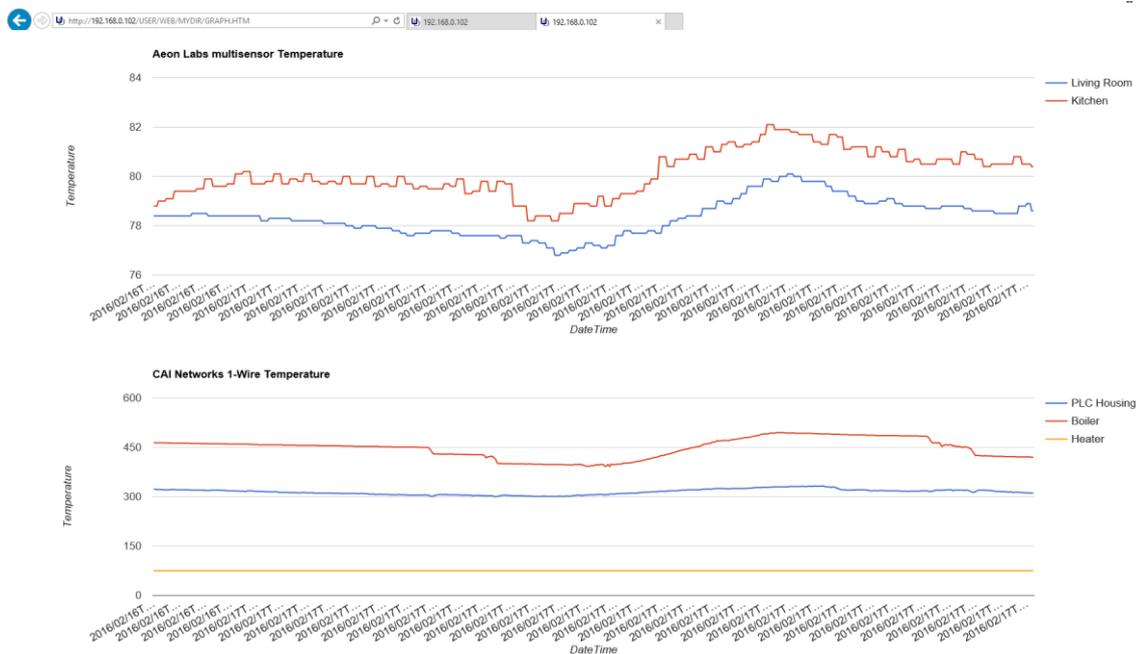


1st of all, I'd like to thank 50-50 for the smart solution by using the Google Charts to locally show data stored on ISY by using the @ to create a local file and Teken for creating the SEG instructions for data logging which makes me purchase the ISY Network Module and pursue a way to won't pay SEG monthly.

This step by step will not only guide you through the entire process of creating a local file to log data on ISY but also explain some characteristics of data storage, strings (as far I know for now) and specially the super easy (for PHDs) way to create graphs using Google Charts.

By following this instruction you'll be able to create something like the picture below that you can access from anywhere.

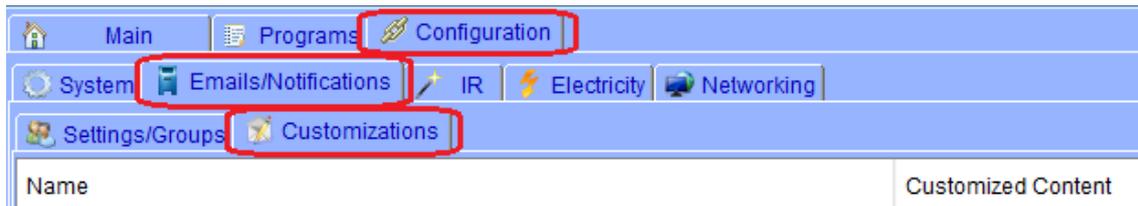


The above image shows two separate graphics, one for my Aeon Labs multisensor, and one for my CAI Networks 1-Wire PLC.

