

# Data Analysis Documentation

The Data Analysis Tool is a html+javascript based utility used by administrators, tuning experts, support engineers and consultants to create professional graphs based various types of input data. The tool is fully automatic and significantly faster than Excel at creating scatterplots, bar charts and line plots together with histogram analysis for almost any column separated input data such as **NMON** (*CPU\_ALL, MEM, MEM\_NEW, PROC, LPAR, PAGE*), **VMSTAT**, **SDFMON**, **TAANA**, **ABAPMETER**, **MHTML**, **NIPING** and **XLSX**. DataAnalysis.HTML will run in all modern web browsers like Chrome, Edge, Firefox, Opera, Safari (*Windows / Linux / MacOS / iPadOS*).

SAP internal note [3169320 - Data Analysis](#)

SAP JAM

Github

<https://community.sap.com/t5/technology-blog-posts-by-sap/data-analysis-tool/ba-p/13548321>

<https://github.com/i813812/DA-Data-Analysis>

## Contents

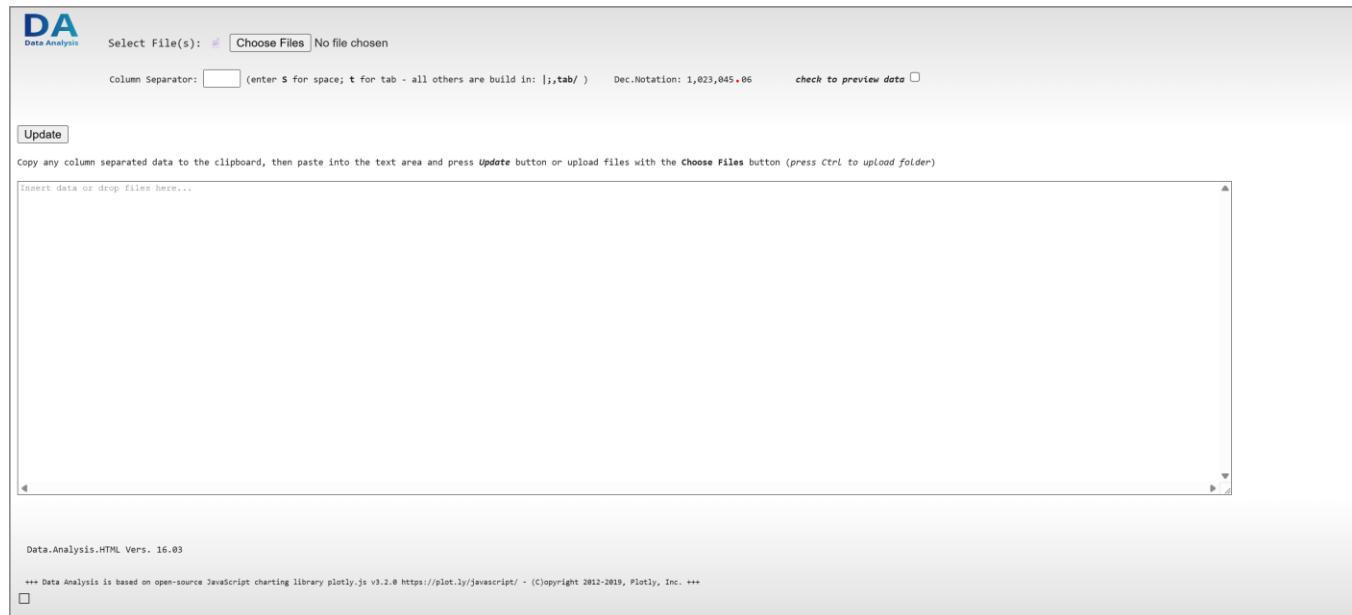
<b>1</b>	<b>SELECT INPUT DATA.....</b>	<b>2</b>
<b>2</b>	<b>REVIEW INPUT DATA.....</b>	<b>3</b>
<b>3</b>	<b>CUSTOMIZE GRAPHIC OUTPUT .....</b>	<b>5</b>
<b>4</b>	<b>ADDITIONAL OPTIONS.....</b>	<b>6</b>
4.1	1 <sup>ST</sup> LINE - X-AXIS .....	7
4.2	2 <sup>ND</sup> LINE - Y-AXIS .....	9
4.3	3 <sup>RD</sup> LINE - 2 <sup>ND</sup> Y-AXIS .....	10
4.4	4 <sup>TH</sup> LINE – HISTOGRAM AND COLORS .....	11
4.5	5 <sup>TH</sup> LINE - CHART OPTIONS.....	13
<b>5</b>	<b>PLOTLY OPTIONS .....</b>	<b>17</b>
<b>6</b>	<b>EXAMPLES .....</b>	<b>18</b>
<b>7</b>	<b>HISTOGRAM .....</b>	<b>19</b>
<b>8</b>	<b>PERFORMANCE.....</b>	<b>20</b>
<b>9</b>	<b>MEMORY USAGE: .....</b>	<b>20</b>
<b>10</b>	<b>ERRORS / BUGS .....</b>	<b>20</b>



## 1 Select Input Data

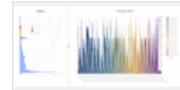
DA Data Analysis allows to open and concatenate multiple input files, it will detect automatically the column separator (*tab*, *semicolon*, *comma..*) - any character not included in 0..9, A..Z, a..z or *space* will be detected. To load data into the Data Analysis tool you have three options:

- Use the "Choose Files" button and select one or multiple files (*press Ctrl Key to toggle between file and folder selection - if folder is active DA will select all files in a folder*)
- Drag and drop one or multiple files over the "Choose files" button
- Copy & paste any column separated data into the input box
- In some cases when many large numbers with thousand delimiter are included a wrong column separator, typically , or . might be selected - in that case we need to specify the separator manually (*use S for space, T for tab or specify any other character //\.,;:-*)
- **Specify Decimal format** (*point or comma - if numbers are too small the program cannot determine the correct format automatically*)



