Baselines, Plotting and Gnuplot Tutorial

Advanced Systems Lab
Fall 2019
3 Clients connect to 1 Server

<table>
<thead>
<tr>
<th><strong>Number of servers</strong></th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Number of client machines</strong></td>
<td>3</td>
</tr>
<tr>
<td><strong>Instances of memtier per machine</strong></td>
<td>1</td>
</tr>
<tr>
<td><strong>Threads per memtier instance</strong></td>
<td>3</td>
</tr>
<tr>
<td><strong>Virtual clients per thread</strong></td>
<td>4, 8, 16, 32</td>
</tr>
<tr>
<td><strong>Workload</strong></td>
<td>Read-only</td>
</tr>
<tr>
<td><strong>Value Size (Bytes)</strong></td>
<td>64, 256, 512, 1024</td>
</tr>
<tr>
<td><strong>Number of middlewares</strong></td>
<td>N/A</td>
</tr>
<tr>
<td><strong>Worker threads per middleware</strong></td>
<td>N/A</td>
</tr>
<tr>
<td><strong>Repetitions</strong></td>
<td>3 or more (at least 1 minute each)</td>
</tr>
</tbody>
</table>
Note: Lines in Your Plots will not look exactly like these ones.
3 Clients connect to 1 Server

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of servers</td>
<td>3</td>
</tr>
<tr>
<td>Number of client machines</td>
<td>3</td>
</tr>
<tr>
<td>Instances of memtier per machine</td>
<td>3</td>
</tr>
<tr>
<td>Threads per memtier instance</td>
<td>1</td>
</tr>
<tr>
<td>Virtual clients per thread</td>
<td>4, 8, 16, 32</td>
</tr>
<tr>
<td>Workload</td>
<td>Read-only</td>
</tr>
<tr>
<td>Value size (Bytes)</td>
<td>64, 256, 512, 1024</td>
</tr>
<tr>
<td>Number of middlewares</td>
<td>N/A</td>
</tr>
<tr>
<td>Worker threads per middleware</td>
<td>N/A</td>
</tr>
<tr>
<td>Repetitions</td>
<td>3 or more (at least 1 minute each)</td>
</tr>
</tbody>
</table>

2.2.1 Setup (5 pts)
Describe the setup you used for this experiment, how did you launch and run the experiments, if you always used the same deployment or not, etc.

2.2.2 Throughput (10 pts)
Plot the throughput as a function of NumClients and value sizes. All saturation points must be included and described. Use the same plot setup as in Section 2.1.2.

2.2.3 Response Time (10 pts)
Plot the response time as a function of NumClients and value sizes. All saturation points must be included and described. Use the same plot setup as in Section 2.1.3.

2.2.4 Result Analysis (5 pts)
Explain the results are explained by the interactive law (and why) and if they are consistent with previous experiments.

2.2.5 Explanation (20 pts)
Describe in which phase the memcached servers are under-saturated, saturated, or over-saturated. Describe how throughput and response time correlate. Explain the effect that different NumClients and value sizes have on throughput and response time. Explain what further conclusions can be drawn from the experiment.

2.3 Summary (20 pts)
Based on the experiments above, fill out the following table:

<table>
<thead>
<tr>
<th>Configuration (number of clients, msg. size)</th>
<th>Maximum Throughput</th>
<th>Minimum Response Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>One memcached server</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Three memcached servers</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2.3.1 Bottleneck Analysis (10 pts)
Describe what are the bottlenecks of this setup is. If the maximum throughput for both experiments is the same, explain why. If it is not the case, explain why not. Explain how the system behaviour when the number of clients and value size increase.

