FUNCTIONAL MODEL OF A SOFTWARE SYSTEM WITH RANDOM TIME HORIZON

Dmitrii A. Zubok, Aleksandr V. Maiatin, Valentina E. Kiryushkina, Maksim V. Khegai

ITMO University, St.Petersburg, Russia
TABLE OF CONTENTS

• Motivation
• Queuing models
• Experiments in simulation environment
  ● Structure of the experiment
  ● Results
• Experiments in software environment
  ● Architecture of the system
  ● Functional model
  ● Algorithm
  ● Results
• Conclusion
Motivation:

- Software systems can be found on every computer.
- Lately systems that use cloud computing became very popular.
- Optimization techniques are required to keep performance level at its maximum.
Queuing models:

- Infrastructure of the system is built to support use of queuing models.
- Each queuing model uses different service discipline.
- Each service discipline gives different performance level.
Queuing models:

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Produces instances of class <em>Entity</em></td>
</tr>
<tr>
<td>Queue</td>
<td>Describes the queue</td>
</tr>
<tr>
<td>CPU1 &amp; CPU2</td>
<td><em>Delay</em> elements. Delay requests (one per moment) for a specified time</td>
</tr>
</tbody>
</table>
EXPERIMENTS IN SIMULATION ENVIRONMENT:

• Both service disciplines are used.
• AnyLogic Professional 7.1 is used as a simulation environment.
EXPERIMENTS IN SIMULATION ENVIRONMENT: STRUCTURE OF EXPERIMENTS

- Set initial model state (an empty queue, two available servers and no processed requests)
- Set the model parameters: rate and experiment time.
- Run the model 500 times.
- Calculated average time by calculating average service time for each run and then calculating average of average values from previous step.
- Write results to a file.
EXPERIMENTS IN SIMULATION ENVIRONMENT: RESULTS

<table>
<thead>
<tr>
<th>Q (threshold value for queue size)</th>
<th>Threshold average time, seconds</th>
<th>Conservative/Threshold, seconds</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>309,2085</td>
<td>1,2470</td>
</tr>
<tr>
<td>2.0</td>
<td>313,9540</td>
<td>1,2281</td>
</tr>
<tr>
<td>3.0</td>
<td>326,7411</td>
<td>1,1801</td>
</tr>
<tr>
<td>4.0</td>
<td>343,0303</td>
<td>1,1240</td>
</tr>
</tbody>
</table>
EXPERIMENTS IN SOFTWARE ENVIRONMENT:

- To gather data from software environment a system must be built.
- Virtualization is used to simulate different physical servers.
- Xen-hypervizor is used to create and manage virtual machines.
EXPERIMENTS IN SOFTWARE ENVIRONMENT: ARCHITECTURE OF THE SYSTEM
EXPERIMENTS IN SOFTWARE ENVIRONMENT: FUNCTIONAL MODEL
EXPERIMENTS IN SOFTWARE ENVIRONMENT: ALGORITHM

- A request is created by user (or, in case of the presented architecture, it's created in different intervals, generated by random numbers generator)
- This request is added to the common queue on the controlling virtual machine
- Following a service discipline the request is sent to appropriate virtual machine
- The request is processed and an answer is sent back to the controlling virtual machine
EXPERIMENTS IN SOFTWARE ENVIRONMENT: RESULTS (1ST VIRTUAL SERVER, 3 SECONDS INTERVAL)
EXPERIMENTS IN SOFTWARE ENVIRONMENT: RESULTS (1ST VIRTUAL SERVER, 1 SECOND INTERVAL)
EXPERIMENTS IN SOFTWARE ENVIRONMENT: RESULTS (2ND VIRTUAL SERVER, 3 SECONDS INTERVAL)
EXPERIMENTS IN SOFTWARE ENVIRONMENT: RESULTS (2ND VIRTUAL SERVER, 1 SECOND INTERVAL)
Experiments in software environment: Results (Conservative discipline)
EXPERIMENTS IN SOFTWARE ENVIRONMENT: RESULTS (THRESHOLD DISCIPLINE)
**Experiments in Software Environment: Results (Tables with Results)**

<table>
<thead>
<tr>
<th>VM</th>
<th>Interval</th>
<th>Average Processing Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>VM1</td>
<td>3 s</td>
<td>0.251110625</td>
</tr>
<tr>
<td>VM1</td>
<td>1 s</td>
<td>0.258107819</td>
</tr>
<tr>
<td>VM2</td>
<td>3 s</td>
<td>0.435131251</td>
</tr>
<tr>
<td>VM2</td>
<td>1 s</td>
<td>0.436286227</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Discipline</th>
<th>Average Processing Time</th>
<th>Requests left in queue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conservative</td>
<td>2,117474206</td>
<td>15</td>
</tr>
<tr>
<td>Threshold</td>
<td>1,401167783</td>
<td>9</td>
</tr>
</tbody>
</table>
CONCLUSION

• Functional model is built and used to build the system for experiments.
• While using the threshold discipline processing time is lower than while using the conservative discipline.
• While using the threshold discipline number of requests in queue is lower than while using the conservative discipline.
Thank you for your attention

Maksim Khegay,
St. Petersburg, Russia, E-mail: MaxHegai@rambler.ru