Analysis of the Trade-offs in Different Modeling Approaches for Performance Prediction of Software Systems

Samuel Kounev[1], Fabian Brosig[2], Philipp Meier[3], Steffen Becker[4], Anne Koziolek[5], Heiko Koziolek[6], and Piotr Rygielski[1]

Abstract: A number of performance modeling approaches for predicting the performance of modern software systems and IT infrastructures exist in the literature. Different approaches differ in their modeling expressiveness and accuracy, on the one hand, and their modeling overhead and costs, on the other hand. Considering a representative set of established approaches, we analyze the semantic gaps between them as well as the trade-offs in using them; we further provide guidelines for selecting the right approach suitable for a given scenario.

Keywords: Modeling, performance prediction, software systems, model transformation

During the last decade, researchers have proposed a number of modeling approaches and respective model-to-model transformations enabling performance prediction of software systems, including their computing, storage and network infrastructures [GMS07]. These transformations map performance-annotated software architecture models into stochastic models solved by analytical means or by simulation. However, so far, a detailed quantitative evaluation of the accuracy and efficiency of different modeling approaches and solution techniques is missing, making it hard to select an adequate transformation for a given context [Ba04].

Approaches based on numerical solvers are known to be fast but often limited in expressiveness to adequately model many realistic situations. Approaches based on simulation are known to be more expressive but often have long execution times leading to high prediction overhead [BHK14]. The intuitively perceived trade-offs between prediction accuracy and solution efficiency in state-of-the-art performance analysis tools are currently not well understood due to the lack of in-depth quantitative evaluations and comparisons. Trade-off decisions between prediction accuracy and time-to-result are important in scenarios where: (a) a large problem space needs to be explored (e.g., scaling of complex cloud applications [Sp15]) or (b) when the prediction results need to be available within a certain time window (e.g., in trigger-based reactive cloud scaling scenarios [HKR13]).

[1] Department of Computer Science, University of Würzburg Am Hubland, 97074 Würzburg. E-mail: {samuel.kounev, piotr.rygielski}@uni-wuerzburg.de
[2] MiNODES GmbH, Friedrichstrae 224, 10969 Berlin, German. E-mail: fabian.brosig@minodes.com
[3] codecentric AG, Elsenheimerstr. 55a, 80687 München, Germany. E-mail: philstyler@googlemail.com
[4] TU Cheminitz, Straße der Nationen 62, 09111 Chemnitz, Germany. E-mail: steffen.becker@tu-chemnitz.de
[5] Karlsruhe Institute of Technology (KIT), Am Fasanengarten 5, 76131 Karlsruhe, Germany. E-mail: koziolek@kit.edu
[6] ABB Corp. Research, Wallstadter Str. 59, 68526 Ladenburg, Germany. E-mail: heiko.koziolek@de.abb.com

We provide an in-depth comparison and quantitative evaluation of the trade-offs in different model transformations for performance evaluation of software systems and IT infrastructures. The semantic gaps between typical source model abstractions and the different performance analysis techniques are examined in detail. The accuracy and efficiency of each transformation are evaluated by considering several case studies representing systems of different size and complexity.

The presented results and insights gained from the evaluation help software architects and performance engineers to select the appropriate transformation for a given context, thus significantly improving the usability of model transformations for performance prediction.

We provide an overview of the results of a recent paper published in [Br15], as well as some follow-up work at SIMUTools 2015 [RKTG15] focussing on data center networks [RKZ13]. For networks, similarly to software systems, there exist multiple performance modeling and solution approaches, so similar trade-offs exist in the selection of a suitable approach for a given scenario.

References

[Ba04] Balsamo, Simonetta; Di Marco, Antinisca; Inverardi, Paola; Simeoni, Marta: Model-Based Performance Prediction in Software Development: A Survey. IEEE Trans. on Software Engineering, 30(5), May 2004.

[BHK14] Brosig, Fabian; Huber, Nikolaus; Kounev, Samuel: Architecture-Level Software Performance Abstractions for Online Performance Prediction. Elsevier Science of Computer Programming Journal (SciCo), Vol. 90, Part B:71–92, September 2014.

[Br15] Brosig, Fabian; Meier, Philipp; Becker, Steffen; Koziolek, Anne; Koziolek, Heiko; Kounev, Samuel: Quantitative Evaluation of Model-Driven Performance Analysis and Simulation of Component-based Architectures. IEEE Transactions on Software Engineering (TSE), 41(2):157–175, February 2015.

[GMS07] Grassi, Vincenzo; Mirandola, Raffaela; Sabetta, Antonino: Filling the Gap Between Design and Performance/Reliability Models of Component-based Systems. J. Syst. Softw., 80(4):528–558, April 2007.

[HKR13] Herbst, Nikolas Roman; Kounev, Samuel; Reussner, Ralf: Elasticity in Cloud Computing: What it is, and What it is Not. In: Proceedings of the 10th International Conference on Autonomic Computing (ICAC 2013). USENIX, June 2013.

[RKTG15] Rygielski, Piotr; Kounev, Samuel; Tran-Gia, Phuoc: Flexible Performance Prediction of Data Center Networks using Automatically Generated Simulation Models. In: Proceedings of the Eighth EAI International Conference on Simulation Tools and Techniques (SIMUTools 2015). August 2015.

[RKZ13] Rygielski, Piotr; Kounev, Samuel; Zschaler, Steffen: Model-Based Throughput Prediction in Data Center Networks. In: Proceedings of the 2nd IEEE International Workshop on Measurements and Networking (M&N 2013). pp. 167–172, October 2013.

[Sp15] Spinner, Simon; Herbst, Nikolas; Kounev, Samuel; Zhu, Xiaoyun; Lu, Lei; Uysal, Mustafa; Griffith, Rean: Proactive Memory Scaling of Virtualized Applications. In: Proceedings of the 2015 IEEE 8th International Conference on Cloud Computing (IEEE CLOUD 2015). IEEE, pp. 277–284, June 2015.