University of Stuttgart
Institute of Software Technology
Reliable Software Systems

A Unified Model-Driven Approach for Extracting and Generating Workload Specifications for Load Testing and Performance Prediction of Application Systems

Christian Vögele, André van Hoorn, Eike Schulz, Wilhelm Hasselbring, and Helmut Krcmar: *WESSBAS*: Extraction of probabilistic workload specifications for load testing and performance prediction—a model-driven approach for session-based application systems.* Software & Systems Modeling (2016).

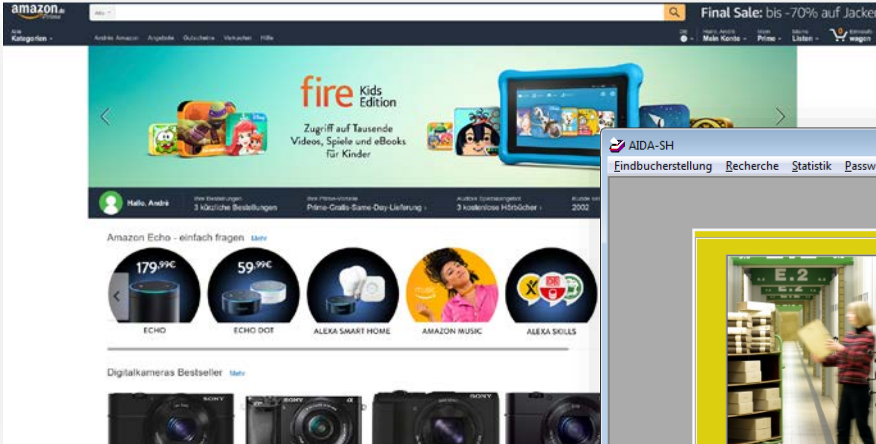
André van Hoorn

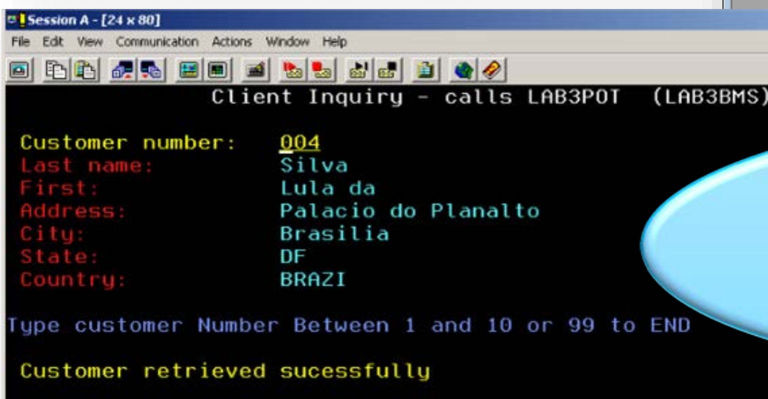
SE 17 SOFTWARE ENGINEERING

February 23, 2017 – Hannover

RSS Reliable Software Systems

Domain – Session-based Application Systems





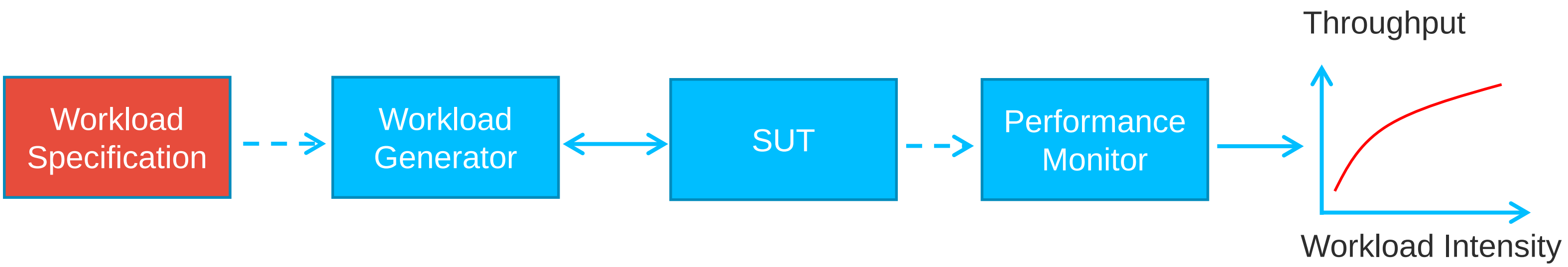
Session:
"A series of consecutive and related requests issued by the same customer"

(*Menascé et al. 1999*)

Problem Statement

Workload specification and execution essential to evaluate performance properties of (session-based) application systems

- Measurement-based approaches (e.g., load testing)
- Model-based approaches (e.g., performance prediction)



Problem Statement

Workload specification and execution essential to evaluate performance properties of (session-based) application systems

- Measurement-based approaches (e.g., load testing)
- Model-based approaches (e.g., performance prediction)

Problems

- Manual creation (and maintenance) of **representative workload specifications** is difficult, time consuming, and error-prone
- **Workload specifications for measurement- and model-based** approaches are modelled separately of each other (M-by-N problem)

WESSBAS* Approach – Overview

Automatic Extraction of WESSBAS-DSL instances

Analysis of request logs

Model transformations

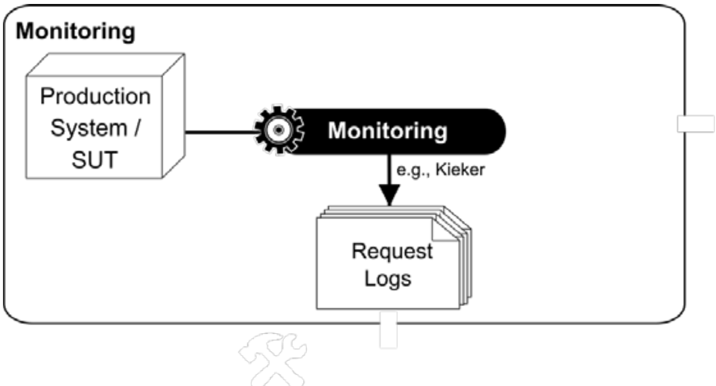
- Load tests scripts
- Workload models for performance prediction

WESSBAS-DSL

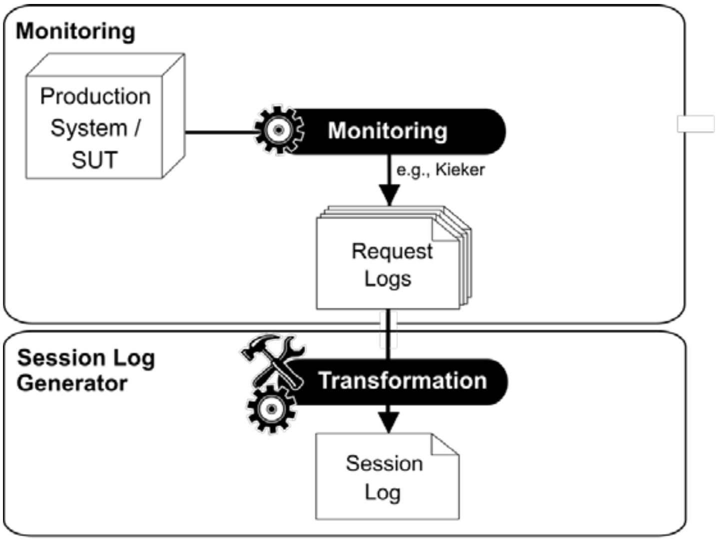
A tool- and system-agnostic (intermediate) modeling language

*WESSBAS is an acronym for
Workload Extraction and Specification for Session-Based Application Systems