The screenshot shows the DA Data Analysis web application. At the top left is the logo 'DA Data Analysis'. Below it is a form with the following fields:  
 - 'Select File(s):' followed by a 'Choose Files' button and a message 'No file chosen'.  
 - 'Column Separator:' with a dropdown menu containing 'S' (selected), 'T', and 'All others'. A note says '(enter \$ for space; t for tab - all others are build in: ;,tab/ )'.  
 - 'Dec. Notation:' with a field containing '1,023,045.06'.  
 - A checkbox 'check to preview data' with an unchecked state.  
 - A 'Update' button.  
 - A note below the update button: 'Copy any column separated data to the clipboard, then paste into the text area and press Update button or upload files with the Choose Files button (press Ctrl to upload folder)'.  
 - A large text area labeled 'Insert data or drop files here...' with scroll bars.  
 - At the bottom, there is footer text: 'Data.Analysis.HTML Vers. 16.03' and 'Data Analysis is based on open-source JavaScript charting library plotly.js v3.2.0 https://plot.ly/javascript/ - (C)opyright 2012-2019, Plotly, Inc. +++'.

Paste content of SDFMON into text area – then use button **Update**



## 2 Review Input Data

On the second screen one can review the data (*if required specify the date format*) and modify the column headers or go directly to the graphic display.

DA
Data Analysis

Graphic

Review data, if required edit header description(s), then press Graphic button to display the histogram & scatterplot - Select Date Format: YYYY-MM-DD ▾ Change

File	Filename	Index	Time	AS Instance	Act. WPs	RFC WPs	CPU Usr	CPU Sys	CPU Idle	Paging in	Paging out	Free Mem	EM alloc	EM attach	Em global	Heap Memor	Pri	Paging Mem	Dia	Upd	Enq		
F01	sdfmon2019.10.15.txt	I00000	23:59:30	chaerp01_S4P_15	1	1	150	2	2	95	0	0	20051768	7148	6884	4416622	0	0	129	16	0		
F01	sdfmon2019.10.15.txt	I00001	23:59:30	chaerp01_S4P_16	1	1	75	11	2	85	0	0	3.744e+07	6984	6884	4381498	0	0	1133	12	0		
F01	sdfmon2019.10.15.txt	I00002	23:59:30	chaerp01_S4P_13	5	4	75	39	3	57	0	0	2.4e-5	7152	7068	4382522	0	0	1698	17	0		
F01	sdfmon2019.10.15.txt	I00003	23:59:30	chaerp01_S4P_19	3	2	75	2	1	96	0	0	29913836	7336	6892	4386268	0	0	739	16	0		
F01	sdfmon2019.10.15.txt	I00004	23:59:30	chaerp01_S4P_25	3	3	75	2	1	96	0	0	27941556	7296	6892	4385594	0	0	547	39	0		
F01	sdfmon2019.10.15.txt	I00005	23:59:30	chaerp01_S4P_26	5	5	75	2	4	95	0	0	30059324	7426	6892	4386279	0	0	893	10	0		
F01	sdfmon2019.10.15.txt	I00006	23:59:30	chaerp11_S4P_27	1	1	0	1	1	95	0	0	0	7136	4484296	0	0	184	9	0			
F01	sdfmon2019.10.15.txt	I00007	23:59:30	chaerp11_S4P_28	6	2	75	59	3	36	0	0	-31980892	8912	7548	4384570	0	0	2478	12	0		
F01	sdfmon2019.10.15.txt	I00008	23:59:30	chaerp11_S4P_29	1	1	75	2	2	95	0	0	30024944	7268	6884	4383546	0	0	666	15	0		
F01	sdfmon2019.10.15.txt	I00009	23:59:30	chaerp11_S4P_30	3	3	75	2	1	95	0	0	33426176	7242	6884	4383546	0	0	879	16	0		
F01	sdfmon2019.10.15.txt	I00010	23:59:30	chaerp11_S4P_32	5	3	75	46	1	52	0	0	22346968	11328	18716	4387643	5873	0	4824	18	0		
F01	sdfmon2019.10.15.txt	I00011	23:59:30	chaerp11_S4P_33	3	3	75	2	1	95	0	0	33844868	7336	6892	4385594	0	0	1001	46	0		
F01	sdfmon2019.10.15.txt	I00012	23:59:30	chaerp11_S4P_34	3	3	75	2	1	95	0	0	30563212	7368	6892	4386619	0	0	716	10	0		
F01	sdfmon2019.10.15.txt	I00013	23:59:30	chaerp11_S4P_35	1	1	150	2	1	95	0	0	28822224	7186	6884	4414574	0	0	72	35	0		
F01	sdfmon2019.10.15.txt	I00014	23:59:30	chaerp11_S4P_36	2	1	75	2	1	95	0	0	39351668	6926	6892	4386559	0	0	9	0	0		
F01	sdfmon2019.10.15.txt	I00015	23:59:30	chaerp11_S4P_37	4	6	75	2	1	95	0	0	32553724	7108	6896	4381498	0	0	752	12	0		
F01	sdfmon2019.10.15.txt	I00016	23:59:30	chaerp11_S4P_38	1	1	150	2	1	95	0	0	27555648	7972	6884	4401261	0	0	58	9	0		
F01	sdfmon2019.10.15.txt	I00017	23:59:30	chaerp21_S4P_40	1	1	75	2	1	95	0	0	31747164	7448	6884	4381498	0	0	1847	15	0		
F01	sdfmon2019.10.15.txt	I00018	23:59:30	chaerp21_S4P_43	3	3	75	2	1	95	0	0	33426192	7284	6892	4382522	0	0	496	12	0		
F01	sdfmon2019.10.15.txt	I00019	23:59:30	chaerp21_S4P_45	1	1	75	2	1	95	0	0	32553724	7242	6884	4382522	0	0	1167	22	0		
F01	sdfmon2019.10.15.txt	I00020	23:59:30	chaerp20_S4P_46	2	1	75	0	1	69	0	0	43238000	9576	8956	4532036	0	0	4963	37	0		
F01	sdfmon2019.10.15.txt	I00021	23:59:30	chaerp30_S4P_47	2	1	75	11	0	0	0	0	34989568	7880	7000	4387643	0	0	6799	13	0		
F01	sdfmon2019.10.15.txt	I00022	23:59:30	chprccapp0_S4P_00	4	4	150	1	1	99	0	0	54209920	19120	4989188	0	0	65648	12	0			
F01	sdfmon2019.10.15.txt	I00023	23:59:30	chaerp01_S4P_15	2	2	150	1	1	95	0	0	29616132	7148	6884	4414574	0	0	139	16	0		
F01	sdfmon2019.10.15.txt	I00024	23:59:30	chaerp11_S4P_15	1	1	75	11	2	95	0	0	37350608	6926	6884	4381498	0	0	281	12	0		
F01	sdfmon2019.10.15.txt	I00025	23:59:30	chaerp01_S4P_33	6	4	75	39	3	57	0	0	24036916	7308	4382522	0	0	398	17	0			
+++ type / number of unique values +++																							
- / 1	- / 1	- / 2	t / 2	- / 2	n / 2	n / 2	m / 2	n / 2	n / 2	n / 2	n / 2	n / 1	n / 1	n / 2	n / 2	n / 2	n / 2	n / 2	n / 2	n / 2	n / 2	n / 2	n / 1