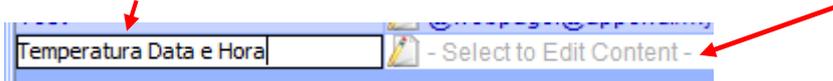
In order to make it happen, you **must have the ISY Network Module** and understand the minimum about Variables and Networking Substitution Variables (I'll try to make it easy).

Step1: Creating a custom email

Go to "Configuration", "Emails/Notifications", "Customizations" and ADD a new email;



Give it a NAME and **don't forget this name**, then double click on "Select to Edit Content"



At this point, I'd like to show exactly what you're about to do and why, but if you're in a hurry go ahead and copy/paste the string and good luck..

The **Subject** field need to be created with the @webpage and @append command plus the directory and filename to be created.

@webpage is used to create a file, the file named at the end of the string

@append is to ADD data to the file in case this file already exist

/MyDir/TEST.CSV is the path to be created by @webpage

The entire Subject string here is: **@webpage:@append:/myDir/TEST.CSV**

The **Body** field is where you have to tell what kind of data the ISY need to put into the file. You can find a list of EMail and Networking Substitution Variables at:

http://wiki.universal-devices.com/index.php?title=ISY-994i_Series:EMail_and_Networking_Substitution_Variables

Unfortunately, it's not that sensitive to understand how to use, especially for noobs like me, so that's why I'll explain here the \$ I've used.

`#{sys.date}` will add the date to your file in this format: 2016/02/16 (YYYY/MM/DD);

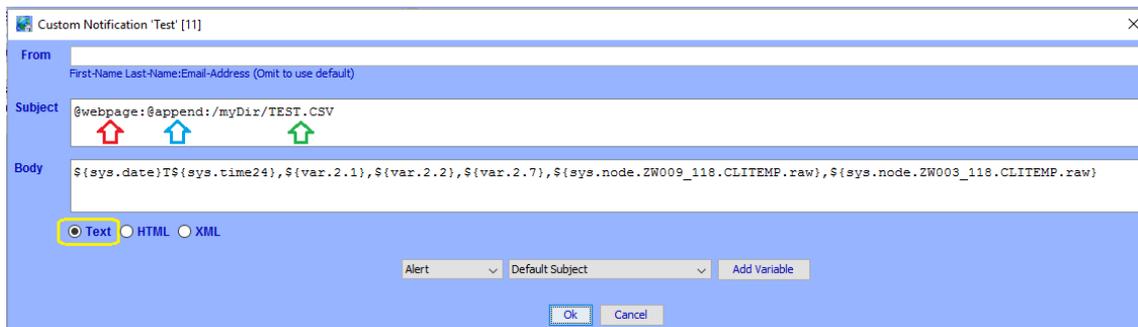
The 'T' here I don't really understood why is there, but 50-50 may explain :P;

`#{sys.time}` will add the time stamping to your file in this format: 21:28:42 (HH:MM:SS);

The comma will give to the CSV the row separation, so the 1st row will have dataTtime;

`#{var.2.1}` will pick from my Variable state number 1 the value and add to the CSV. Note that the if I change the number 2 for 1 it will get the variable from Integer table. Ex: `#{var.1.12}` will add to the CSV file the Integer position 12.

I'll explain again on next page...



