

NMIS8 Graphs Showing letters after the numbers in the y-axis

- [Summary](#)
- [Example](#)
- [Customization](#)
 - [Unit autoscaling](#)
 - [Fixed Y-Axis Range](#)

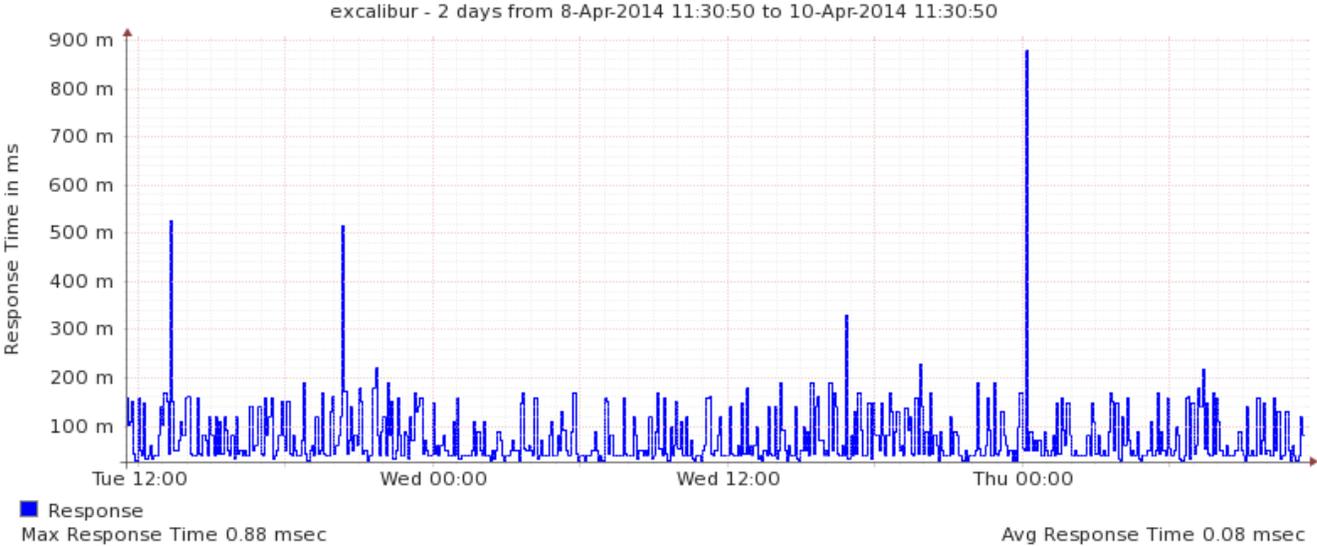
Summary

NMIS8 uses the [RRDTool time series database tool](#) for data storage and graphing. When RRDTool has to represent very small or very large numbers, it will normally use SI units instead of showing many decimal places. The following table summarises those units.

Unit	Name	Decimal Places
a	atto	-18
f	femto	-15
p	pico	-12
n	nano	-9
μ	micro	-6
m	milli	-3
	(no unit)	0
k	kilo	3
M	mega	6
G	giga	9
T	tera	12
P	peta	15
E	exa	18

Example

So in the chart below the little m in this case is 100m to 900m milliseconds is in fact 0.1 to 0.9 milliseconds.



Customization

If you find this feature undesirable (e.g. because the graph is for percentages), then there are two options for you: you can disable the unit autoscaling, or you can give the graph a fixed y-axis range. Of the two choices, disabling the automatic scaling has less potential for unintended side-effects, but it is still **highly advisable** that you read the relevant [documentation on rrdtool graph options](#) before making that choice.

Unit autoscaling

You can adjust the relevant graph definitions to not use unit autoscaling. Find the graph file you want to adjust in `/usr/local/nmis8/models`, and open it with an editor. Locate the `options` section, and add the `--units-exponent` setting to both `standard` and `small` graph definitions, like in the example below:

```
# ... header lines
%hash = (
# ...
  'option' => {
    'standard' => [
      '--units-exponent', '0',
      'DEF:ifInUcastPkts=$database:ifInUcastPkts:AVERAGE',
# ... lots of graph definition statement lines
    ],
    'small' => [
      '--units-exponent', '0',
      'DEF:ifInUcastPkts=$database:ifInUcastPkts:AVERAGE',
# ...
```

Fixed Y-Axis Range

As above you need to find the relevant graph file, open it with an editor, and change the relevant `option` section to include `lower-limit` and `upper-limit`, and possibly `rigid` as well.

The example below is for a fixed-range 0..100 percent graph:

```
'option' => {
  'standard' => [
    '--upper-limit', '100',
    '--lower-limit', '0',
    '--rigid',
```