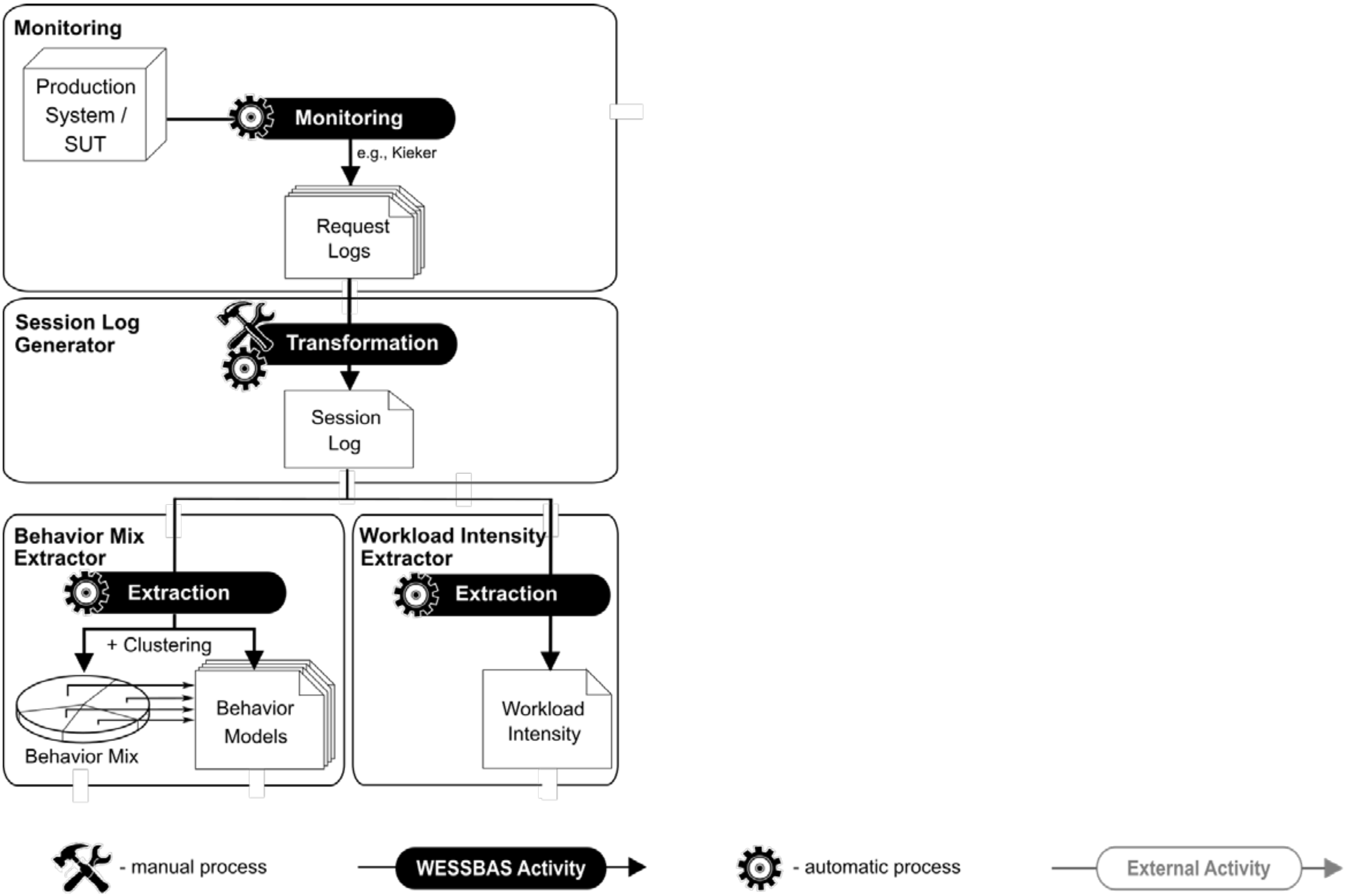
WESSBAS Approach – Overview



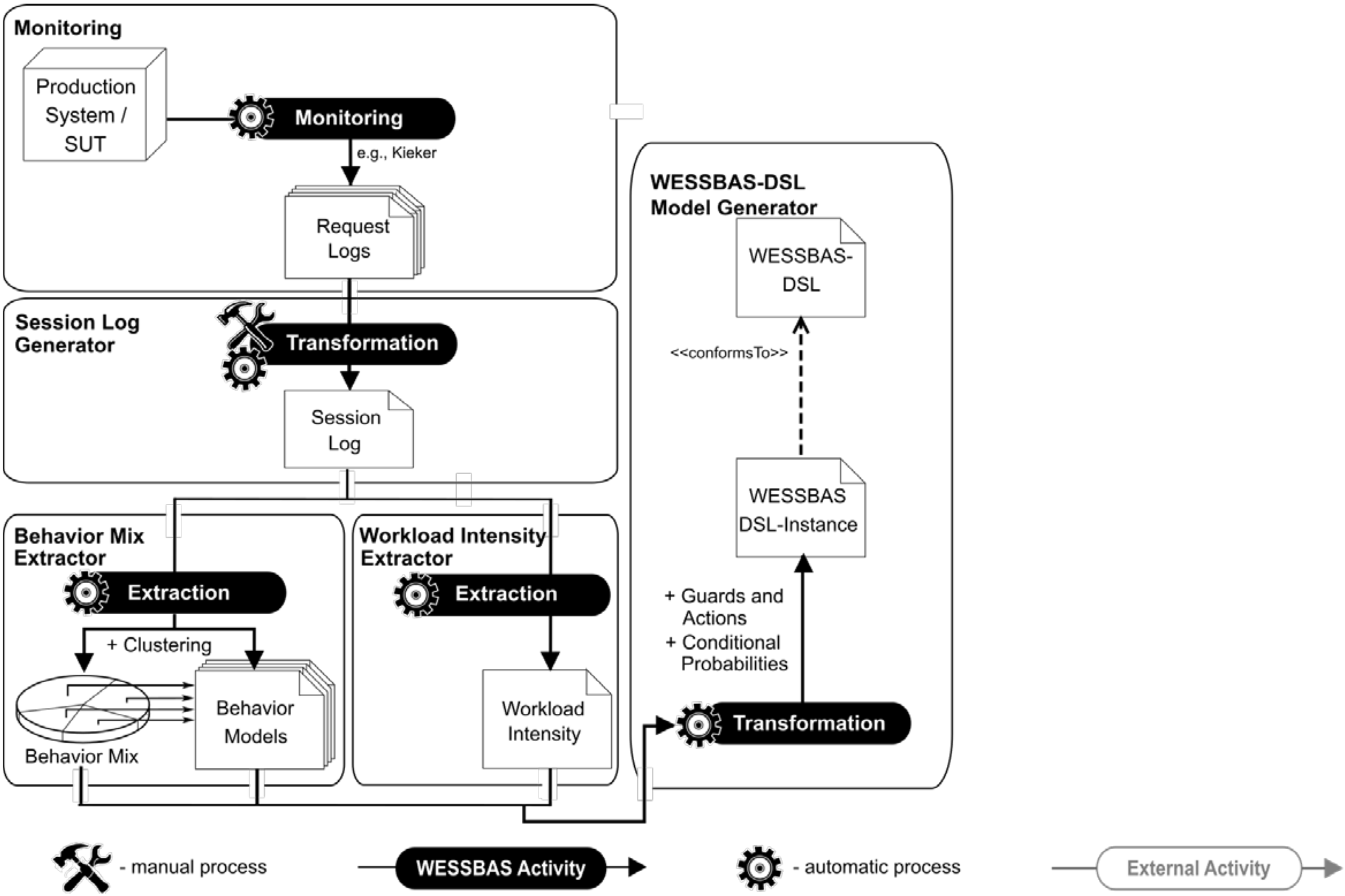
WESSBAS Approach – Overview



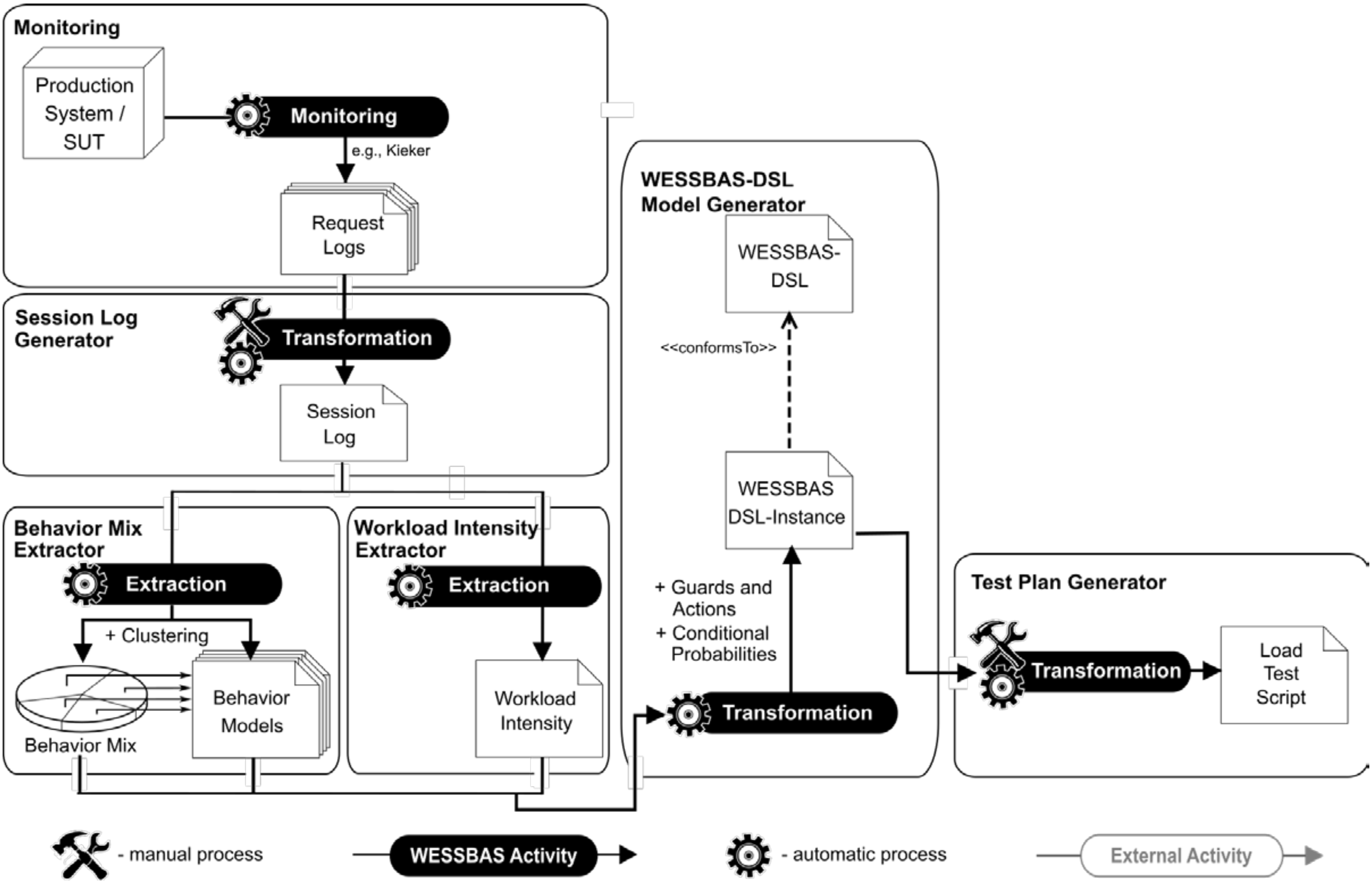
WESSBAS Approach – Overview



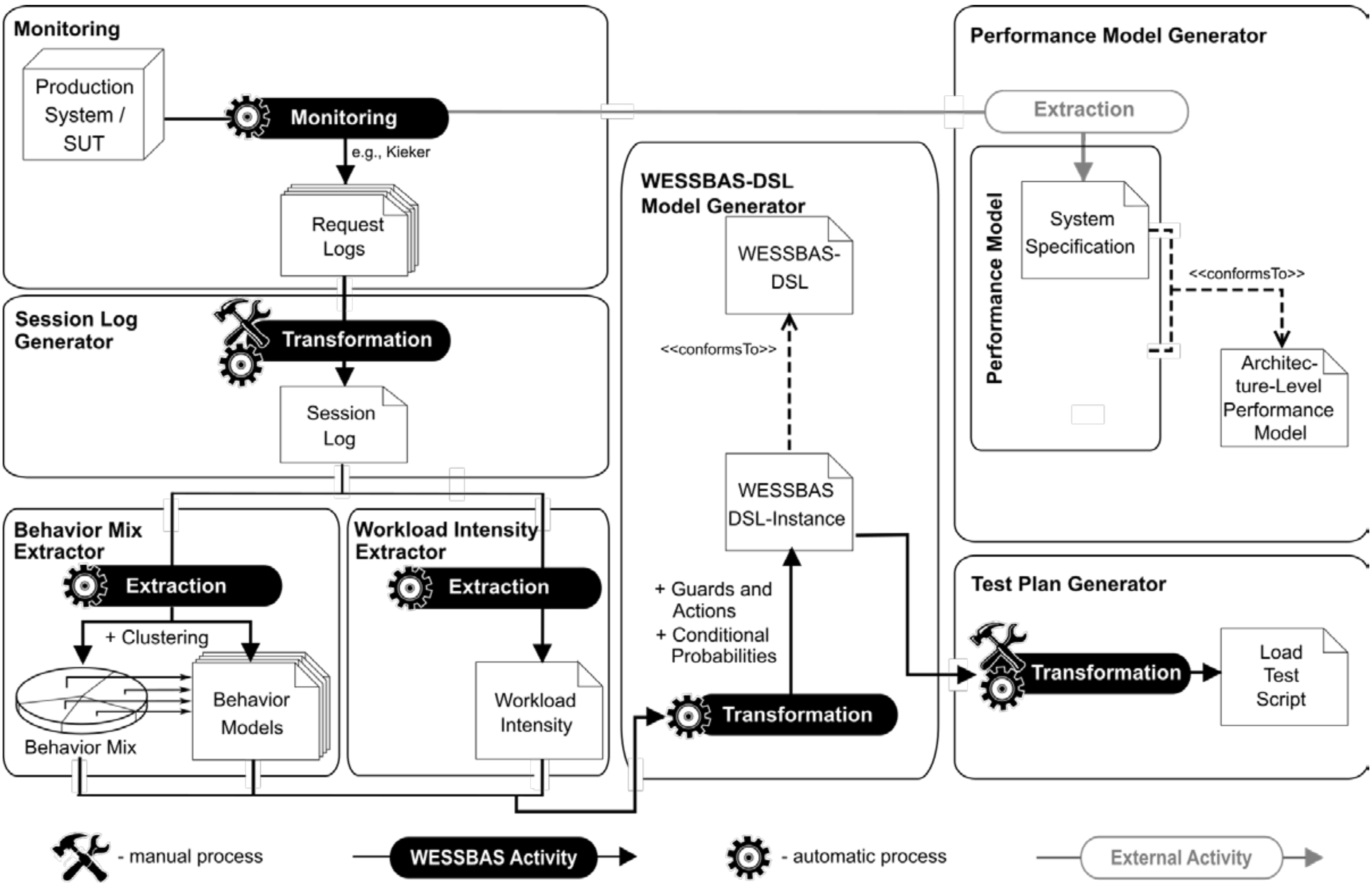
WESSBAS Approach – Overview



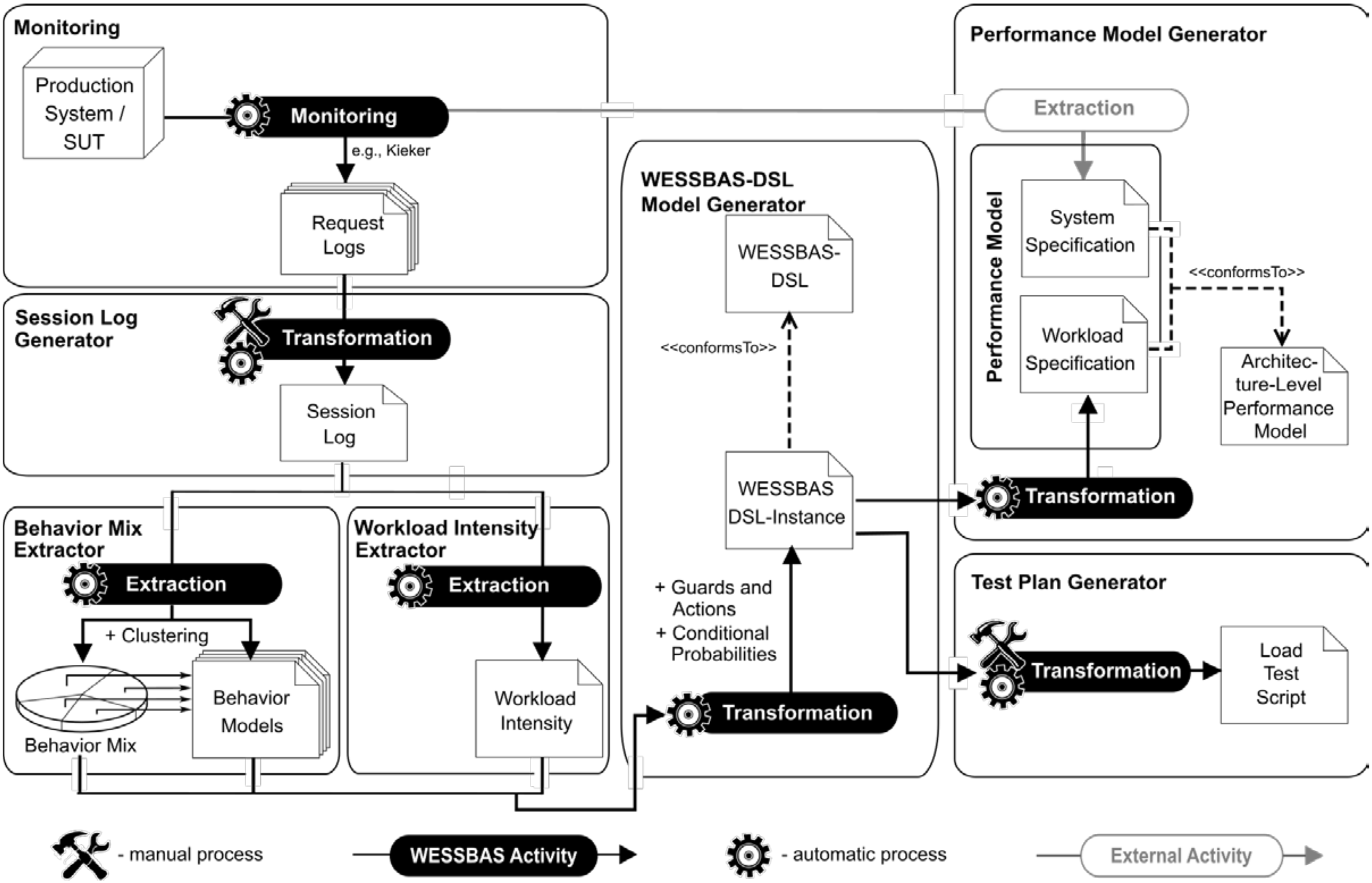
WESSBAS Approach – Overview



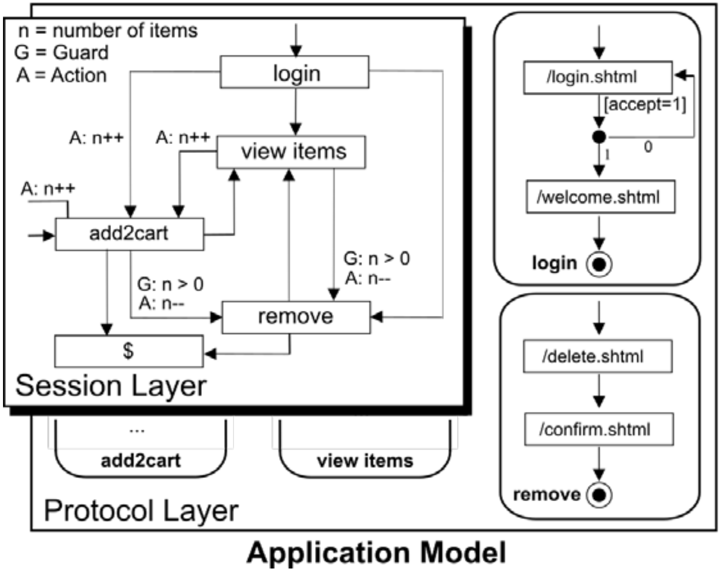
WESSBAS Approach – Overview



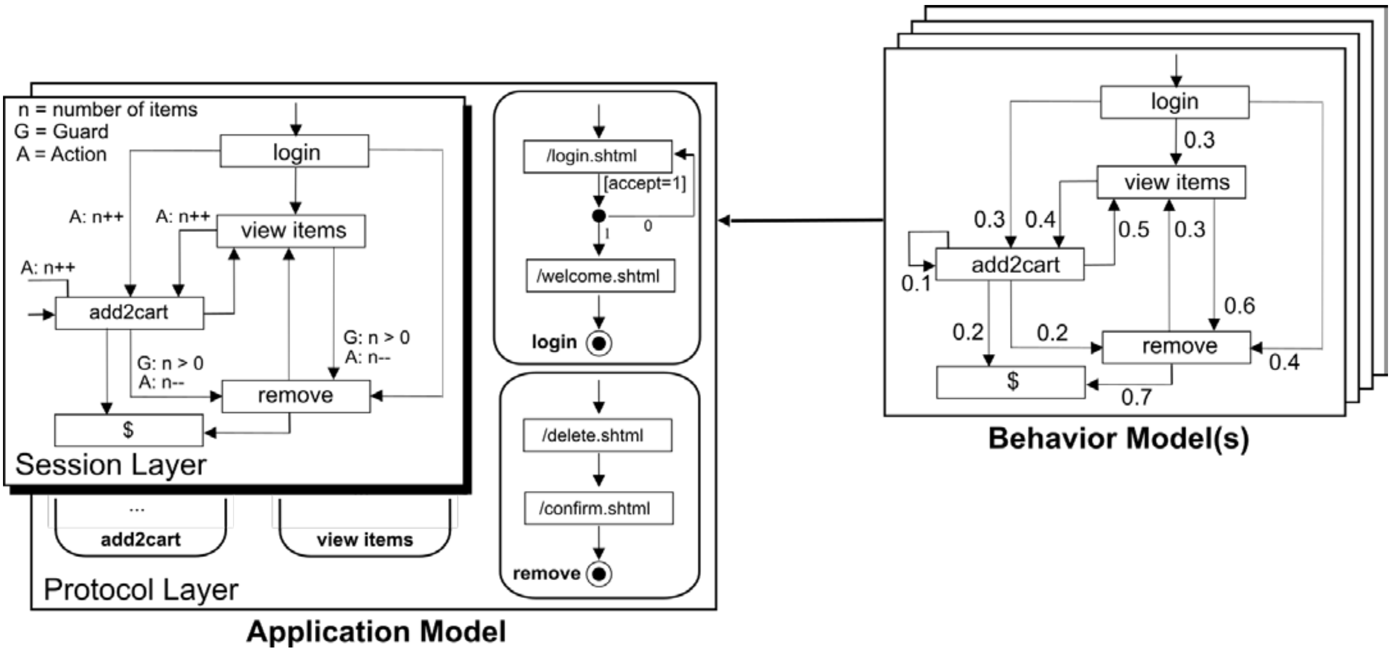