REMEMBER TO HIT ENTER ONE TIME ONLY AT THE END, IT WILL ACT AS A SPACE SEPARATOR

What we have here is an example on how to map the Networking Substitution Variables

ID	Name	Init	Value	Last Changed
1	Caixa.PLC.CAI.Networks	0	310	Seg 2016/02/15 11:35:34 PM
2	Temperatura.Boiler	0	512	Seg 2016/02/15 11:37:35 PM
3	Spare1	0	0	Dom 2016/01/31 07:01:58 PM
4	Spare2	0	0	Dom 2016/01/31 07:01:58 PM
5	Spare3	0	0	Dom 2016/01/31 07:01:58 PM
6	Hot.Water.Booster.Pump	0	0	Seg 2016/02/15 11:22:33 PM
7	Boiler.Heater	0	75	Qui 2016/02/04 12:49:36 PM
8	Spare	0	1	Qui 2016/02/04 12:47:35 PM

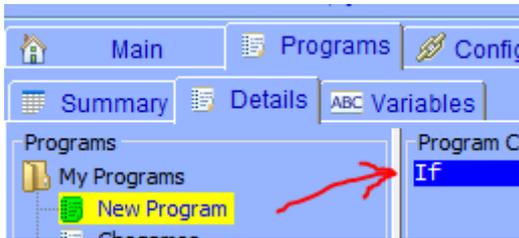
Yeah, this one was easy no? So, try to figure how to send values from devices like Z-Wave Aeon Labs temperature :P

Let's continue, remember it's still the 1st step!!!!

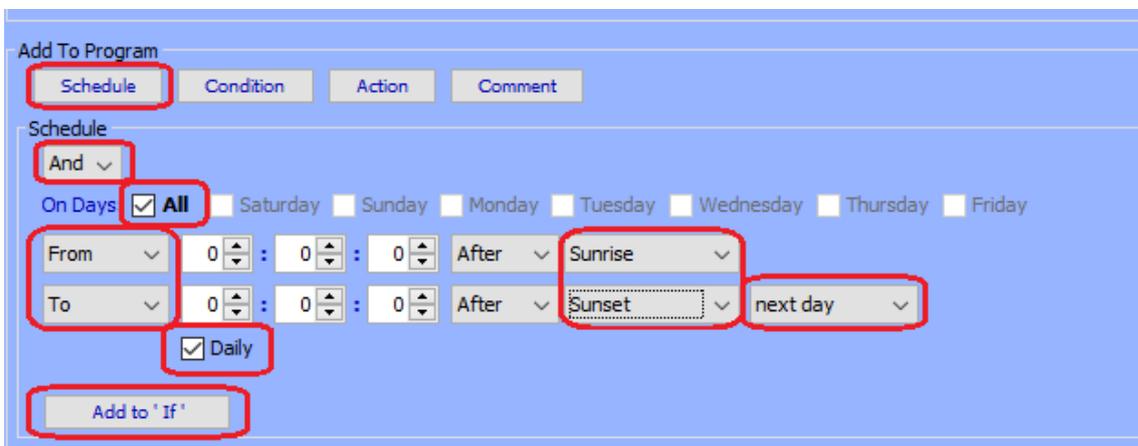
After create the custom email with ALL data you want to export to a CSV file (in fact the extension CSV doesn't really matter for us on this tutorial), you have to SAVE it, so press SAVE!

Step2: Create a routine to call the custom email every other delta time

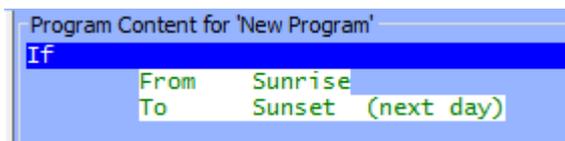
Go to “Program”, “Details” and right click wherever you want on “Programs” tree and add a new program, select “If”...