PgD ▼ PgU ▲ find:  replace with:

Download Data
Remove Duplicates
Delete empty columns
Delete every 2nd Row
Count Unique Values

Number of Records: 66227 - Records: 66227 (total) / 66227 (selected) / 66227 (display) - Runtime: 1366.30 (156.70) ms

\*\*\* Data Analysis is based on open-source JavaScript charting library plotly.js v3.2.0 <https://plot.ly/javascript/> - (C)opyright 2012-2019, Plotly, Inc. \*\*\*

The last line in the table is indicating what type of data has been detected and how many distinct values exist in each column - the format is **<type> / <number>**

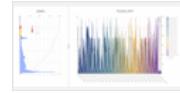
F01	sdfmon2019.10.15.txt	I66225	00:06:00	chaerp15_S4P_32	5	3	75	16	2	81	0	0	36074568	7324	7148	4384579	0	0	4382	1	0	0
F01	sdfmon2019.10.15.txt	I66226	00:06:00	chaerp16_S4P_33	2	1	75	9	2	83	0	0	32209636	7476	7248	4385545	0	0	4722	2	0	0
F01	sdfmon2019.10.15.txt	I66225	00:06:00	chaerp14_S4P_30	5	1	75	14	2	83	0	0	35608192	7288	7628	4381522	0	0	9095	1	0	0
F01	sdfmon2019.10.15.txt	I66225	00:06:00	chaerp15_S4P_32	5	3	75	8	1	90	0	0	34517416	7916	7688	4383546	12	1	6886	1	0	0
F01	sdfmon2019.10.15.txt	I66226	00:06:00	chaerp16_S4P_33	2	2	75	3	1	95	0	0	37350808	6968	6888	4383547	0	0	1008	19	0	0

The <type> indicates:

- text value
- d date
- t time
- n numeric

Author(s): Christoph Weyd  
Changed: 10.12.2025

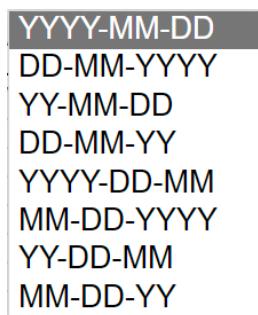
3 of 20



The user has some options to manipulate the input data (*the buttons are only available if the user is putting the cursor into the header line of a column*)

- Find and/or Replace values
- Sort Data
- Delete Values
- Split or Delete Columns
- Perform basic calculations like  
 $= <\text{value}> | \$nn [ \# [ <\text{value}> | \$nn ] ]$  or  
 $\# <\text{value}> | \$nn$   
 where  $\#$  can be any of the operators  $+-* /$   
 $\$nn$  refers to the content of column nn

In some occasions the program might not be able to determine the date format automatically, in that case the user can choose from different date formats



Additionally, the user can download the data (tab separated text file with extension .xls). To reduce the amount of data, columns which do not contain significant information (all values identical) can be removed. If files are too big then the user can also remove every 2<sup>nd</sup> data row.

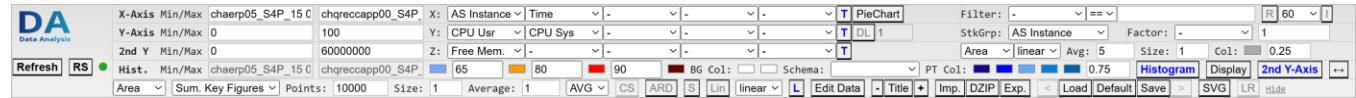
When selecting the header line of a non-numeric column you can use button **Pie Chart** to generate a Pie Chart showing the data distribution of the Top nn entries





## 3 Customize Graphic Output

Below the various options to customize the graphic output

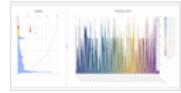


1. **Specify the fields for the X-Axis. Available are non-numeric fields (*date, time, server names...*) and all numerical fields**
2. **Specify the fields for the Y-Axis. Available are only numerical fields (*integer or float*)**