Name: YOUR NAME Legi: YOUR LEGI 6
Note: Lines in Your Plots will not look exactly like these ones.
1 MW and 1 Server

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of servers</td>
<td>1</td>
</tr>
<tr>
<td>Number of client machines</td>
<td>3</td>
</tr>
<tr>
<td>Instances of memtier per machine</td>
<td>1</td>
</tr>
<tr>
<td>Threads per memtier instance</td>
<td>2</td>
</tr>
<tr>
<td>Virtual clients per thread</td>
<td>4, 8, 16, 32</td>
</tr>
<tr>
<td>Workload</td>
<td>Read-only</td>
</tr>
<tr>
<td>Value size (Bytes)</td>
<td>64, 256, 512, 1024</td>
</tr>
<tr>
<td>Number of middlewares</td>
<td>1</td>
</tr>
<tr>
<td>Worker threads per middleware</td>
<td>8, 32, 64</td>
</tr>
<tr>
<td>Repetitions</td>
<td>3 or more (at least 1 minute each)</td>
</tr>
</tbody>
</table>
Plotting Best Practices

• Start axis at zero, try and keep same range for related graphs
• Label both axis, state units clearly
  • Use Ops/s not Ops/minute, and other “exotic” units
  • Instead of 12000000 use 1.2 million
• Caution with logarithmic scales on axis
• Include error bars!
• Make sure system configuration is easily found
• You will see more examples in the exercises…
What is Wrong with the Plot?

- Line colors are similar.
- Missing over saturated part of the graph.
And this one?

“The response time grows significantly after 48 concurrent clients for most thread levels.”

What the student should do?

- Check if losing requests?
- Were the experiments run on the same setup?
- Are the measurements sound?
- Sleeps or timeouts in worker threads?
What is Wrong with the Plot?

- Y axis does not start from 0.
- Use lines with points.

“Also, observe that the data fluctuates a lot but this is again due to the lack of repetitions. The experiments between 300 and 550 clients were performed at a different time than the others, implying that the virtual machines were shut down, which explains this performance drop.”

What the student should do?

- The graph is overdone, the part after 400 clients is not necessary.
- Fluctuations will appear less if Y axis starts from 0.
- Should do repetitions.
What is Wrong with the Plot?

- Misses under and over saturated parts of the plot.
- Y axis does not start from 0. lines will be flat if it starts from 0.

“I can report 60, 800 as my middleware highest throughput since the throughput doesn’t change much compared to previous reported configuration”
Student explanation:

“Zero mean test suggests that Configurations 120, 160 and 200 Threads do not have statistically significant difference, hence we pick 120 threads configuration at 300 clients.”
Description vs. Explanation

• Example sentence 1: We observe in Figure 1 that the throughput increases linearly with the number of clients, until X number of clients is reached.

• Example sentence 2: Since the rate of increase in response time changes suddenly at X number of clients, we deduce the system is saturated at...

• Example sentence 3: The throughput saturation is reached at X number of clients, because ...

• Example sentence 4: The interactive law holds as expected, as shown in Figure 1.
Description vs. Explanation

- Example sentence 1: We observe in Figure 1 that the throughput increases linearly with the number of clients, until X number of clients is reached.

- Example sentence 2: Since the rate of increase in response time changes suddenly at X number of clients, we deduce the system is saturated at…

- Example sentence 3: The throughput saturation is reached at X number of clients, because …

- Example sentence 4: The interactive law holds as expected, as shown in Figure 1.
Gnuplot 101

• Gnuplot is a command line tool for generating 2D and 3D plots of your data
• Works on Linux and Windows
• Can be used in interactive or scripted mode

<table>
<thead>
<tr>
<th>#</th>
<th>x</th>
<th>y</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>0.3</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>0.4</td>
<td>-1.1</td>
<td>0.99</td>
<td></td>
</tr>
<tr>
<td>2.3</td>
<td>1.0</td>
<td>0.97</td>
<td></td>
</tr>
</tbody>
</table>
How to organize your data?