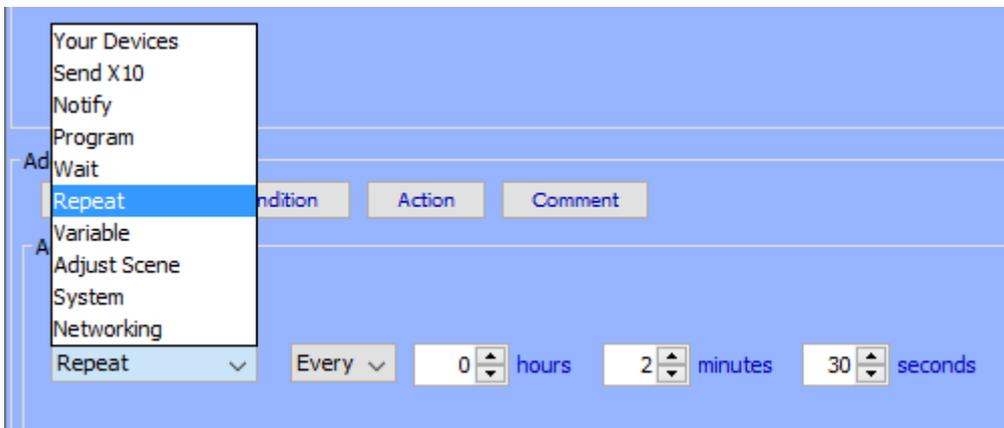
Click on “Schedule”, leave “AND”, decide if you want it to run every day and leave “ALL”, change to “From – To”, modify to “Sunrise and Sunset of next day”, leave “Daily” and click on “Add to If”.



Should look like this...

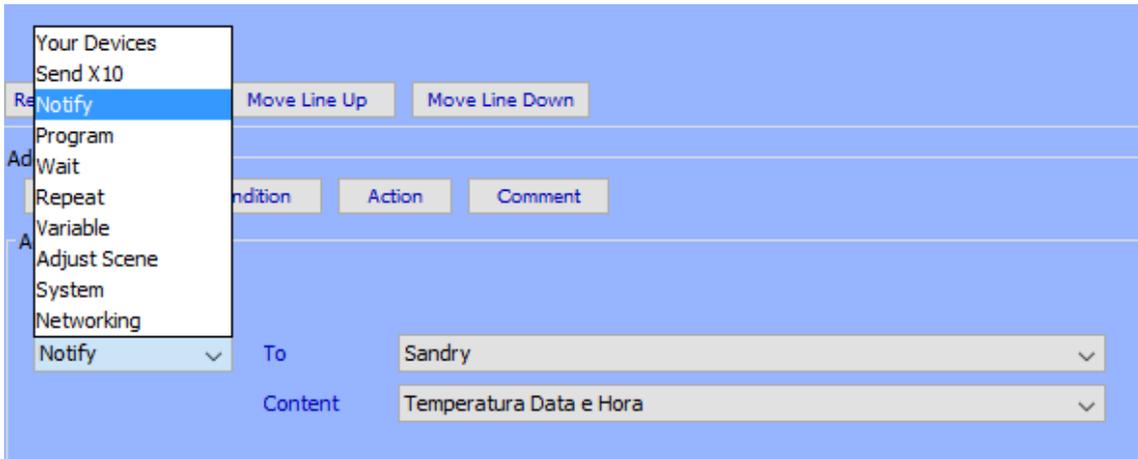


Select “Then”, select “Repeat” and chose the time interval to feed data to your file, here in this example I’m using two and a half minutes (copy/paste from 50-50 idea)



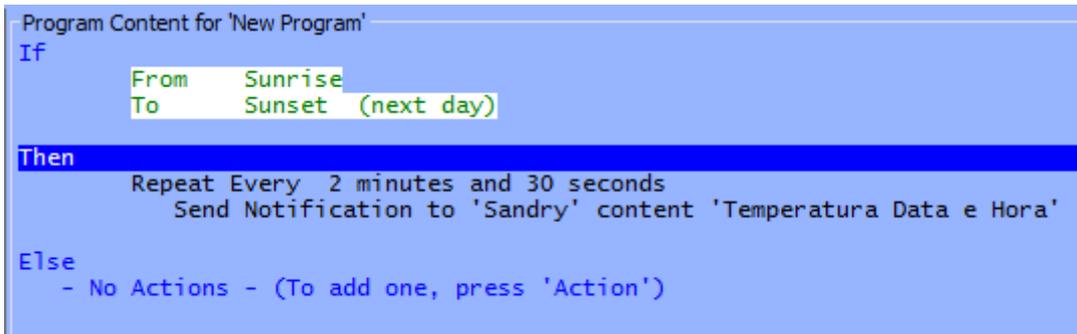
Click on “Add to Then” !!!!