*Important: Certain fields for the X-Axis like Date/Time are automatically proposed - for the Y Axis the DataAnalysis will default CPU User, CPU System, Resp.Time...*

### Optional

3. Activate 2nd Y-Axis and specify fields, chart type...
4. Select the chart type - available options are:
  - Scatter Plot
  - Line Chart
  - Bar Chart
  - Area Chart
5. If multiple key figures are select for the Y-Axis specify if those values should be summarized, stacked over each other, or shown independently
  - Summarize Key Figures
  - Stack Key Figures
  - Independent Key Figures
6. Select if any key figures should be used to group/stack the data. Each different value for the selected Stack variable will be shown in a different color. This option is only available if there are multiple different values available. A graph can only be stacked with this option if for each x-value there are an equal number of y-values available. If any x-value will have a different number of y-values this x-value will not be shown (*this might happen if x-values (time) differ by a few seconds - as workaround one can try to activate rounding*).



## 4 Additional Options

In the upper left you see the actual status of the Chart and you can Restart, Refresh or Reset the Graph

- **Restart**

If you click on the Logo itself you can restart DA Data Analysis to load another data source.

- Recalculate and Refresh the Graphic

**Refresh**

- Reset Graphic and recalculate Axis scaling

**RS**

- **Status**

Ready ●    Refresh Pending ○    Busy ●



## 4.1 1st line - X-Axis

X-Axis Min/Max chaerp05\_S4P\_15\_0 chqreccapp00\_S4P\_ X: AS Instance v Time v - v - v - T PieChart Filter: - == R 60 v I

- X-Axis Min/Max – Specify the X-Axis Range

X-Axis Min/Max 00:34:00 | 64940 | 23:59:30 | 100000

- Select fields for X-Axis (Show/Hide Data Labels)

X: Host v Date v Time v Index v

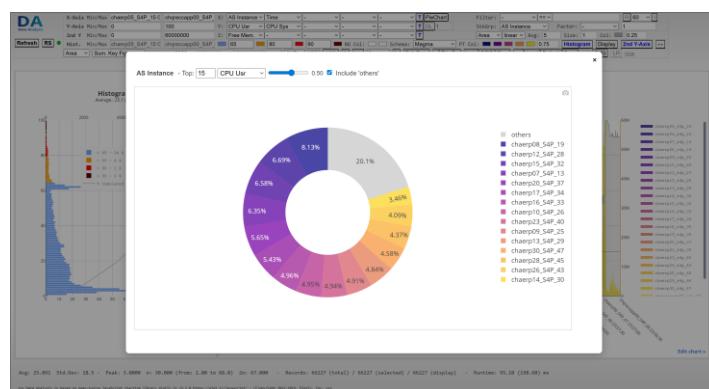
- Show/Hide X-Axis Labels (blue = Show, gray = hide)

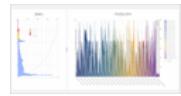


- Filter by selected x-value (== x-values equal selected values; <> x-values not equal selected values; multiple conditions can be combined with ||, wildcard \* allowed)

Filter: Host v == v bcgammma101 || bcgammma1

- If the selected field for the X-Axis is non-numeric a user can generate a Pie-Chart to show the Top nn values.





- Round time figures (select **R** to activate) to multiple of 1, 5, 10, 15 ... 60 ... 3600 seconds

 **R** 60 

Available Rounding Values are (in seconds)

1  
5  
10  
15  
20  
30  
**60**  
120  
300  
600  
900  
1200  
1800  
3600

- Specify if data points not present in all stack groups should be added or eliminated

 **I**  **A**  **E**

- I      Ignore data points which are not present in all stack groups  
A      Add missing data points with zero value  
E      Erase Data point from all stack groups if not present in all of them



## 4.2 2<sup>nd</sup> line - Y-Axis

Y-Axis Min/Max 0 | 100 | Y: CPU Usr ▾ CPU Sys ▾ - ▾ - ▾ - ▾ T DL 1 | StkGrp: AS Instance ▾ Factor: - ▾ 1

- Y-Axis Min/Max – Specify the Y-Axis Range

**Y-Axis Min/Max** 0 | 50

- Select fields for Y-Axis (Show/Hide Data Labels)

Y: CPU User% ▾ CPU Sys% ▾ - ▾ - ▾

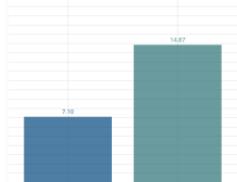
- Show/Hide Y-Axis Labels (blue = Show, gray = hide)

**T** **T**

- Data Labels (Bar Charts only)

**DL** 2

Show data Labels above Bars with n decimals



- Stack Group

StkGrp: Host ▾

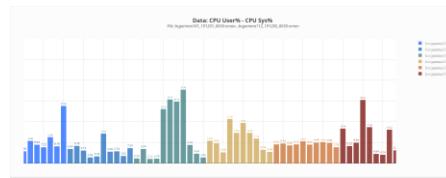
Data Points can be grouped by non-numeric fields as a separate data set. Each Stack Group will show in a separate color which will be interpolated from the four colors defined in the 4<sup>th</sup> line

### Example

X-Axis: Host + Time

Y-Axis: CPU Usage; Stack Group: Host

Colors: 



- Factor – we can multiply the Y-Values with the value of another column.

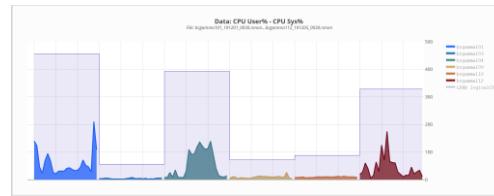
Factor: LPAR logicalC ▾ 0.01

A useful example is to multiply the CPU% with the total number of logical CPUs to display the logical CPUs used. We add here an additional factor of 0.01 to adjust for the percentage itself.