WESSBAS Approach – Overview



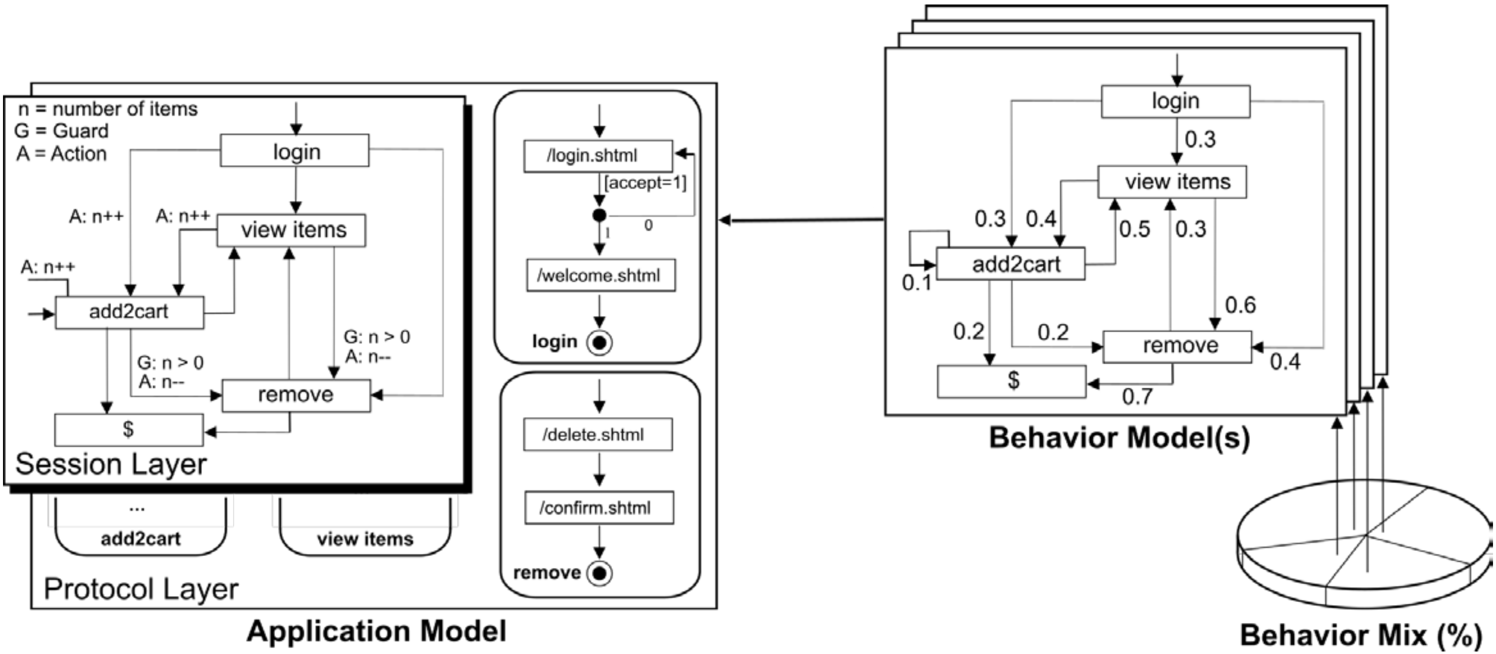
Specifying Session-Based Workloads



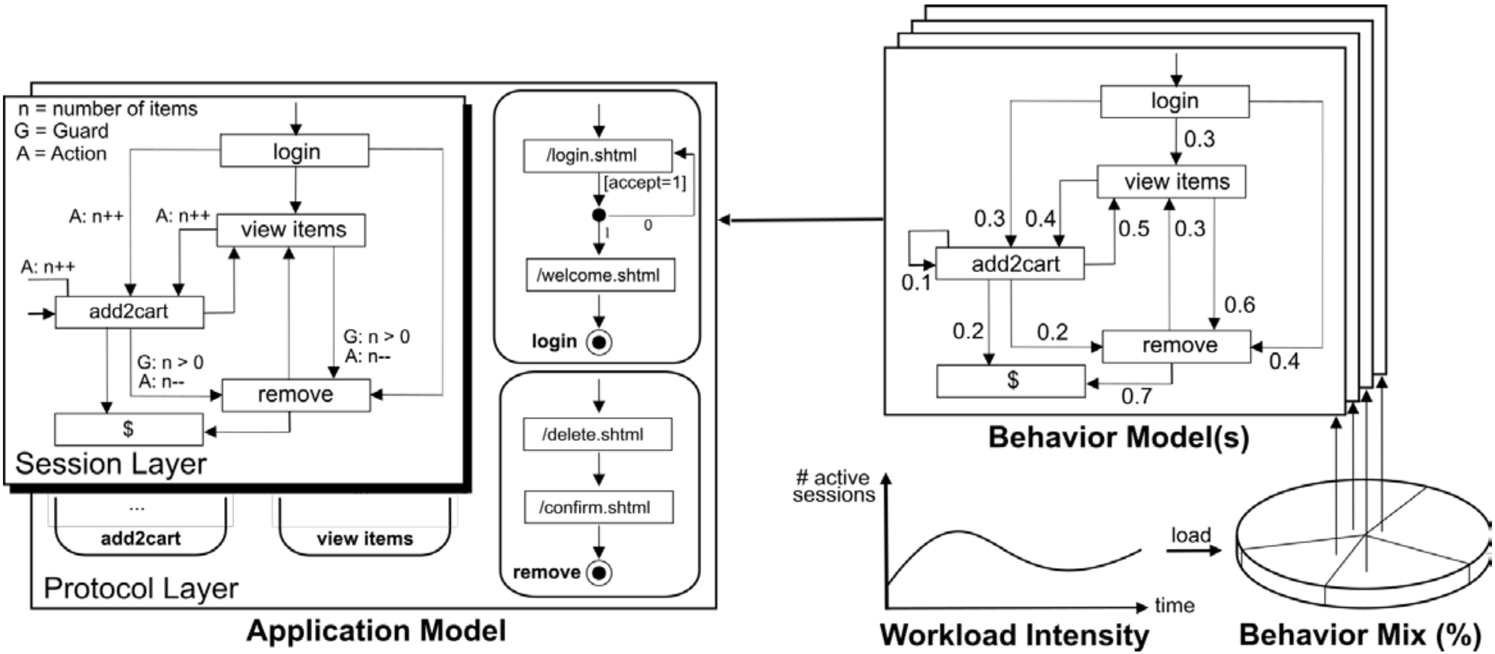
Specifying Session-Based Workloads



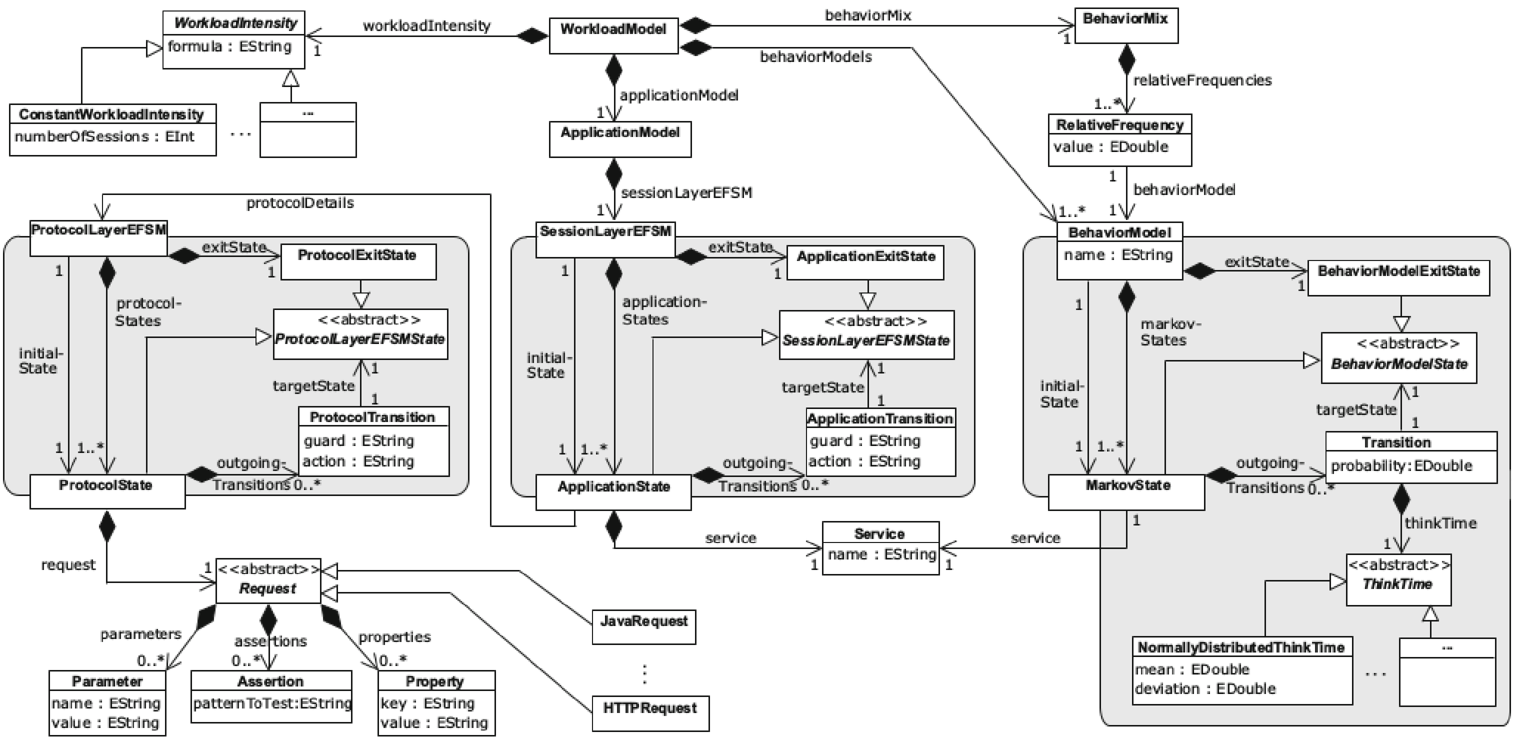
Specifying Session-Based Workloads



Specifying Session-Based Workloads

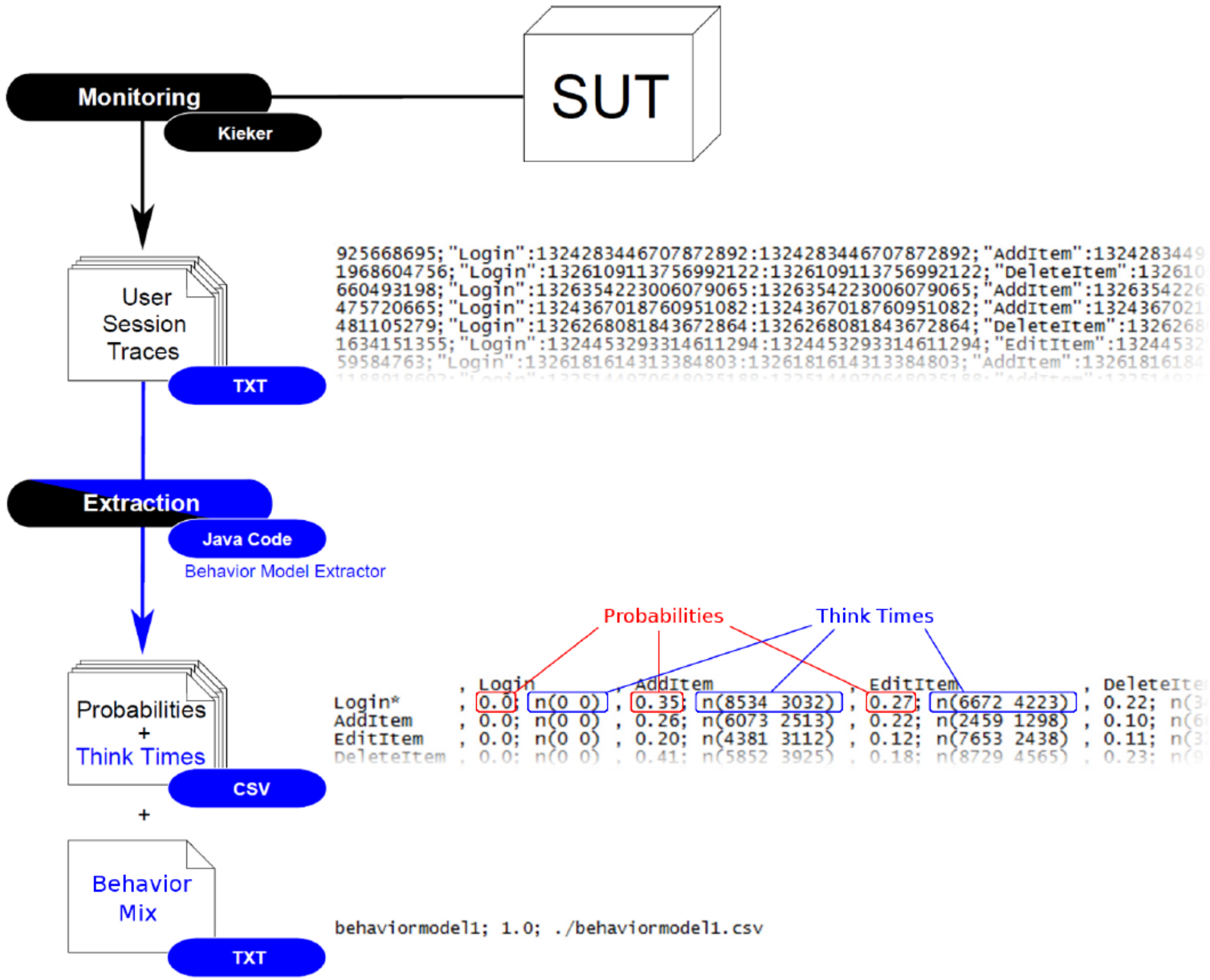


WESSBAS-DSL

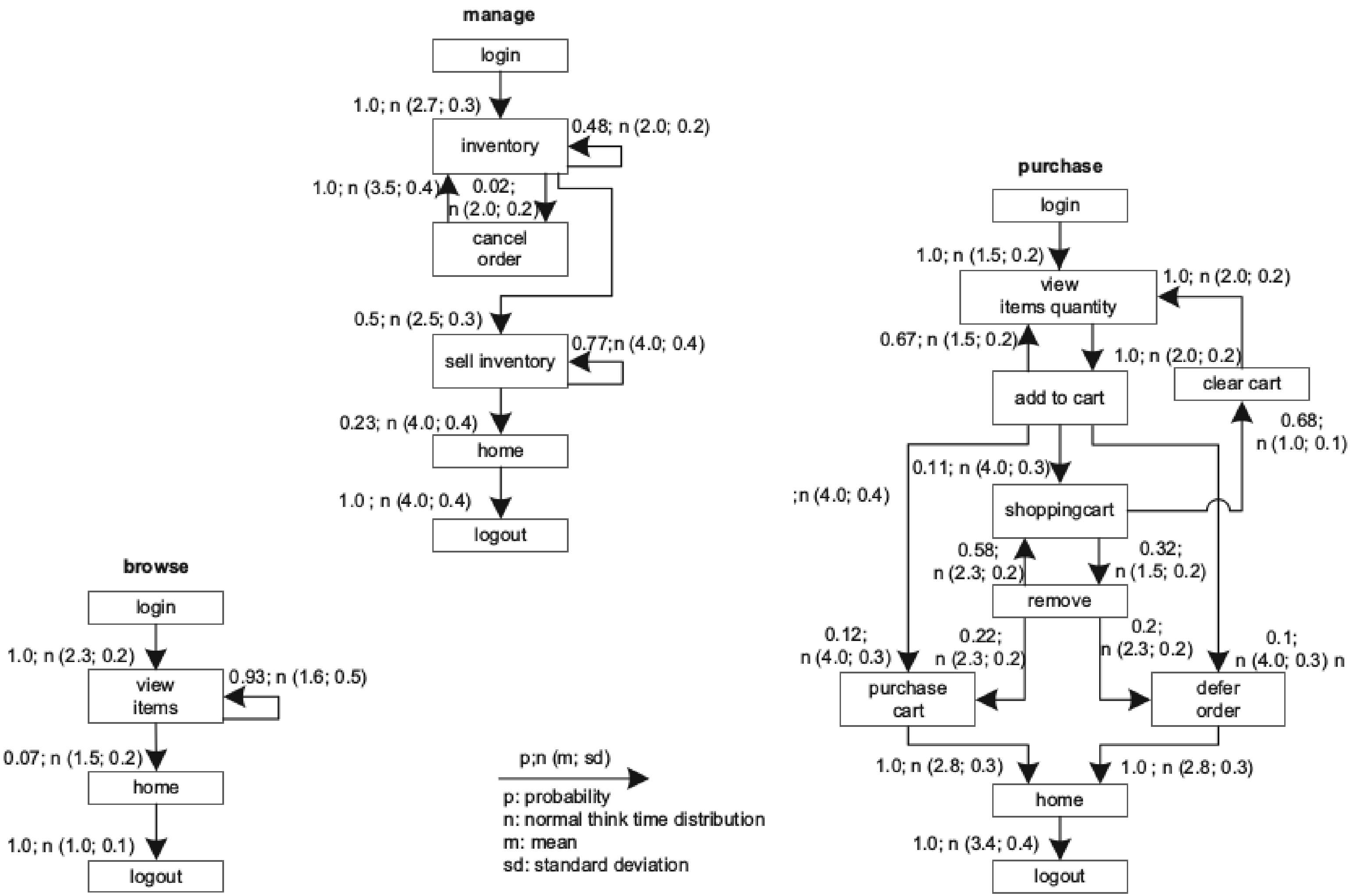


WorkloadIntensity
formula : EString
workloadIntensity
WorkloadModel
behaviorMix
BehaviorMix
behaviorModels
ConstantWorkloadIntensity
numberOfSessions : EInt
applicationModel
ApplicationModel
relativeFrequencies
RelativeFrequency
value : EDouble
sessionLayerEFSM
behaviorModel
protocolDetails
ProtocolLayerEFSM
exitState
ProtocolExitState
protocol-States
<<abstract>>
ProtocolLayerEFSMState
initial-State
targetState
ProtocolTransition
guard : EString
action : EString
outgoing-Transitions
ProtocolState
SessionLayerEFSM
ApplicationExitState
application-States
SessionLayerEFSMState
ApplicationTransition
ApplicationState
BehaviorModel
name : EString
BehaviorModelExitState
markov-States
BehaviorModelState
Transition
probability : EDouble
MarkovState
thinkTime
service
Service
request
Request
JavaRequest
HTTPRequest
ThinkTime
NormallyDistributedThinkTime
mean : EDouble
deviation : EDouble
parameters
assertions
properties
Parameter
name : EString
value : EString
Assertion
patternToTest:EString
Property
key : EString
value : EString