Now select “Notify” then select any recipient on “To” and the “Content” must be the Customized email you created and I asked you to don’t forget the name!



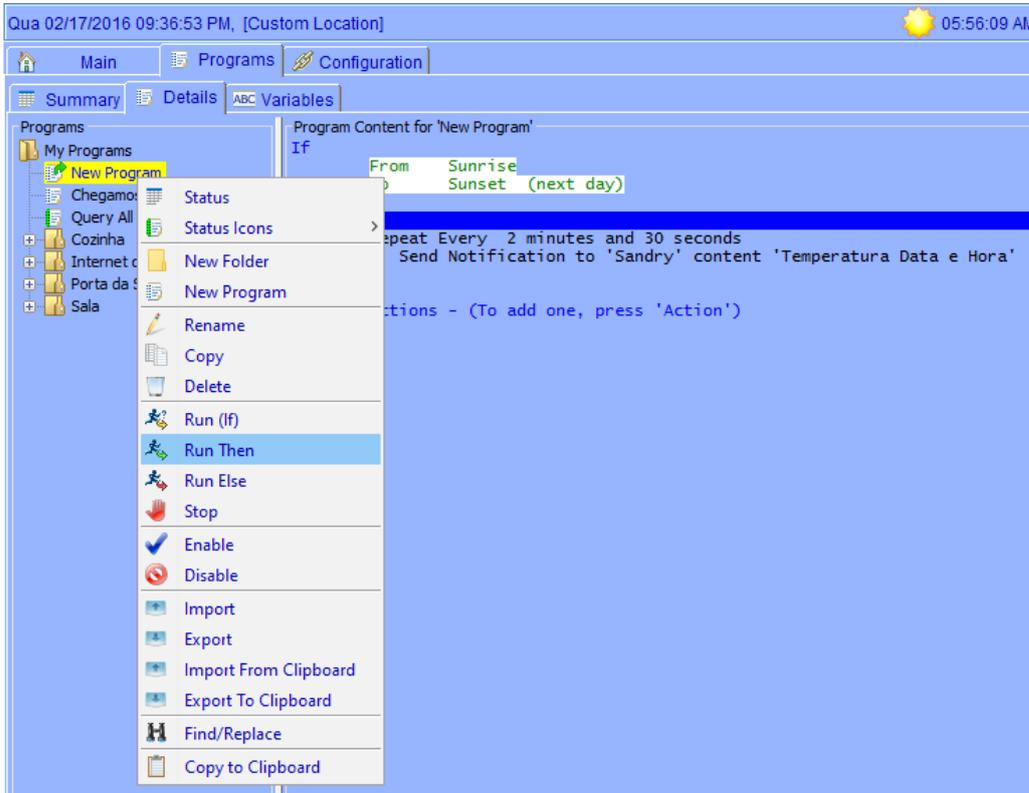
Click on “Add to Then” again, and click on “Save Changes” (lower left corner).

So, your program should look like this..