## Example

X-Axis: Host + Time  
 Y-Axis: CPU Usage; Stack Group: Host  
 Factor: logical CPUs \* 0.01  
 2<sup>nd</sup> Y-Axis: logical CPUs



## 4.3 3<sup>rd</sup> line - 2<sup>nd</sup> Y-Axis

2nd Y Min/Max	0	60000000	Z: Free Mem.	-	-	-	-	-	T	Area	linear	Avg: 5	Size: 1	Col:	0.25
---------------	---	----------	--------------	---	---	---	---	---	---	------	--------	--------	---------	------	------

- 2<sup>nd</sup> Y-Axis Min/Max – Specify the Y-Axis Range

2nd Y Min/Max	0	500
---------------	---	-----

- Select fields for 2<sup>nd</sup> Y-Axis (Show/Hide Data Labels)

Z:	LPAR logicalC	-	-	-	-	-
----	---------------	---	---	---	---	---

- Show/Hide 2<sup>nd</sup> Y-Axis Labels (blue = Show, gray = hide)

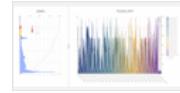
<span style="color: blue;">T</span>	<span style="color: gray;">T</span>
-------------------------------------	-------------------------------------

- Details for 2<sup>nd</sup> Y-Axis

Area	steps	Avg: 1	Size: 1	Col:	0.1
------	-------	--------	---------	------	-----

Here we specify how the 2<sup>nd</sup> Y-Axis is drawn. From left to right

- Chart Type (Area, Scatter, Line)
- How to connect Data Points (line, spline, steps)
- Average n data points
- Point/Line Size
- Point/Line Color
- Transparency (from 0.05 to 1.00)



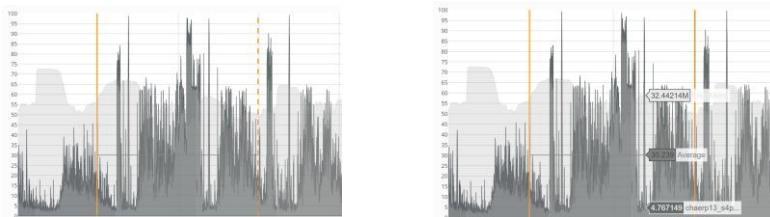
## 4.4 4<sup>th</sup> line – Histogram and Colors

Hist. Min/Max chaerp05\_S4P\_15 C chqreccapp00\_S4P\_ [65] [80] [90] BG Col: [ ] Schema: Magma PT Col: [ ] 0.75 Histogram Display 2nd Y-Axis [ ]

- Histogram Range (Display only)

Hist. Min/Max chaerp05\_S4P\_15 C chqreccapp00\_S4P\_

Single click on the Chart to set first the left border (the right border will show in a dashed orange line), then click again to set the left border.



- Specify data range and color for the histogram (3rd line)

[65] [80] [90] [ ]

- Select background and chart color

[ ] BG Col: [ ] [ ]

- Color Schema

Schema: Magma [ ]

available color options are

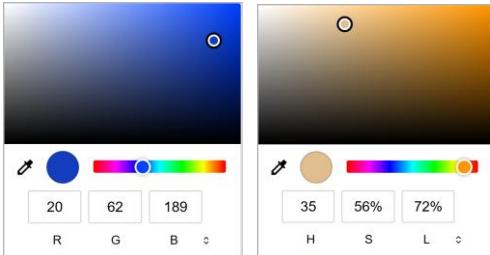
Blue	[Color Swatch]					
Orange	[Color Swatch]					
Red	[Color Swatch]					
Green	[Color Swatch]					
Cyan	[Color Swatch]					
Purple	[Color Swatch]					
Black to White	[Color Swatch]					
Vivid	[Color Swatch]					
Cividis	[Color Swatch]					
Magma	[Color Swatch]					
Spectral	[Color Swatch]					
Rainbow	[Color Swatch]					



- Specify color for the 1<sup>st</sup>, 2<sup>nd</sup>, 3<sup>rd</sup> and 4<sup>th</sup> select Y-Value and Transparency. If a stack/group variable is used this for color will specify the color range used.



Click on any color to manually select or choose the color picker 



The transparency ranges from 0.05 (almost invisible) to 1.00 (opaque)

- Show/Hide the histogram



- Display Data used in Chart



- Enable/Disable 2<sup>nd</sup> Y-Axis



- Swap primary with 2nd Y-Axis



- Switch to 3D Display





## 4.5 5<sup>th</sup> line - Chart Options

Area  Points:  Size:  Average:  AVG  ARD  Lin linear  Edit Data   Imp. DZIP Exp. < Load Default Save >  LR Hide

- Select Chart Type - available Chart Types are:

Bar  
Scatter  
Line  
**Area**

- Handling of Data Points. If multiple key figures are select for the Y-Axis specify if those values should be summarized, stacked over each other or shown independently

**Sum. Key Figures**  
Stack Key Figures  
Indep. Key Figures

- Maximum number of points to be drawn - if the data contains after aggregation  $m$  points and only  $x$  point should be drawn then only every  $n^{th} = m/x$  points will be drawn

Points:

- Specify point size (*also line width in points, default = 3*)

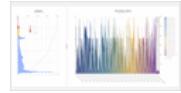
Size:

- Select how many data points should be aggregated (*calculate average of  $n$  points*)

Average:

- Specify calculation type if multiple data points are cumulated - default is AVG (average), available options are:

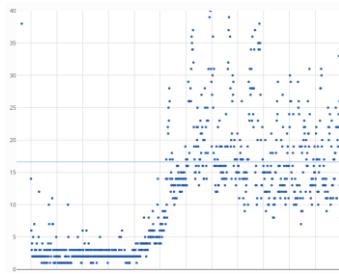
SUM  
**AVG**  
MAX  
MIN



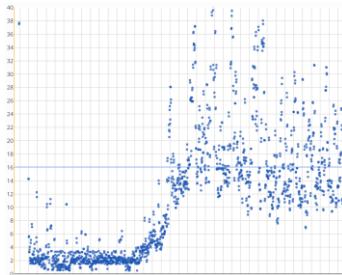
- Add random decimals to prevent banding.

