Chapter 39 Data Mining System Execution Traces to Validate Distributed System Qualityof-Service Properties

James H. Hill Indiana University-Purdue University Indianapolis, USA

ABSTRACT

System Execution Modeling (SEM) tools enable distributed system testers to validate Quality-of-Service (QoS) properties, such as end-to-end response time, throughput, and scalability, during early phases of the software lifecycle. Analytical capabilities of QoS properties, however, are traditionally bounded by a SEM tool's capabilities. This chapter discusses how to mine system execution traces, which are a collection of log messages describing events and states of a distributed system throughout its execution lifetime, generated by distributed systems so that the validation of QoS properties is not dependent on a SEM tool's capabilities. The author uses a real-life case study to illustrate how data mining system execution traces.

INTRODUCTION

Enterprise distributed systems, such as mission avionic systems, traffic management systems, and shipboard computing environments, are transitioning to next-generation middleware, such as service-oriented middleware (Pezzini & Natis, 2007) and component-based software engineering

DOI: 10.4018/978-1-4666-2455-9.ch039

(Heineman & Councill, 2001). Although nextgeneration middleware is improving enterprise distributed system functional properties (i.e., its operational scenarios), Quality-of-Service (QoS) properties (e.g., end-to-end response time, throughput, and scalability) are not validated until late in the software lifecycle, i.e., during system integration time. This is due in part to the *serialized-phasing development* problem (Rittel & Webber, 1973). As illustrated in Figure 1, in serialized-phasing development, the infrastructure and applicationlevel system entities, such as components that encapsulate common services, are developed during different phases of the software lifecycle. Software design decisions that affect QoS properties, however, are typically not discovered until final stages of development, e.g., at system integration time, which is too late in the software lifecycle to resolve performance bottlenecks in an efficient and cost effective manner (Mann, 1996; Snow & Keil, 2001; Woodside, Franks, & Petriu, 2007).

System Execution Modeling (SEM) tools (Smith & Williams, 2001), which are a form of modeldriven engineering (Schmidt, 2006), assist distributed system developers in overcoming the serialized-phasing development problem shown in Figure 1. SEM tools use domain-specific modeling languages (Ledeczi, Maroti, Karsai, & Nordstrom, 1999) to capture both platform-independent attributes (such as structural and behavioral concerns of the system) and platform-specific attributes (such as the target architecture of the system) as high-level models. Model interpreters then transform constructed models into source code for the target architecture. This enables distributed system testers to validate QoS properties continuously throughout the software lifecycle while the "real" system is still under development. Likewise, as development of the real system is complete, distributed system testers can incrementally replace *faux* portions of the system with its real counterpart to produce more realistic QoS validation results.

Although SEM tools enable distributed system developers and testers to validate distributed system QoS properties during early phases of the software lifecycle, QoS validation capabilities are typically bounded to a SEM tool's analytical capabilities. In order to validate QoS properties unknown to a SEM tool, distributed system testers have the following options:

• Use handcrafted solutions: This option typically occurs outside of the SEM. Moreover, this option is traditionally not applicable across different application do-

System infrastructure error recover effector components planner planner developed first effector sensor configuration Application components developed only Level of Abstraction Domain Laver after infrastructure is sufficiently mature urce Pool Laver Complexities System infrastructure cannot be tested ce Layer adequately until applications are done Entire system must be deployed & configured properly to meet QoS requirements Existing evaluation tools do not support "what if" evaluation Target Infrastructure Development Timeline

Figure 1. Overview of serialized-phased development in distributed systems

22 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/data-mining-system-execution-traces/73469

Related Content

Predicting Future Customers via Ensembling Gradually Expanded Trees

Yang Yu, De-Chuan Zhan, Xu-Ying Liu, Ming Liand Zhi-Hua Zhou (2007). *International Journal of Data Warehousing and Mining (pp. 12-21).* www.irma-international.org/article/predicting-future-customers-via-ensembling/1781

Business Process Improvement through Data Mining Techniques: An Experimental Approach

Loukas K. Tsironis (2016). *Big Data: Concepts, Methodologies, Tools, and Applications (pp. 1737-1754).* www.irma-international.org/chapter/business-process-improvement-through-data-mining-techniques/150240

A Survey of Spatio-Temporal Data Warehousing

Leticia Gómez, Bart Kuijpers, Bart Moelansand Alejandro Vaisman (2009). *International Journal of Data Warehousing and Mining (pp. 28-55).* www.irma-international.org/article/survey-spatio-temporal-data-warehousing/3895

Web Usage Mining for Ontology Management

Brigitte Trousse, Marie-Aude Aufaure, Bénédicte Le Grand, Yves Lechevallierand Florent Masseglia (2009). *Handbook of Research on Text and Web Mining Technologies (pp. 418-447).* www.irma-international.org/chapter/web-usage-mining-ontology-management/21739

A Novel Filter-Wrapper Algorithm on Intuitionistic Fuzzy Set for Attribute Reduction From Decision Tables

Thang Truong Nguyen, Nguyen Long Giang, Dai Thanh Tran, Trung Tuan Nguyen, Huy Quang Nguyen, Anh Viet Phamand Thi Duc Vu (2021). *International Journal of Data Warehousing and Mining (pp. 67-100).* www.irma-international.org/article/a-novel-filter-wrapper-algorithm-on-intuitionistic-fuzzy-set-for-attribute-reduction-fromdecision-tables/290271