panel.gauge maxValue) * 100;
703         return delimiter;
704     })),
705     arcPadding: 6,
706     arcPaddingColor: "#000",
707     arcLabels: data.thresholds, //the limit is 100
708     arcLabelFontSize: true,
709     arcOverEffect: true,
710     arcOverEffect: false,
711     // label options
712     rangeLabel: [panel.gauge minValue + '', panel.gauge maxValue + ''],
713     centralLabel: data.valueFormated + '', //If you coment this line, appears the needle instead of the numbers value.
714     rangeLabelFontSize: panel.gauge.fontSize,
715     labelsFont: 'Consolas',
716 }
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736

```

Compile Hero: Off Ln 710, Col 32 Spaces: 2 UTF-8 LF TypeScript Go Live 4.0.3 Prettier

And also, I add the possibility to add text I want bellow the gauge and for that I add the next part.

```

oct 21 12:44
File Edit Selection View Go Run Terminal Help
editor.html module.html module.ts
711 // Label options
712 rangeLabel: [panel.gauge minValue + '', panel.gauge maxValue + ''], // above
713 centralLabel: data.valueFormated + '', //If you coment this line, appears the needle instead of the numbers value.
714 rangeLabelFontSize: panel.gauge.fontSize,
715 labelsFont: 'Consolas',
716 color: 'white',
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736

```

In this part I delete all about the other gauge and I only left the code of the new gauge.

And that's all the code that I changed and added in this file.

To make work the code in the previous file I need to make changes in other files.

Now I'm going to do the change in the module.html file, in this file I add the id of my new gauge and I add the style of the gauge.

And that's all the things that I do in this file.

```
oct 21 12:21  
module.html - plugins - Visual Studio Code  
File Edit Selection View Go Run Terminal Help  
EXPLORER File editor.html module.html module.ts  
OPEN EDITORS  
blendsstat-grafana  
PLUGINS  
blendstat-grafana  
dist node_modules src  
img lib blending.html editor.html mappings.html module.html module.ts 1,M  
plugin.json README.md styles.css .gitignore Gruntfile.js LICENSE package-lock.json package.json README.md tscommand-8abbb5f8.tmp.txt tscommand-beba29d8.tmp.txt tslint.json yarn.lock  
farski-blendstat-panel grafana-echarts OUTLINE TIMELINE  
master < Live Share  
Compile Hero: Off Ln 27, Col 1 Spaces: 4 UTF-8 LF HTML Prettier
```

```
10 // Copyright 2016-2017 the original author or authors.  
11 // The copyright owners do not represent or warrant that  
12 // ownership of this software has been transferred to other  
13 // persons, and agree to the following conditions:  
14  
15 E SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR  
16 PLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,  
17 FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE  
18 THORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER  
19 ABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,  
20 T OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE  
21 FTWARE. -->  
22  
23 | -- In this part I add the gauge id and add the styles of the gauge in the main panel -->  
24 | iv class="singlestat-panel" id="gaugeArea" style="display: flex; justify-content: center; align-items: center; overflow: hidden;">  
25 |  
26 | div>  
27 |
```

Now I have to do some changes in the editor.html file.

Previously I did changes in this file, I removed some things from the options panel and now I need to add new options.

As you can see I added the options to add the text below the gauge and the option to edit the font size of the gauge text and the font size of the text below the gauge.

The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** Activities > Visual Studio Code - editor.html - plugins - Visual Studio Code
- Status Bar:** Oct 21 12:34
- File Tree (Left):** Shows the project structure with files like `editor.html`, `module.html`, `module.ts`, `blendingstat-grafana`, `blending.html`, `mappings.html`, `module.html`, `module.ts`, `plugin.json`, `README.md`, `styles.css`, `.gitignore`, `Gruntfile.js`, `LICENSE`, `package-lock.json`, `package.json`, `README.md`, `tscommand-8a1bb5f8.txt`, `tscommand-beba29de.txt`, `tslint.json`, and `yarn.lock`.
- Editor Area (Main):** The `editor.html` file is open, displaying code related to a gauge input field. The code includes labels for "Min" and "Max" values, and a note about adding forms for font size and text input.

This is the final result.