**ARD**

If the measurements like CPU usage is only available with limited decimals values the graph will look unnatural because all values are on discrete lines. With button S one can add to each key figure a random value  $r$  between  $-d/2 < r < d/2$  where  $d$  is the smallest detected difference between the y values. See below Example



CPU values from SDFMON do not have decimals



Random decimals added (here between +/- 0.5)

- Reduce Number of Data Points (Algorithm)

**S** **T**

**S** Skip every n'th data point

**T** Largest-Triangle-Three-Buckets

(see sveinn-steinarsson/plot-downsample: Downsample plugin for Flot charts. (github.com))

- Switch between linear or logarithmic Y-Axis

**Log** **Lin**

- Specify how data points get connected in a line chart (linear, spline, or steps)

**linear**

spline

steps

- Display or Hide Chart Legend

**L**

- Edit/Preview Data

**Edit Data**

- Set Chart & Axis Title (*supported are the modifiers: <b> Bold </b>, <i> italic </i>, <sub> subscript </sub>, <sup> superscript </sup>, and line breaks <br>* ) and increase or decrease the font size

**-** **Title** **+**



- Import/export Layout Profile or Export Chart

Imp. DZIP Exp.

The two buttons IMP. and Exp. allow to import or export a JSON which contains all chart settings. Those settings can be used when another data source is loaded.

The button DZIP is generating a compressed JSON file which contains the chart settings and the data itself. Typically the DZIP file is around 20% of the original file size.

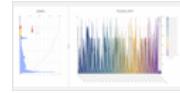
- Load/Save Settings set defaults. Undo/Redo last changes.

< Load Default Save >

The buttons Load and Save import/export the current chart settings into memory. They can be applied to the next file loaded. The two buttons <> allow to undo or redo the last changes.

- Save Chart as SVG Vector graphic

SVG

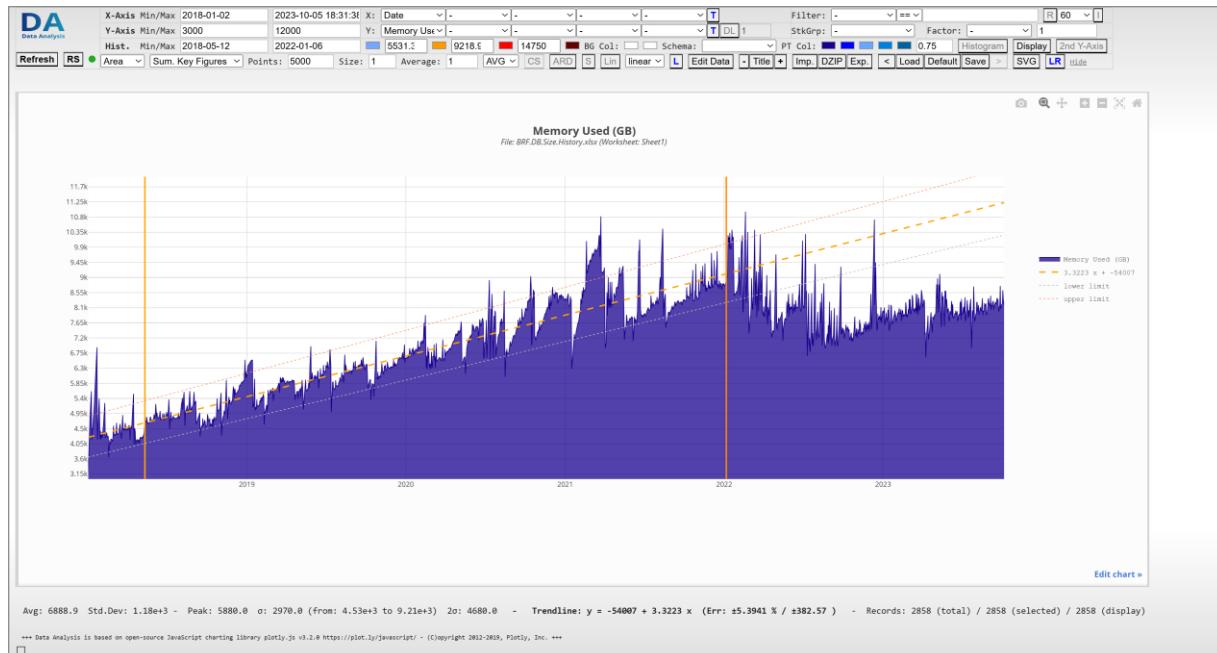


- Create linear regression based trendline (only if X-Axis is Date/Time)

 disabled

 calculate trendline between delimiters

 calculate extended trendline



The trendline will indicate the forecast and upper/lower limit; the equation is shown below the graphic. The factor is always measured in xxx per day.