• Output your experimental results as a comma/space/tab separated file

```
# x  value1  value2
0.1  0.3     11
0.4  -1.1    0.99
2.3  1.0     0.97
0.1  -0.1    1.8
```

• You can also include text

```
# category   x  value1  value2
First        0.1  0.3     11
Second       0.4  -1.1    0.99
Third        2.3  1.0     0.97
Fourth       0.1  -0.1    1.8
```
Hello world

• Start Gnuplot and type interactively, or create a text file with commands:

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>#</td>
<td>age</td>
</tr>
<tr>
<td>18</td>
<td>100</td>
</tr>
<tr>
<td>19</td>
<td>200</td>
</tr>
<tr>
<td>20</td>
<td>300</td>
</tr>
<tr>
<td>21</td>
<td>400</td>
</tr>
</tbody>
</table>

plot “results.txt” using 1:2 with lines

plot title “hello world”
Gnuplot scripts

• Plots are set up declaratively – see online documentation for all commands

Start with set term png/pdf/svg to select the output type
• set output “filename.png”
• set xlabel “x axis label”
• set ylabel “y axis label”
• plot “filename1” using 1:2 with lines title “first”,
  “filename2” using 1:3 with linespoints title “second”
Column values can be combined

- It is possible to plot simple expressions:
  - Plot “file.txt” using $1:($2+$3+$4) with lines title “summed columns”

<table>
<thead>
<tr>
<th>#</th>
<th>t</th>
<th>req_type1</th>
<th>req_type2</th>
<th>req_type3</th>
<th>$2+$3+$4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>111</td>
<td>0</td>
<td></td>
<td>111</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>100</td>
<td>3</td>
<td></td>
<td>105</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>120</td>
<td>0</td>
<td></td>
<td>124</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>100</td>
<td>22</td>
<td></td>
<td>122</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>102</td>
<td>2</td>
<td></td>
<td>107</td>
</tr>
</tbody>
</table>

- You can use other arithmetic operations as well!
Example: Plotting response time

<table>
<thead>
<tr>
<th></th>
<th>CLIENTS</th>
<th>RESPONSE_TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>10000</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>20000</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>40000</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>60000</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>100000</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>150000</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>200000</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>300000</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td>800000</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td>1500000</td>
</tr>
</tbody>
</table>
Example: Plotting response time

set style line 1 lc rgb '#0060ad' lt 1 lw 2 pt 7 ps 1.5  # --- blue
set xlabel ""
set ylabel ""
set title ""
set xrange [-10:10]
set yrange [-1500000:1500000]
plot "data.txt" using 1:2 ls 1

What is wrong here?
Example: Plotting response time

set style line 1 lc rgb '#0060ad' lt 1 lw 2 pt 7 ps 1.5  # --- blue
set xlabel ""
set ylabel ""
set title ""
set xrange [-10:10]
set yrange [-1500000:1500000]
plot "data.txt" using 1:2 ls 1

- No labels or title.
- Legend is meaningless.
- Ranges are completely off.
- No lines.
Example: Plotting response time

```bash
set style line 1 lc rgb '#0060ad' lt 1 lw 2 pt 7 ps 1.5  # --- blue
set xlabel ""
set ylabel ""
set title ""
set xrange [-10:10]
set yrange [-1500000:1500000]
plot "data.txt" using 1:2 with lines ls 1
```

- No labels or title.
- Legend is meaningless.
- Ranges are completely off.
- No ticks.
Example: Plotting response time

- No labels or title.
- Legend is meaningless.
- Ranges are completely off.
Example: Plotting response time

set style line 1 lc rgb '#0060ad' lt 1 lw 2 pt 7 ps 1.5  # --- blue
set xlabel " "
set ylabel " "
set title " "
set xrange [0:11]
set yrange [10000:1500000]
plot "data.txt" using 1:2 with linespoints ls 1

What is wrong here?
Example: Plotting response time

set style line 1 lc rgb '#0060ad' lt 1 lw 2 pt 7 ps 1.5  # --- blue
set xlabel ""
set ylabel ""
set title ""
s xrange [0:11]
set yrange [10000:1500000]
plot "data.txt" using 1:2 with linespoints ls 1

- No labels or title.
- Legend is meaningless.
- y-axis range does not start from 0.
Example: Plotting response time

set style line 1 lc rgb '#0060ad' lt 1 lw 2 pt 7 ps 1.5  # --- blue
set xlabel " "
set ylabel " in thousands "
set title " "
set xrange [0:11]
set yrange [0:1600]
plot "data.txt" using 1:($2/1000) with linespoints ls 1

