Austin, TX, USA, 2015-02-02

Using Dynatrace Monitoring Data for Generating Performance Models of Java EE Applications

Tool Paper

International Conference on Performance Engineering (ICPE) 2015

Felix Willnecker¹, Andreas Brunnert¹, Wolfgang Gottesheim², Helmut Krcmar³

¹fortiss GmbH, ²Compuware Austria GmbH, ³Technische Universität München
Motivation & Vision

• Creating performance models requires a lot of expert knowledge
  – The effort required for creating performance models manually leads to a low adoption rate of model-based performance evaluations in practice (Mayer et al. 2011)

• Model-based performance evaluation techniques proposed by the scientific community are often only applicable once a model of a system exists
  – Automated performance model generators help to make them better applicable

• Several approaches were proposed to automatically construct performance models (e.g., Brosig et al. 2009 and Brunnert et al. 2013a)
  – Due to license restrictions or limitations of the available data they often use self-written monitoring solutions
Motivation & Vision

- To use monitoring data collected by industry-standard Application Performance Management (APM) solutions in order to automatically construct performance models

→ Better integration between model- and measurement-based performance evaluation techniques

Tool Architecture

**Previous Work**

- Builds upon an existing performance model generation framework introduced in Brunnert et al. (2013a) and Brunnert et al. (2014b)
- Uses self-written monitoring components (e.g., ServletFilters, EJB Interceptors) in order to collect the required monitoring data
Tool Architecture

Current State – Integration with Dynatrace

• Extends the persistence layer of the model generation framework to use the REST API of Dynatrace in order to collect the required monitoring data.
Application Areas

Use Cases

• Use your existing APM knowledge for model-based Software Performance Engineering (SPE) activities such as:

  – Early Performance Predictions (e.g., when reusing existing components, Brunnert et al. 2013b)

  – Architecture Optimizations (Koziolek et al. 2013)

  – Detecting Performance Changes (Brunnert/Krcmar 2014c)

  – Capacity Planning (Brunnert et al. 2014a)

  – ...

[1] SPE and APM integration cycle taken from the DevOps Performance Working Group Poster presented at SOSP ’14 and ICPE ’15
References


Andreas Brunnert
brunnert@fortiss.org
performancegroup@fortiss.org
pmw.fortiss.org