**Trendline:  $y = -54007 + 3.3223 x$  (Err: ±5.3941 % / ±382.57 )**

- Autohide the chart options (that will make the charts bigger if the cursor is moved over them, the menu is shown again if the cursor is moved to the top)

[Hide](#) [Show](#)



## 5 Plotly Options

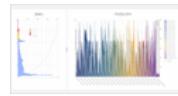
The below options are visible when you hover the mouse above the graph



- **Zoom** - use the mouse to select a rectangular area to be shown
- **Move or Re-Scale axis** (*move mouse to the end of the axis scales until an arrow is shown, then lets-click and move*)
- **Download Graph as PNG file**
- **Autoscale / Reset Scale**
- **Export** and open chart in Plotly Chart Studio
- Change scaling of X-Axis (double click on graph) – A popup allow to enter the tick distance (in hours, minutes, seconds...)

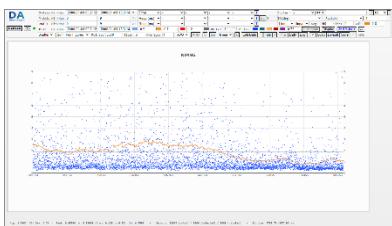


- If multiple values are shown, click on the legend to show/hide individual key figure or stack values

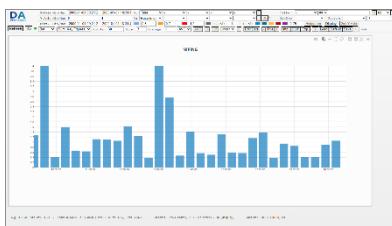


## 6 Examples

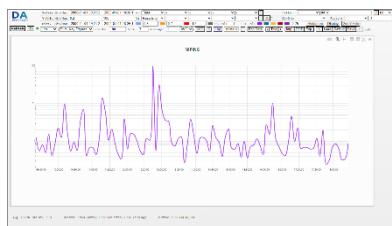
Available graph options are:



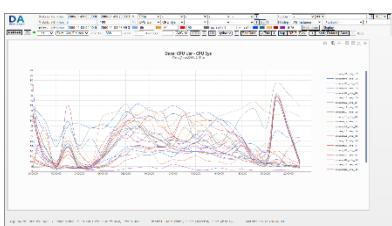
Scatter plot



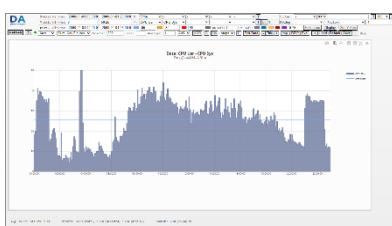
Bar chart



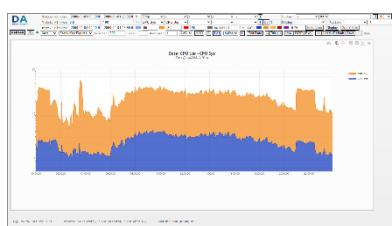
Line chart



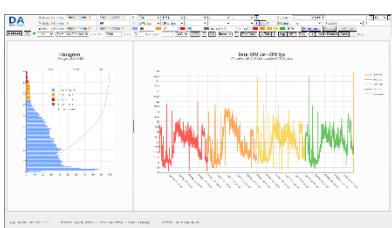
Multi Line Charts



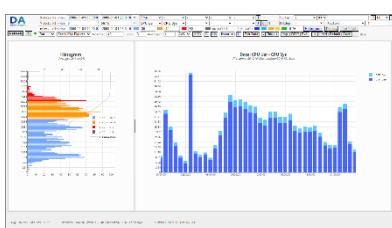
Area Charts



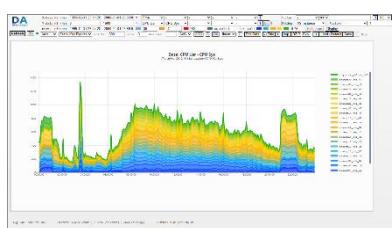
Stacked Area Charts



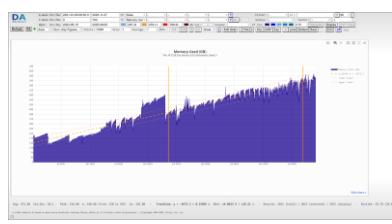
Grouped Line Charts  
(SDFMON per Date)



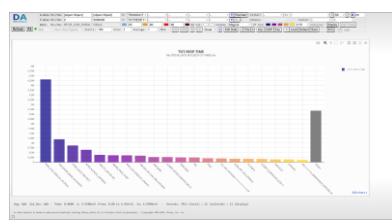
Stacked Bar Charts  
(SDFMON per Appl.Server)



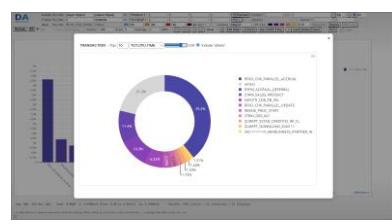
Stacked Area Charts  
(SDFMON per Appl.Server)



