# The Descartes Modeling Language for Self-Aware Performance and Resource Management

Samuel Kounev, Fabian Brosig, Nikolaus Huber

Department of Computer Science, University of Würzburg
Am Hubland, 97074 Würzburg
{samuel.kounev,fabian.brosig,nikolaus.huber}@uni-wuerzburg.de

**Abstract:** The Descartes Modeling Language (DML) is a novel architecture-level language for modeling performance and resource management related aspects of modern dynamic software systems and IT infrastructures. Technically, DML is comprised of several sub-languages, each of them specified using OMG's Meta-Object Facility (MOF) and referred to as meta-model in OMG's terminology. The various sub-languages can be used both in offline and online settings for application scenarios like system sizing, capacity planning and trade-off analysis, as well as for self-aware resource management during operation.

Modern software systems have increasingly distributed architectures composed of loosely-coupled services that are typically deployed on virtualized infrastructures. Such system architectures provide increased flexibility by abstracting from the physical infrastructure, which can be leveraged to improve system efficiency. However, these benefits come at the cost of higher system complexity and dynamics. The inherent semantic gap between application-level metrics, on the one hand, and resource allocations at the physical and virtual layers, on the other hand, significantly increase the complexity of managing end-to-end application performance.

To address this challenge, techniques for *online performance prediction* are needed. Such techniques should make it possible to continuously predict at runtime: a) changes in the application workloads [HHKA14], b) the effect of such changes on the system performance, and c) the expected impact of system adaptation actions [BHK14]. Online performance prediction can be leveraged to design systems that *proactively* adapt to changing operating conditions, thus enabling what we refer to as *self-aware*[1] performance and resource management [KBH14, HvHK+14, KBHR10]. Existing approaches to performance and resource management in the research community are mostly based on coarse-grained performance models that typically abstract systems and applications at a high level, e.g., [JHJ+10, ZCS07, CAAS07]. Such models do not explicitly model the software architecture and execution environment, distinguishing performance-relevant behavior at the virtualization level vs. at the level of applications hosted inside the running VMs. Thus, their online prediction capabilities are limited and do not support complex scenarios such as, for example, predicting how changes in application workloads propagate through the

[1] Self-awareness is understood as adopted for Dagstuhl Seminar 15041 (`http://www.dagstuhl.de/15041`)

layers and tiers of the system architecture down to the physical resource layer, or predicting the effect on the response times of different services, if a VM in a given application tier is to be replicated or migrated to another host, possibly of a different type.

To enable online performance prediction in scenarios such as the above, *architecture-level* modeling techniques are needed, specifically designed for use in *online* settings. We present a new architecture-level language, called Descartes Modeling Language (DML)[2], which provides appropriate modeling abstractions to describe the resource landscape, the application architecture, the adaptation space, and the adaptation processes of a software system and its IT infrastructure [BHK14, HvHK+14]. We present an overview of the different constituent parts of DML and describe how they can be leveraged to enable online performance prediction and proactive model-based system adaptation. The complete DML specification is available as a technical report [KBH14]. A set of related tools and libraries are available from the DML website at `http://descartes.tools/dml`. Finally, we present some exemplary results from an industrial case study showing the applicability of our approach in a real-life setting [HvHK+14].

## References


[BHK14] F. Brosig, N. Huber, and S. Kounev. Architecture-Level Software Performance Abstractions for Online Performance Prediction. *Elsevier Science of Computer Programming Journal (SciCo)*, Vol. 90, Part B:71–92, 2014.

[CAAS07] I. Cunha, J Almeida, V. Almeida, and M. Santos. Self-Adaptive Capacity Management for Multi-Tier Virtualized Environments. In *IFIP/IEEE Int. Symposium on Integrated Network Management*, pages 129–138, 2007.

[HHKA14] N. Herbst, N. Huber, S. Kounev, and E. Amrehn. Self-Adaptive Workload Classification and Forecasting for Proactive Resource Provisioning. *Concurrency and Computation - Practice and Experience, John Wiley and Sons*, 26(12):2053–2078, 2014.

[HvHK+14] N. Huber, A. van Hoorn, A. Koziolek, F. Brosig, and S. Kounev. Modeling Run-Time Adaptation at the System Architecture Level in Dynamic Service-Oriented Environments. *Service Oriented Computing and Applications Journal*, 8(1):73–89, 2014.

[JHJ+10] Gueyoung Jung, M.A. Hiltunen, K.R. Joshi, R.D. Schlichting, and C. Pu. Mistral: Dynamically Managing Power, Performance, and Adaptation Cost in Cloud Infrastructures. In *IEEE Int. Conf. on Distributed Computing Systems*, pages 62 –73, 2010.

[KBH14] S. Kounev, F. Brosig, and N. Huber. The Descartes Modeling Language. Technical report, Department of Computer Science, University of Wuerzburg, October 2014. http://nbn-resolving.org/urn:nbn:de:bvb:20-opus-104887.

[KBHR10] S. Kounev, F. Brosig, N. Huber, and R. Reussner. Towards self-aware performance and resource management in modern service-oriented systems. In *7th IEEE International Conference on Services Computing (SCC 2010)*, 2010.

[ZCS07] Qi Zhang, Ludmila Cherkasova, and Evgenia Smirni. A Regression-Based Analytic Model for Dynamic Resource Provisioning of Multi-Tier Applications. In *Proceedings of the 4th International Conference on Autonomic Computing*, 2007.


[2] `http://descartes.tools/dml`