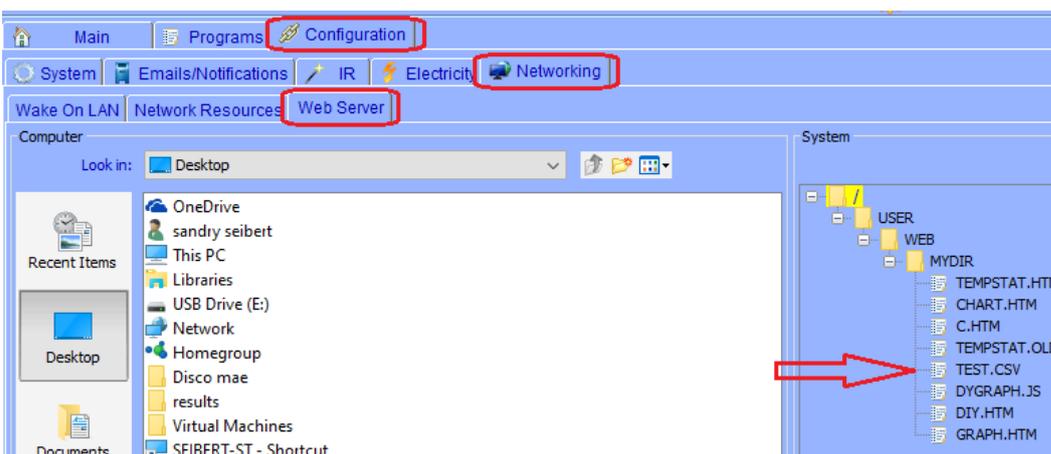


Yeah, I know you’re in a hurry and I wrote too many things and nothing really exciting happened yet, but now you can do a test and see if you are smart or should go back to school.

Right click on the program you just created and click “Run Then”, it will execute the command on demand, so you can do this all the time to fill data to your file... File? Wow, let’s take a look on your file..



By running the 1st time the customized email, it will create the CSV file you give the name, in this tutorial I used TEST.CSV. Go to “Configuration”, “Networking”, “Web Server” and expand the “System” Web Server tree. Is there a TEST.CSV? Congratulations, it’s just the beginning!



You can right click and open it, if you have Excel, this is what you should see:

	A	B
1	2016/02/16T21:28:42,323,465,75,78.40,78.80	
2	2016/02/16T21:31:12,323,464,75,78.40,78.80	
3	2016/02/16T21:33:42,322,464,75,78.40,78.80	
4	2016/02/16T21:36:12,322,464,75,78.40,79.00	
5	2016/02/16T21:38:42,323,464,75,78.40,79.00	

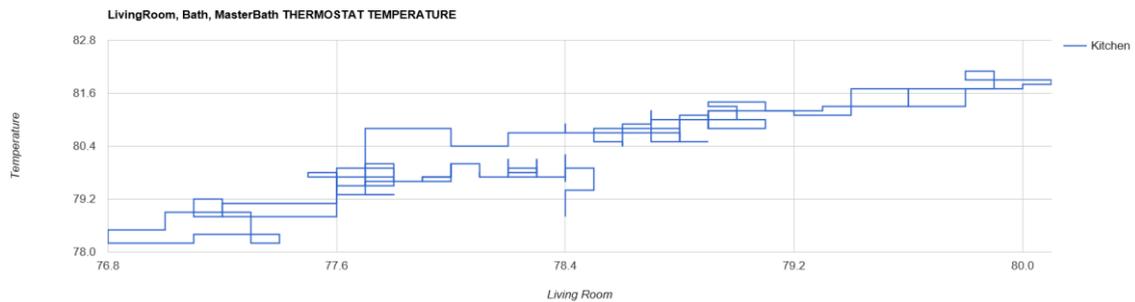
Step3: Add a header to your CSV file

If you opened the CSV file on your Excel, add a line before line1 and create the header names for your table like the example below:



A	
1	DateTime,PLC Housing,Boiler,Heater,Living Room,Kitchen
2	2016/02/16T21:28:42,323,465,75,78.40,78.80

You may have noticed that I created one header for each parameter, I'm not entirely sure what will happen if you don't do this, but probably will mess up your data. Here's an example:



I spent about 2 hours to fix it!

BE AWARE THAT THE COMMA SEPARATE THE ROWS

Let's go to next step.. THE SCRIPT

Step4: Create Copy with some mods a JavaScript gently done by 50-50

When I say Copy, just copy... It's pretty hard to handle JavaScripts... Any missing and you'll need to be an expert to troubleshoot using advanced technique's like Debug Console, etc.

