Using and Extending LIMBO for Descriptive Modeling of Arrival Behaviors

Symposium on Software Performance

Jóakim v. Kistowski, Nikolas Herbst, Samuel Kounev

Chair for Software-Engineering, Uni Würzburg

28.11.2014
Motivation

Page Requests for the German Wikipedia
Motivation

Page Requests for the German Wikipedia
Page Requests for the German Wikipedia
Additive decomposition into seasonal part, trend, and remainder. Created using BFAST [1].
Problem: No means to effectively capture, reproduce, and modify varying load intensity of real-world cloud systems

Idea: Support load intensity profile description by creating a meta-model and tooling

Benefits: Enable more precise communication and creation of realistic load scenarios for benchmarking

Actions: Creation of meta-models, processes, and tools for load intensity extraction and description
Descartes Load Intensity Model (DLIM)

- Describes arrival rate variations over time
- Provides structure for piece-wise mathematical functions
- Independent of work/request type
Benefits of DLIM:

- Powerful and expressive
- Easy derivation of arrival rates or request time-stamps

Drawbacks of DLIM:

- Instances can become complex
- Large trees may be unintuitive

Solution: high-level DLIM

- Fewer parameters for load intensity profile description
- Strictly structured into single *Seasonal*, *Trend*, recurring *Burst*, and *Noise* parts
Automated process for extracting DLIM or hl-DLIM instances from existing arrival rate traces

- Structured into *Seasonal*, *Trend*, *Burst*, and *Noise* part extraction

- Noise reduction and extraction is optional and separate
LIMBO: EMF-based modeling platform that uses DLIM for load intensity description. It offers a New Model Creation Wizard based on hl-DLIM, which allows the generation of arrival rate and request time-stamp profiles. This can be used in JMeter using the TimestampTimer by Andreas Weber (KIT). LIMBO also visualizes and compares arrival rate profiles and provides automated model instance extraction. For more information, visit the GitHub repository: https://github.com/andreaswe/JMeterTimestampTimer.
Evaluation

Usability Evaluation
- Usability evaluated using a questionnaire
- Users are computer scientists from five different organizations
- Mean Usability (1 = easy, 4 = difficult): 1.44
- Mean Feature Usefulness (1 = useful, 4 = not useful): 1.2

Model Extraction Accuracy Evaluation
- 9 real world web server traces
- Metric: median arrival rate deviation
- S-MIEP and hl-MIEP applied to all traces
- P-MIEP to traces longer than one month
<table>
<thead>
<tr>
<th>Trace</th>
<th>S-MIEP relative median error (%)</th>
<th>BFAST relative median error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ClarkNet-HTTP</td>
<td>12.409</td>
<td>12.243</td>
</tr>
<tr>
<td>NASA-HTTP</td>
<td>18.812</td>
<td>-</td>
</tr>
<tr>
<td>Saskatchewan-HTTP</td>
<td>26.492</td>
<td>-</td>
</tr>
<tr>
<td>WorldCup98</td>
<td>12.979</td>
<td>-</td>
</tr>
<tr>
<td>IBM Transactions</td>
<td>29.199</td>
<td>-</td>
</tr>
<tr>
<td>de.wikipedia.org</td>
<td>8.538</td>
<td>11.223</td>
</tr>
<tr>
<td>fr.wikipedia.org</td>
<td>7.6</td>
<td>8.511</td>
</tr>
<tr>
<td>ru.wikipedia.org</td>
<td>9.912</td>
<td>5.809</td>
</tr>
<tr>
<td>wikipedia.org</td>
<td>4.855</td>
<td>2.302</td>
</tr>
</tbody>
</table>

S-MIEP performs on average **8354** times faster than BFAST
Two Meta-Models for load intensity variation description

- **DLIM**: Powerful and expressive
- **hl-DLIM**: Abstract and concise

Modeling Platform: **LIMBO**

- Enables creation of custom load intensity variations for open workload based benchmarking
- Provides automated load intensity profile extraction

Automated model instance extraction:

- **S-MIEP**: most accurate, median deviation: 12.4%
- **P-MIEP**: good for regular profiles, median deviation: 37.6%
- **hl-MIEP**: relies on noise reduction, median deviation: 15.6%

**LIMBO** is open-source¹ and already being used in different contexts.

¹LIMBO: http://descartes.tools/limbo
Thank you for your Interest!

Our future work on LIMBO:

- Extraction of multiple and overlaying seasonal patterns
- Change detection
- Advanced calibration and noise reduction

Ideas for integration/extension

- Extending Markov4JMeter to use LIMBO timestamps
- Extending PCM and DML to use DLIM instances or LIMBO timestamps
- Using DLIM models for improving anomaly detection accuracy