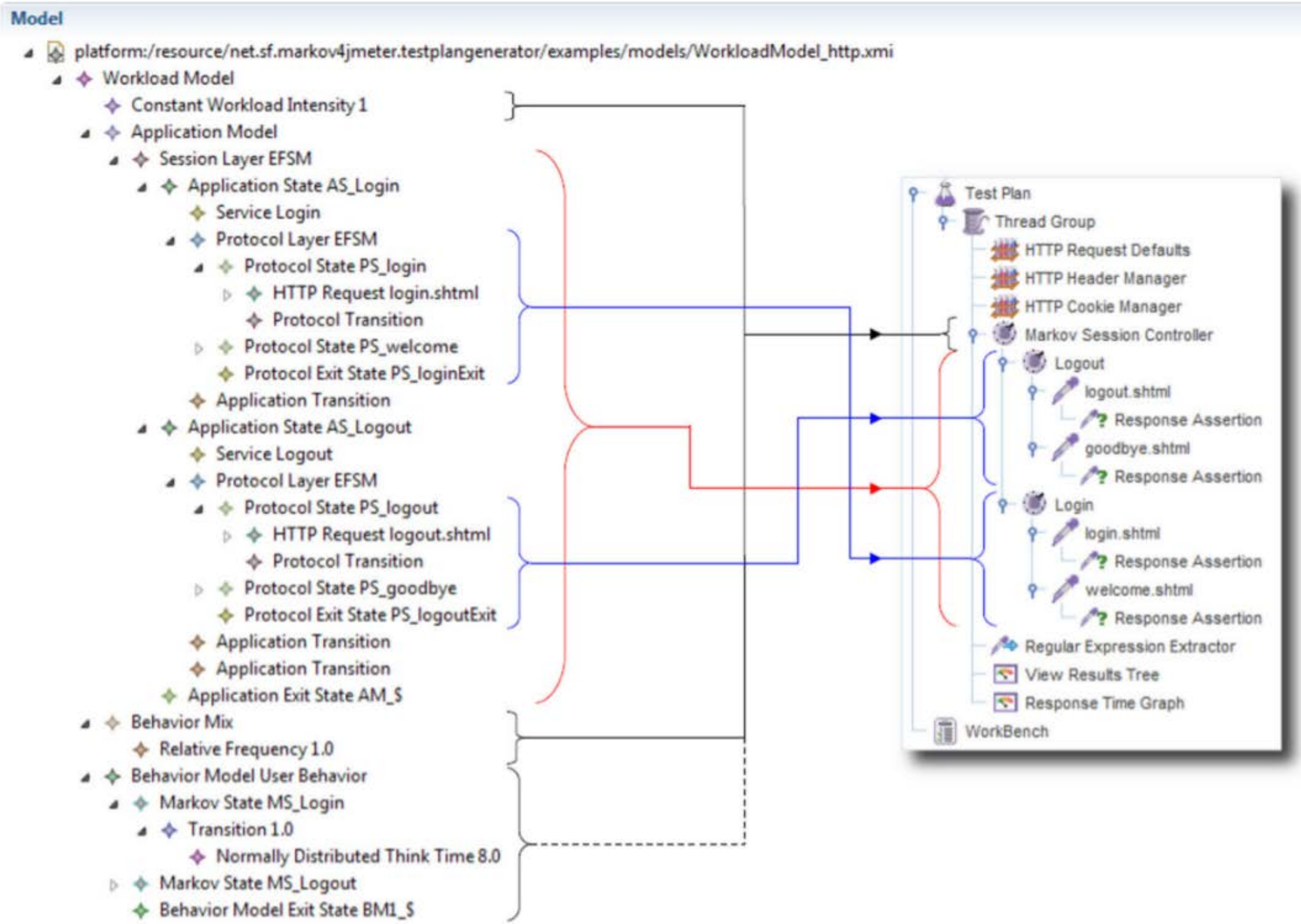
Behavior Model Extraction



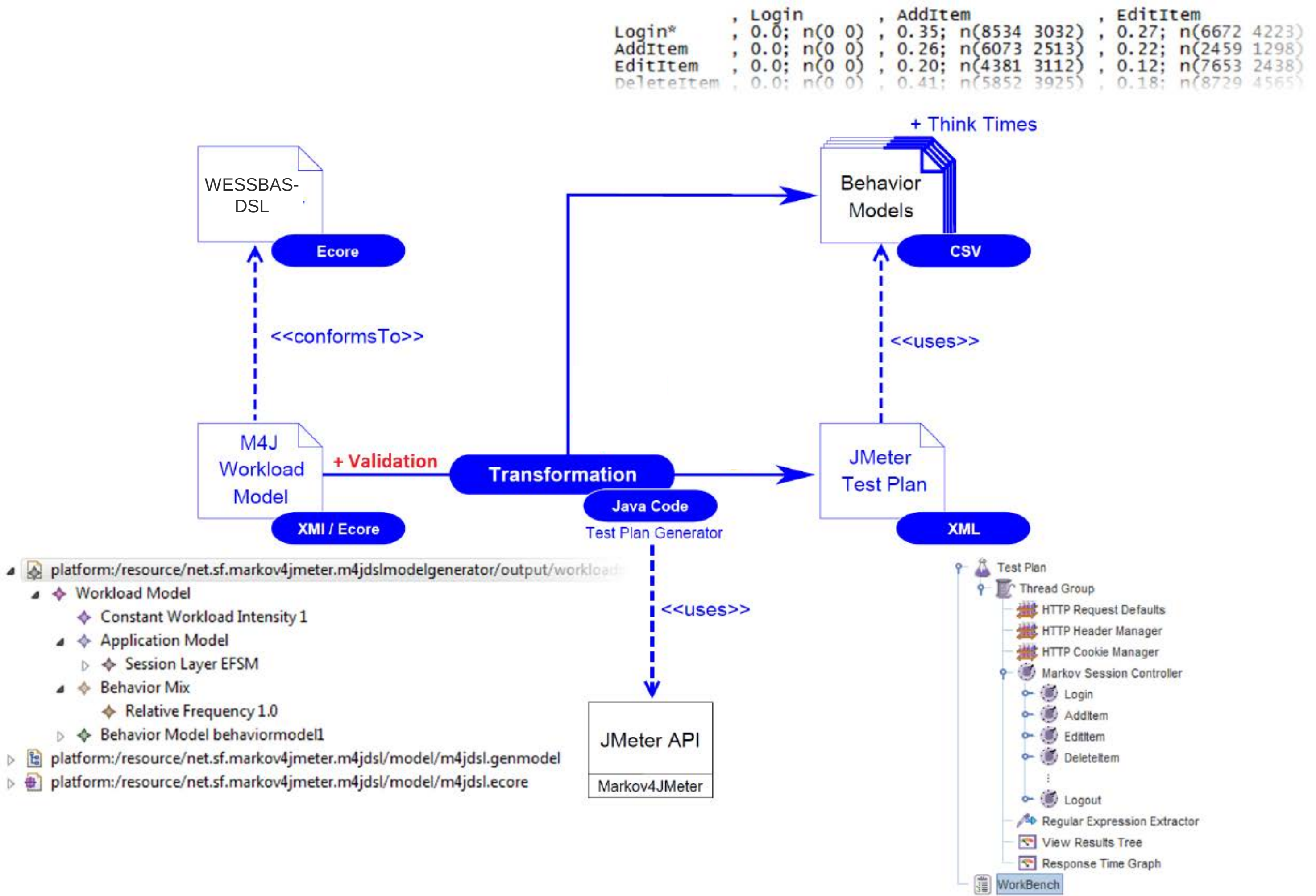
Probabilistic Representation of SPECjEnterprise2010 Workload



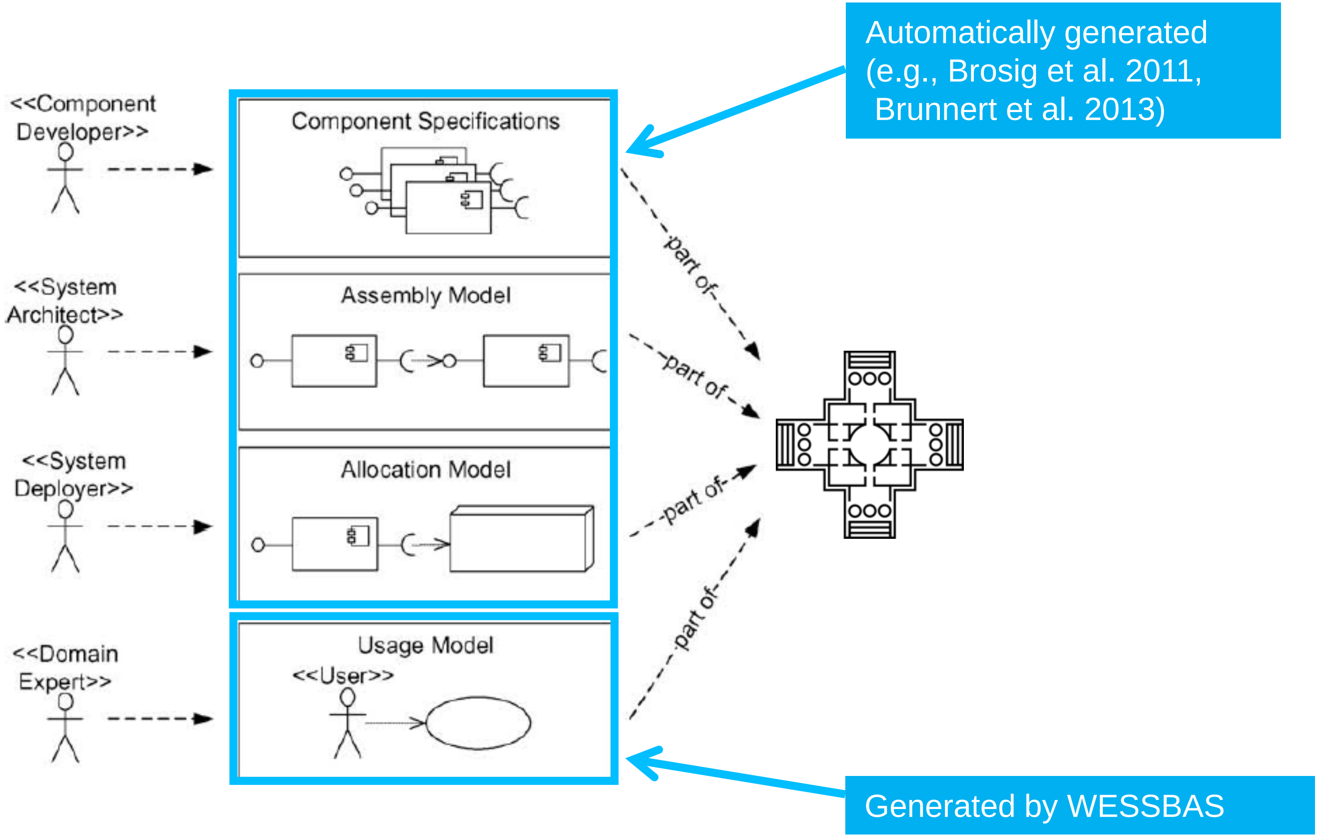
Transformation into Apache JMeter Test Plans



Transformation into Apache JMeter Test Plans

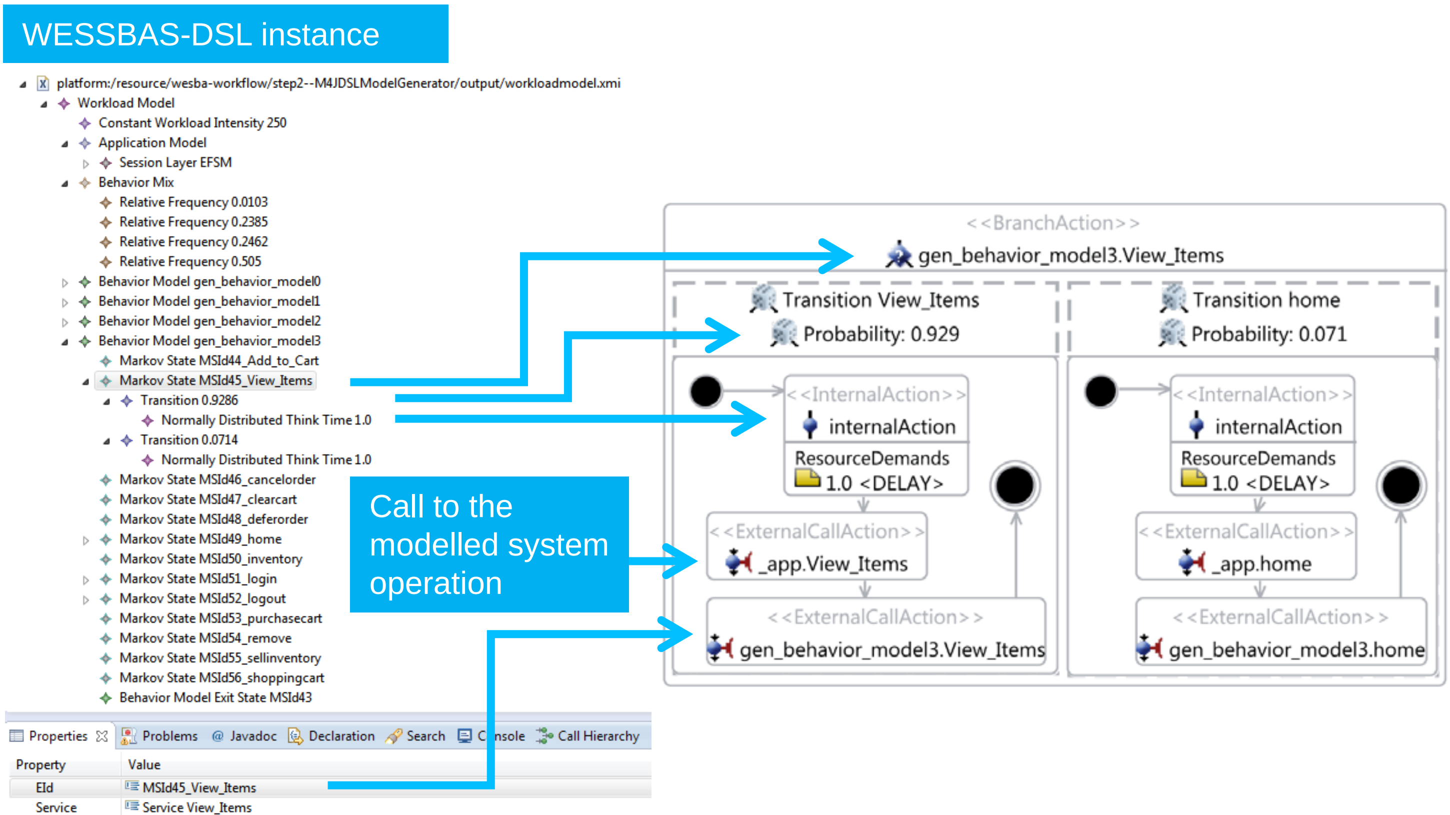


Transformation into Palladio Component Models



Becker et al. (2009)

Transformation into Palladio Component Models



Evaluation Research Questions and Methodology

RQ1: How accurately do the clustering results match the input Behavior Mix?

RQ2: What is the impact of the clustering results on the workload characteristics of the executed and predicted workload?

RQ3: How accurately do the performance characteristics of the production system/SUT match the performance characteristics using the generated and predicted workload?