Trendline



Top n (descending)  
for Bar Charts only

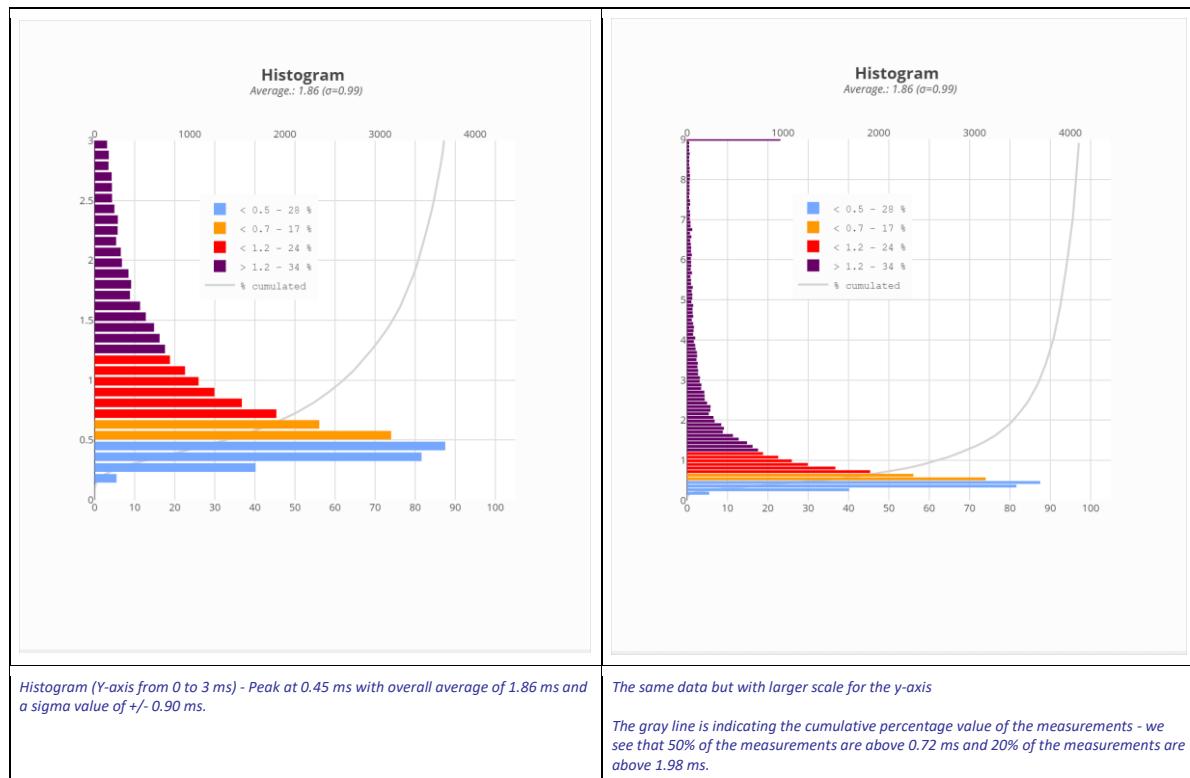


Top n Pie Chart  
(only for non-numeric fields)



## 7 Histogram

The histogram is calculated for the specified y-axis range divided in 100 equal intervals. For each of those intervals we count how many measurements are recorded and display the result as a horizontal bar chart. The color coding can be specified in the top of the screen:



Especially for highly fluctuating data (*NIPING results, CPU usage, number of active work processes*) a line/bar or scatterplot alone are difficult to interpret and often lead to wrong conclusions. A quantifiable metric to describe the difference in fluctuations is not available on those charts.

**Only a histogram analysis allows to exactly describe the random data distribution.** Especially for sizing purposes a statement like "95% (2 sigma) of all measurements are below 75 of CPU usage" is much more accurate than the standard deviation a line drawn above the highest observed values. Scatterplots combined with a histogram show the reality, how the individual measurements are fluctuating and at the same time to quantify the fluctuations, distribution and upper lower limits in percentage to the number of recorded measurements. Using maximum/minimum or averages do **NOT** show the reality and should **NOT** be used in the context of capacity analysis. Assuming the histogram follows a gauss normal distribution then we can conclude that:

- In the interval of the deviation  $\pm\sigma$  of the expected value, 68.27 % of all measured values can be found,
- In the interval of the deviation  $\pm 2\sigma$  of the expected value 95.45 % of all measured values are found,



## 8 Performance

Depending on the file type, record length, number of columns the Data Analysis tool can load and parse approximately **140.000** records per second. The throughput of the graphic engine depends on the number of operations and fields but typically Data Analysis can calculate and draw around **15.000** measurement points per second. We can estimate the average total processing time **T** of the Data Analysis tool by  $T \text{ (ms)} = R/1000 * 7 + P/1000 * 60$  where **R** is the total number of records loaded, **P** the number of data points in the graphic.

### Example(s):

- We process 5 NIPING files (*with 1 sec frequency*) which contain a total of **432.000 records** - from those records we plot **20.000 data points**. The estimated total processing time will be at  $432 * 5 + 20 * 60 = 4224 \text{ ms (4.2 seconds)}$ .
- A single NIPING file of 86.400 records with 10k data point will be ready in  $0.60 + 0.60 = 1.2$  seconds.
- We open 30 nmon files (*24h capture, snapshot every 5min, 57 CPU cores, each file with 150.000 records*) with the nmon analyzer to extract CPU\_ALL, PROC, LPAR, MEM, MEM\_NEW then each file takes about 1 minute of processing time. If we merge all files together with the nmon analyzer the process takes around 30 minutes. The Data Analysis tool can load and display the same data in less than 10 seconds.

## 9 Memory Usage:

The memory usage of Data Analysis tool depends on the file size and number of data points to be displayed. When we open a SDFMON file with 1.586.920 records the initial memory consumption reached 5GB of memory. During the creation of the graphic the total memory consumption of the browser (opera) increased to 11.5 GB of memory drawing all 1.586.920 data points (runtime 75 seconds). When we limit the number of data points to 200.000 the memory consumption reached only 2.1GB. The memory consumption scales with the number of data points drawn in the graphic and can be estimated at about 5-10 kB per data point.

## 10 Errors / Bugs

In case of any errors, please send an email to [c.weyd@sap.com](mailto:c.weyd@sap.com) and include a copy (*or link*) to the used data source and a screenshot of the browser window (*including the used settings in the top*). Please also describe which browser and operating system was used.

*Data Analysis is based on open-source JavaScript charting library [plotly.js https://plot.ly/javascript/](https://plot.ly/javascript/)  
(Copyright 2012-2019, Plotly, Inc. Licensed under the MIT license ([view the source on GitHub](#).)*