- No labels or title.
- Legend is meaningless.
- x-axis ticks are not correct
Example: Plotting response time

```
set style line 1 lc rgb '#0060ad' lt 1 lw 2 pt 7 ps 1.5   # --- blue
set xlabel "Number of Clients"
set ylabel "Response time (s)"
set title "Response time over Number of Clients"
set xrange [0:11]
set yrange [0:1600]
plot "data.txt" using 1:($2/1000):xticlabel(1) with linespoints ls 1 nolegend
```

Graph showing response time over number of clients.
Example: Plotting Error Bars

set style line 1 lc rgb '#0060ad' lt 1 lw 2 pt 7 ps 1.5  # --- blue
set xlabel " Number of Clients"
set ylabel " Throughput (transactions/second)"
set title " Throughput vs. Number of clients"
set xrange [0:11]
set yrange [0:80]
plot "data.txt" using 1:2:3:xticlabel(1) with errorlines title "System X, config ABC" ls 1

<table>
<thead>
<tr>
<th># CLIENTS</th>
<th>THROUGHPUT</th>
<th>STDDEV</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>11.2</td>
<td>1.1</td>
</tr>
<tr>
<td>2</td>
<td>20.1</td>
<td>1.2</td>
</tr>
<tr>
<td>3</td>
<td>28.9</td>
<td>1.4</td>
</tr>
<tr>
<td>4</td>
<td>37.0</td>
<td>4.5</td>
</tr>
<tr>
<td>5</td>
<td>40.1</td>
<td>5.5</td>
</tr>
<tr>
<td>6</td>
<td>42.2</td>
<td>7.0</td>
</tr>
<tr>
<td>7</td>
<td>43.0</td>
<td>14.4</td>
</tr>
<tr>
<td>8</td>
<td>43.3</td>
<td>22.2</td>
</tr>
<tr>
<td>9</td>
<td>44.4</td>
<td>17.0</td>
</tr>
<tr>
<td>10</td>
<td>42.1</td>
<td>2.2</td>
</tr>
</tbody>
</table>
Example: Plotting Multi Lines

set style line 1 lc rgb '#0060ad' lt 1 lw 2 pt 7 ps 1.5
set style line 2 lc rgb '#ad6000' lt 1 lw 2 pt 7 ps 1.5
set style line 3 lc rgb '#60ad00' lt 1 lw 2 pt 7 ps 1.5
set xlabel "Number of Clients"
set ylabel "Throughput (transactions/second)"
set title "Throughput vs. Number of clients"
set xrange [0:11]
set yrange [0:80]
plot "data.txt" index 00:00 using 1:2:3:xticlabel(1) with errorlines title "System X, config ABC" ls 1,
"data.txt" index 01:01 using 1:2:3:xticlabel(1) with errorlines title "System X, config DEF" ls 2,
"data.txt" index 02:02 using 1:2:3:xticlabel(1) with errorlines title "System X, config GHI" ls 3

Legends are important in multi-line graphs.
Example: Plotting Stacked Bar Charts

set style data histogram
set style histogram rowstacked
set style fill solid 1.0 border -1
set boxwidth 0.8 relative
set xlabel "Number of Clients"
set ylabel "Throughput (transactions/second)"
set title "Throughput vs. Number of clients"
set xrange [-1:10]
set yrange [0:100]
plot "DATA_FILE" using 2 title "Throughput 1", " using 3 title "Throughput 2"
Linear vs. Log Scale

Response time over Number of Clients

- Linear scale: The response time increases linearly with the number of clients, showing a nearly constant slope.
- Log scale: The response time increases logarithmically with the number of clients, showing a diminishing increase as the number of clients grows.
Recommendations

• As you progress and rerun experiments save results in different files – you can regenerate graphs for different versions of the result

• Keep style consistent over graphs

• Use bash scripts for exporting graphs!
Alternatives

• Bash (GNUplot), Python (matplotlib), Windows (Excel), Latex (pgfplots) ...
• No hand-drawn graphs!