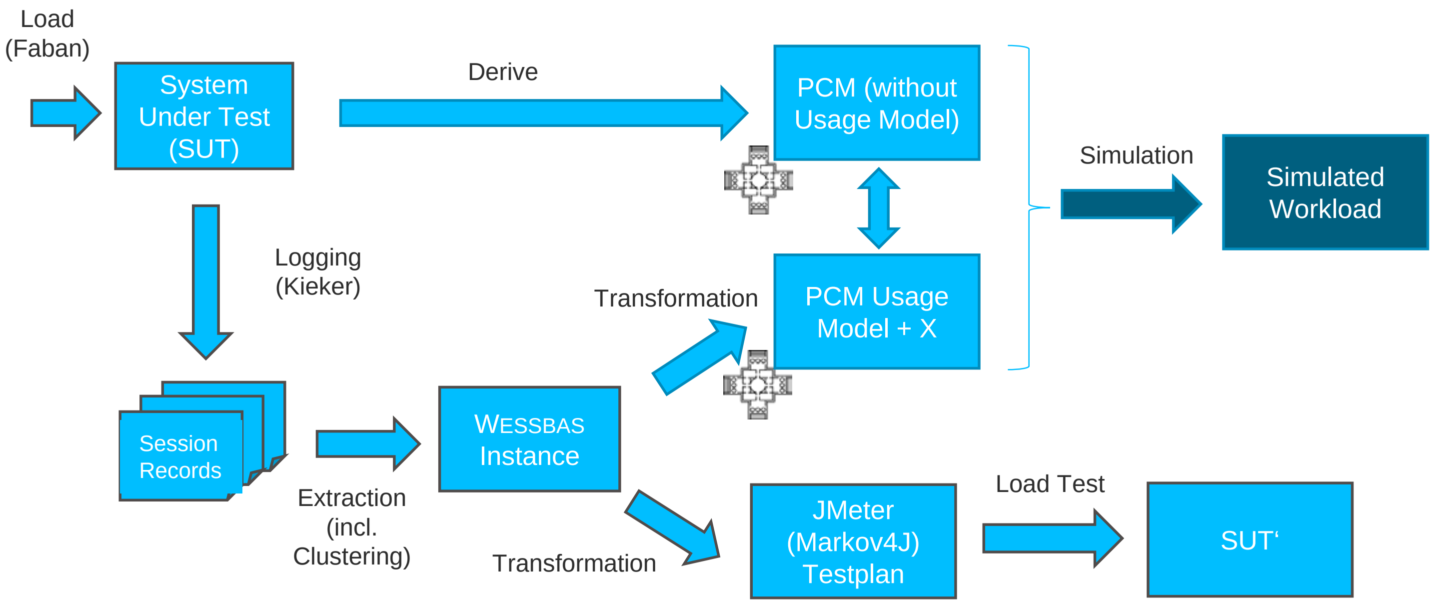
RQ4: How accurately do the workload and performance characteristics match when applying different workload settings to the extracted workload?

RQ5: What is the impact of GaAs on the workload and performance characteristics?

Evaluation – Case Studies

1. **SPECjEnterprise2010**
   - Scenario 0: Generate benchmark load with the Faban harness (+ monitoring)
   - Scenario 1: Workload generation (+ monitoring)
   - Scenario 2: Performance prediction
   - Monitored (and predicted) measures:
     - request and session statistics
     - response times, CPU and memory utilization

SPECjEnterprise2010 Case Study Setting



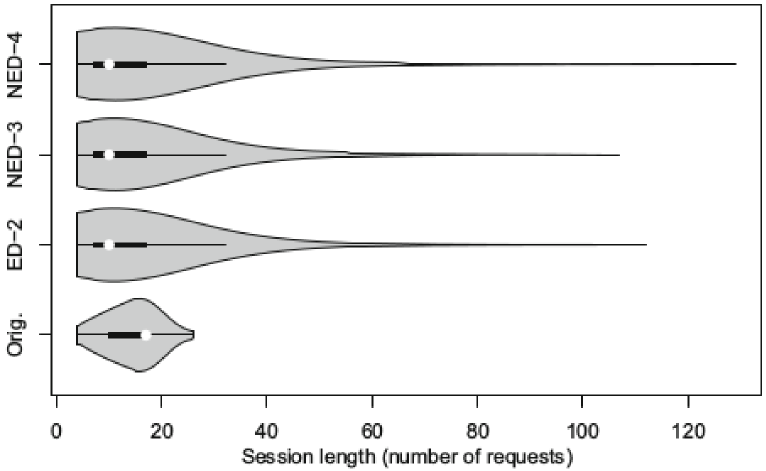
Evaluation – Case Studies

1. **SPECjEnterprise2010**
   - Scenario 0: Generate benchmark load with the Faban harness (+ monitoring)
   - Scenario 1: Workload generation (+ monitoring)
   - Scenario 2: Performance prediction
   - Monitored (and predicted) measures:
     - request and session statistics
     - response times, CPU and memory utilization

2. **FIFA World Cup 1998** web server access logs
   - Scenario 1: Workload generation (+ monitoring)
   - Scenario 2: Performance Prediction
   - Measures: request and session statistics

Selected Results for SPECjEnterprise2010

RQ2: What is the impact of the clustering results on the workload characteristics of the executed and predicted workload?

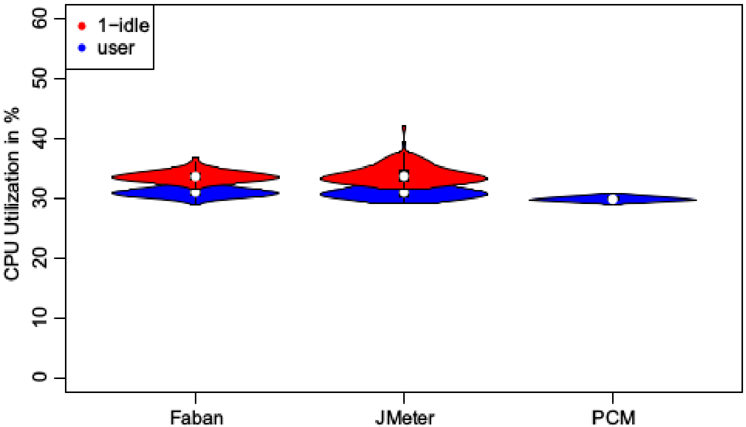


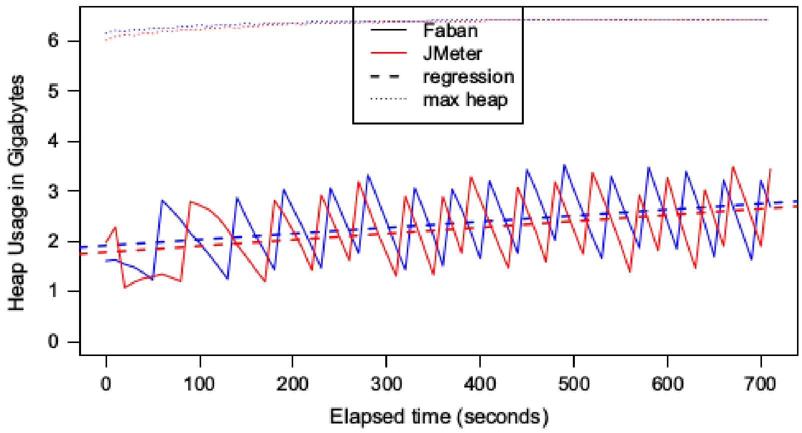
	Min.	Q_1	*Med.*	*Mean*	$CI_{0.95}$	Q_3	*Max.*	*N*
Orig.	4	10	17	14.18	[14.12, 14.24]	17	26	19,890
ED-2	4	7	10	13.96	[13.81, 14.11]	17	112	20,119
NED-3	4	7	10	13.88	[13.73, 14.02]	17	107	20,358
NED-4	4	7	10	13.96	[13.82, 14.11]	17	129	20,299

Request	Orig.	ED-2	NED-3	NED-4	Rel.
(a) Absolute and relative (Rel.) counts (JMeter)					
1. Add to cart	20,625	21,474	21,129	21,217	0.07
2. Cancel order	191	198	176	168	0.00
3. Clear cart	1932	2129	2011	1976	0.01
4. Defer order	2236	2228	2218	2312	0.01
5. Home	19,371	20,119	20,358	20,299	0.07
6. Inventory	10,034	10,273	10,136	10,064	0.04
7. Login	19,890	20,119	20,358	20,299	0.07
8. Logout	19,372	20,119	20,358	20,299	0.07
9. Purchase cart	2682	2780	2873	2795	0.01
10. Remove	923	660	675	723	0.00
11. Sell inventory	21,949	22,703	21,854	21,653	0.08
12. Shopping cart	2855	2789	2686	2699	0.01
13. View items	139,370	133,766	136,529	137,723	0.49
14. View items quantity	20,625	21,474	21,129	21,217	0.07
(b) Absolute and relative (Rel.) counts (PCM)					
1. Add to cart	20,625	22,416	22,466	21,936	0.07
2. Cancel order	191	217	165	208	0.00
3. Clear cart	1932	2094	2222	2062	0.01
4. Defer order	2236	2425	2379	2275	0.01
5. Home	19,371	21,131	21,190	20,990	0.07
6. Inventory	10,034	10,703	10,656	10,932	0.04
7. Login	19,890	21,128	21,190	20,997	0.07
8. Logout	19,372	21,128	21,190	20,997	0.07
9. Purchase cart	2682	2806	2919	2840	0.01
10. Remove	923	711	713	692	0.00
11. Sell inventory	21,949	23,867	23,552	23,807	0.08
12. Shopping cart	2855	2808	2939	2755	0.01
13. View items	139,370	146,637	146,903	148,698	0.49
14. View items quantity	20,625	22,425	22,472	21,930	0.07

Selected Results for SPECjEnterprise2010

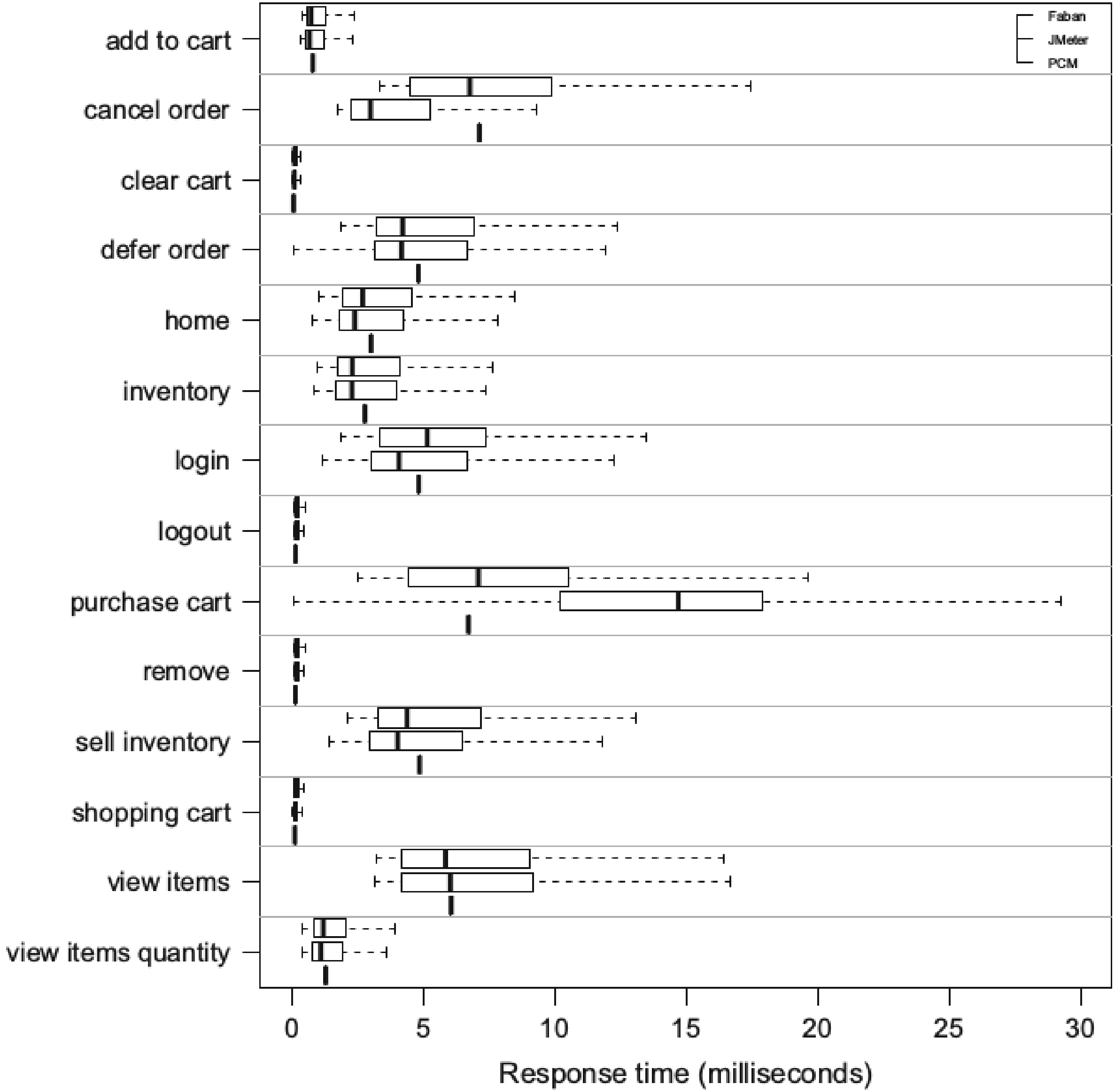
RQ3: How accurately do the performance characteristics of the production system/SUT match the performance characteristics using the generated and predicted workload?





