

How to build plots on response time, memory and CPU usage.

Building all plots require *gnuplot* and *latex* to be installed.

Building response time plots:

1. Check a log format name on response time. Change the log file format in *createdat.sh* file if needed.
createdat.sh takes as a parameter delay time in the name of a response log file. Typical response log file should contain strings like 'WRITE 251111000 READ 255283000'.
2. Run *createdat.sh*. *createdat.sh* requires *mean.pl* file.
This will create a response time distribution over the delay time in ms in *.dat* files.
3. Run *function.sh*. Change log file format in *function.sh* if needed.
The output of *function.sh* is data to build plots (mean response delay to request load). *Function.sh* sometimes can't compute a mean response delay for the particular request load; in this case 'manual computing' need to be done.
4. Sort the output of *function.sh* by request load and store it in a *.dat* file, for example *nio2.sync.dat*. This file is used by *lmls.gp* to build a plot.
5. Repeat 1-4 for different server types or server modes.
6. Edit *lmls.gp* file. List here all data files obtained in step 4 you are going to show on the same plot.
7. Run *gen.sh*. *gen.sh* requires *lmls.gp* and *t.tex* file.
8. Check generated *.pdf* file.

Manual computing of a mean of the response time distribution:

1. Type *gnuplot*
2. Type gamma distribution equation:
$$f(x) = (\text{abs}(b) ** \text{abs}(a)) / \text{gamma}(\text{abs}(a)) * x^{(\text{abs}(a)-1)} * \exp(-\text{abs}(b) * x).$$

Type initial values of *a* and *b*, for example *a* = 5 *b* = 8. The default values of *a* and *b* equal to 1, so sometimes it is too far from the solution.
3. Fit *f(x)* with some distribution of the response time for the particular delay, for example:
fit f(x) "1000-75-100000.dat" using 1:2 via a,b.
4. To look at the plot on a particular interval type [0:150]:
plot [x=0:150] "./1000-75-100000.dat" with linespoints, f(x) with lines;

Building CPU usage plots.

1. Check log file format name. Change format in *cpu-dat.sh* if needed.
2. Run *cpu-dat.sh*. *cpu-dat.sh* requires *cpu.pl*.
3. Sort the output of *cpu-dat.sh*, store it in some *.dat* file, for example, *nio2.sync.dat*.
4. Repeat 1-3 for different server type log files.
5. Edit *lmls.gp*.
6. Run *gen.sh*. *gen.sh* requires *lmls.gp* and *t.tex* files.
7. Check generated *.pdf* file.

Building memory usage plots.

1. Check log file format name. Change format in *cpu-dat.sh* if needed.
2. Run *mem-dat.sh*. *mem-dat.sh* requires *mem.pl*.
3. Sort the output of *mem-dat.sh*, store it in some *.dat* file, for example, *nio2.sync.dat*.
4. Repeat 1-3 for different server type log files.
5. Edit *lmls.gp*.
6. Run *gen.sh*. *gen.sh* requires *lmls.gp* and *t.tex* files.
7. Check generated *.pdf* file.