Paste below is my script, there I'll explain some components you may want to play with...

```
<html>
<head>
  <script src="https://www.google.com/jsapi"></script>
  <script src="http://code.jquery.com/jquery-1.10.1.min.js"></script>
  <script src="http://jquery-csv.googlecode.com/git/src/jquery.csv.js"></script>
  <script src="http://momentjs.com/downloads/moment.js"></script>

  <script type="text/javascript">

    google.load("visualization", "1", {packages:["corechart", "controls"]});
    google.setOnLoadCallback(getCSV);

    function handleQueryResponse(response) {

      if (response.isError()) {

        alert('Error in query: ' + response.getMessage() + ' ' + response.getDetailedMessage());

        return;

      }

      var data = response.getDataTable();

      // Numbers are to select which row you want to draw, you may want to plot 2, 4, 1, 5... or like me//
      //This example plot two graphs, so there are two vars named t_view and r_view//

      var t_view = new google.visualization.DataView(data);
      t_view.setColumns([0,1,2,3]); //This numbers... ZERO must be there, it's your TIME STAMP//

      var r_view = new google.visualization.DataView(data);
      r_view.setColumns([0,4,5]);

      //Title for your chart, in this example I used CAI Networks blablabla//

      var t_options = {
        title: "CAI Networks 1-Wire Temperature",

        hAxis: {title: t_view.getColumnLabel(0), minValue: t_view.getColumnRange(0).min, maxValue:
t_view.getColumnRange(0).max},

        vAxis: {title: 'Temperature', minValue: t_view.getColumnRange(1).min, maxValue:
t_view.getColumnRange(1).max}

      };

      //Title for your chart, in this example I used Aeon blablabla//

      var r_options = {
        title: "Aeon Labs multisensor Temperature",

        hAxis: {title: r_view.getColumnLabel(0), minValue: r_view.getColumnRange(0).min, maxValue:
r_view.getColumnRange(0).max},

        vAxis: {title: 'Temperature', minValue: r_view.getColumnRange(1).min, maxValue:
r_view.getColumnRange(1).max}

      };

    }

  </script>
</head>
</html>
```

```

var r_chart = new google.visualization.LineChart(document.getElementById('CAI'));
r_chart.draw(r_view, r_options);
var t_chart = new google.visualization.LineChart(document.getElementById('AEON'));
t_chart.draw(t_view, t_options);

}

//Function to read data from your CSV file//

function getCSV() {

    var csvURL = "TEST.CSV"; //Modify HERE the name of the file you created, no need path if located on same
    place you place this script file//

    //Here is important to correctly size your chart, the 1st option is a String because it's a text, then you have to
    add as many , 'number' you have data. Each will be a row//

    var queryOptions = {

        csvColumns: ['string', 'number', 'number', 'number', 'number', 'number' ],

        csvHasHeader: true

    }

    var query = new google.visualization.Query(csvURL, queryOptions);
    query.send(handleQueryResponse);

}

</script>

</head>

<body>

    <div id="CAI" style="height: 500px; width: 1800px;"></div>
    <div id="AEON" style="height: 500px; width: 1800px;"></div>

</body>

</html>

```

END OF SCRIPT, DO NOT COPY THIS LINE!!

Copy this script to a notepad file, save as HTM. Sometimes your OS hide known file extensions and you may have a GRAPH.HTM.TXT which of course will not work, so change the way your OS show files in order to be able to save a HTM.

You can name your HTM as you want keeping in mind you can't use more than eight characters neither special characters.

GRAPH.HTM is a good deal.

Go to "Configuration", "Networking", "Web Server" and expand the "System" Web Server tree and copy the file from the location you created into MYDIR.

To call your graph just open your browser and type:
<http://your.IP.or.DOMAIN/USER/WEB/MYDIR/GRAPH.htm>

Port forward may be needed to be externally accessed.

Once again, thank you 50-50 and Teken for the inspiration.
 Sandry Seibert