	Mean	$\pm CI_{0.95}$	*Std. dev.*	*Median*	*N*
Faban (1-idle)	33.67	± 0.22	0.92	33.62	72
Faban (user)	31.06	± 0.21	0.89	31.02	72
JMeter (1-idle)	33.99	± 0.38	1.63	33.66	72
JMeter (user)	31.36	± 0.37	1.60	31.01	72
PCM	29.84	± 0.10	0.41	29.80	72

	Mean	$\pm CI_{0.95}$	*Stddev.*	*Median*	*N*
Faban	2.35	± 0.15	0.64	2.35	72
JMeter	2.23	± 0.16	0.66	2.23	72



Faban
JMeter
PCM
add to cart
cancel order
clear cart
defer order
home
inventory
login
logout
purchase cart
remove
sell inventory
shopping cart
view items
view items quantity
0
5
10
15
20
25
30
Response time (milliseconds)

Summary of Results

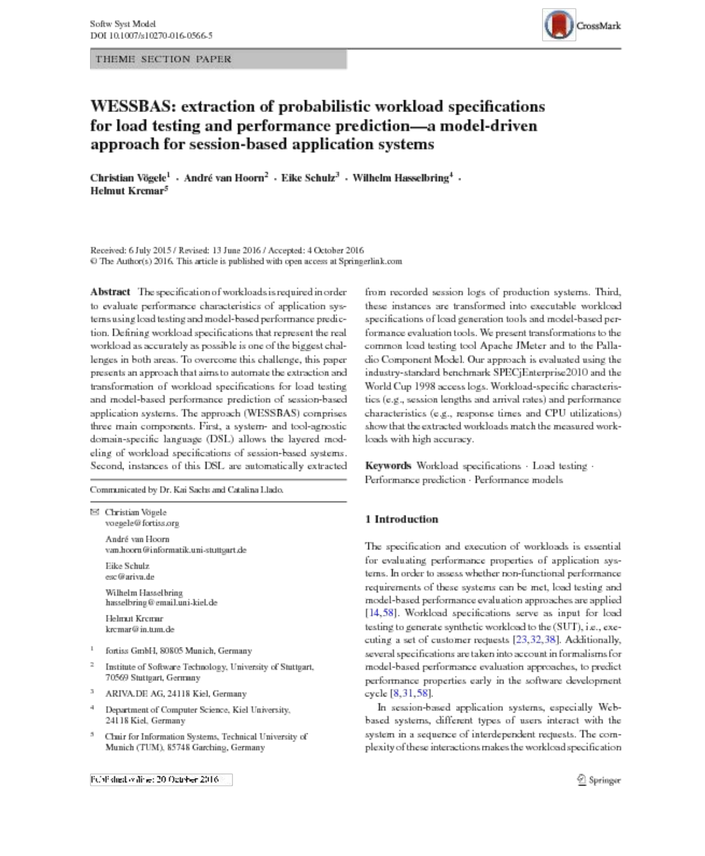
RQ2: What is the impact of the clustering results on the workload characteristics of the executed and predicted workload?

- The session-based characteristics, like session length and the number of distinct sessions, deviate from the measured logs in case of SPECjEnterprise2010.
- The invocation frequencies for requests match with almost 100 %

RQ3: How accurately do the performance characteristics of the production system/SUT match the performance characteristics using the generated and predicted workload?

- Performance characteristics in terms of CPU utilization, response times and heap usage are, with a few minor exceptions similar to the original executed workload.

SoSyM Article for this Talk



Softw Syst Model
DOI 10.1007/s10270-016-0566-5

CrossMark

THEME SECTION PAPER

WESSBAS: extraction of probabilistic workload specifications for load testing and performance prediction—a model-driven approach for session-based application systems

Christian Vögele[1] · André van Hoorn[2] · Eike Schulz[3] · Wilhelm Hasselbring[4] · Helmut Krcmar[5]

Received: 6 July 2015 / Revised: 13 June 2016 / Accepted: 4 October 2016
© The Author(s) 2016. This article is published with open access at Springerlink.com

Abstract The specification of workloads is required in order to evaluate performance characteristics of application systems using load testing and model-based performance prediction. Defining workload specifications that represent the real workload as accurately as possible is one of the biggest challenges in both areas. To overcome this challenge, this paper presents an approach that aims to automate the extraction and transformation of workload specifications for load testing and model-based performance prediction of session-based application systems. The approach (WESSBAS) comprises three main components. First, a system- and tool-agnostic domain-specific language (DSL) allows the layered modeling of workload specifications of session-based systems. Second, instances of this DSL are automatically extracted from recorded session logs of production systems. Third, these instances are transformed into executable workload specifications of load generation tools and model-based performance evaluation tools. We present transformations to the common load testing tool Apache JMeter and to the Palladio Component Model. Our approach is evaluated using the industry-standard benchmark SPECjEnterprise2010 and the World Cup 1998 access logs. Workload-specific characteristics (e.g., session lengths and arrival rates) and performance characteristics (e.g., response times and CPU utilizations) show that the extracted workloads match the measured workloads with high accuracy.

Keywords Workload specifications · Load testing · Performance prediction · Performance models

Communicated by Dr. Kai Sachs and Catalina Llado.

✉ Christian Vögele
voegele@fortiss.org

André van Hoorn
van.hoorn@informatik.uni-stuttgart.de

Eike Schulz
esc@ariva.de

Wilhelm Hasselbring
hasselbring@email.uni-kiel.de

Helmut Krcmar
krcmar@in.tum.de

1 fortiss GmbH, 80805 Munich, Germany
2 Institute of Software Technology, University of Stuttgart, 70569 Stuttgart, Germany
3 ARIVA.DE AG, 24118 Kiel, Germany
4 Department of Computer Science, Kiel University, 24118 Kiel, Germany
5 Chair for Information Systems, Technical University of Munich (TUM), 85748 Garching, Germany

1 Introduction

The specification and execution of workloads is essential for evaluating performance properties of application systems. In order to assess whether non-functional performance requirements of these systems can be met, load testing and model-based performance evaluation approaches are applied [14,58]. Workload specifications serve as input for load testing to generate synthetic workload to the (SUT), i.e., executing a set of customer requests [23,32,38]. Additionally, several specifications are taken into account in formalisms for model-based performance evaluation approaches, to predict performance properties early in the software development cycle [8,31,58].

In session-based application systems, especially Web-based systems, different types of users interact with the system in a sequence of interdependent requests. The complexity of these interactions makes the workload specification

Published online: 20 October 2016

Christian Vögele, André van Hoorn, Eike Schulz, Wilhelm Hasselbring, and Helmut Krcmar: *WESSBAS*: extraction of probabilistic workload specifications for load testing and performance prediction—a model-driven approach for session-based application systems.* Software & Systems Modeling (2016).

Data: http://dx.doi.org/10.5281/zenodo.54859

Software: http://wessbas.github.io/

*WESSBAS is an acronym for Workload Extraction and Specification for Session-Based Application Systems

Future Work

- Extensions
  - Support for workload intensity
  - Inclusion of input data
  - Additional transformations
    - to alternative workload generators
    - to other architecture-level performance models
    - from PCM to WESSBAS-DSL
- Online clustering to detect evolution of behavior mix
- Co-evolution of manually created and generated parts in the specification
- Applications
  - Load test selection and prioritization (in continuous SE)
  - Performance regression testing and diagnosis (in continuous SE)
  - Model-driven software modernization/evolution
- Industrial case study with Sonatype (Nexus)

References

- M. F. Arlitt, D. Krishnamurthy, and J. Rolia. Characterizing the scalability of a large web-based shopping system. ACM TOIT, 1(1):44-69, 2001.
- A. Ciancone, A. Filieri, M. Drago, R. Mirandola, and V. Grassi. Klapersuite: An integrated model-driven environment for reliability and performance analysis of component-based systems. In Objects, Models, Components, Patterns, volume 6705 of LNCS, pages 99-114. Springer Berlin Heidelberg, 2011.
- A. van Hoorn, M. Rohr, and W. Hasselbring. Generating probabilistic and intensity-varying workload for Web-based software systems. In Proc. SIPEW '08, pages 124-143, 2008.
- A. van Hoorn, C. Vögele, E. Schulz, W. Hasselbring, and H. Krcmar. Automatic extraction of probabilistic workload specifications for load testing session-based application systems. VALUETOOLS, 2014.
- A. van Hoorn, J. Waller, W. Hasselbring: Kieker: A Framework for Application Performance Monitoring and Dynamic Software Analysis. In Proc. ACM/SPEC ICPE '12, ACM, 2012, 247-248
- J. v. Kistowski, N. R. Herbst, and S. Kounev. Modeling variations in load intensity over time. In Proc. LT '14, pages 1-4. ACM, 2014.
- D. Krishnamurthy, J. A. Rolia, and S. Majumdar. A synthetic workload generation technique for stress testing session-based systems. IEEE TSE, 32(11):868-882, 2006.
- D. A. Menascé, V. A. F. Almeida, R. Fonseca, and M. A. Mendes. A methodology for workload characterization of e-commerce sites. In Proc. EC '99, pages 119-128, 1999.
- D. Pelleg, A. W. Moore. X-means: Extending K-means with ecient estimation of the number of clusters. In ICML '00, pages 727-734, 2000.
- M. Shams, D. Krishnamurthy, and B. Far. A model-based approach for testing the performance of web applications. In Proc. SOQUA 2006, pages 54-61. ACM, 2006.
- C. Smith and C. Llado. Performance model interchange format (pmif 2.0): Xml defintition and implementation. In Proc. QEST '04, pages 38-47, Sept 2004.
- C. Vögele, A. Brunnert, A. Danciu, D. Tertilt, H. Krcmar: Using performance models to support load testing in a large SOA environment. In Proc. LT '15, pages 5-6, 2014
- M. Woodside, D. C. Petriu, D. B. Petriu, H. Shen, T. Israr, and J. Merseguer. Performance by unified model analysis (puma). In Proc. WOSP '05, pages 1-12, New York, NY, USA, 2005. ACM.