



HAL
open science

Early Timing Analysis based on Scenario Requirements and Platform Models (Extended Abstract)

Jörg Holtmann, Julien Deantoni, Markus Fockel

► **To cite this version:**

Jörg Holtmann, Julien Deantoni, Markus Fockel. Early Timing Analysis based on Scenario Requirements and Platform Models (Extended Abstract). Software Engineering 2023, Feb 2023, Paderborn, Germany. pp.69-70. hal-04259779

HAL Id: hal-04259779

<https://inria.hal.science/hal-04259779>

Submitted on 26 Oct 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - ShareAlike 4.0 International License

Early Timing Analysis based on Scenario Requirements and Platform Models (Extended Abstract)

Jörg Holtmann,¹ Julien Deantoni,² Markus Fockel³

Abstract: This extended abstract summarizes our article [HDF22], published in the Journal of Software and Systems Modeling and presented as Journal First paper at MODELS'22.

Keywords: scenario-based requirements; platform modeling; real-time systems; timing analysis

Distributed, software-intensive systems must fulfill communication requirements under hard real-time constraints. The requirements have to be documented and validated carefully using a systematic requirements engineering (RE) approach, for example, by applying scenario-based requirements notations. The resources of the execution platforms and their properties induce effects on the timing behavior, which may lead to violations of the real-time requirements.

Such violations of the real-time requirements can occur for various reasons: The ECUs executing the software have restricted resources that increase execution times; the buses and wireless communication media have restricted resources increasing transmission times; the preemption induced by scheduling policies increase response times, et cetera. More generally, the various properties of the particular resources of the execution platform (*resource properties*) impact the timing behavior by inducing *timing effects* (i.e., delays) during the provision of the actual functionality.

Nowadays, the platform resource properties and their induced timing effects are verified against the real-time requirements by means of timing analysis techniques mostly implemented in commercial-off-the-shelf tools. However, such timing analyses are conducted in late development phases since they rely on artifacts produced during these phases.

For enabling timing analyses already in the early RE phase, related work provides means to specify and analyze timed behavioral models (typically relying on scenario- or automata-based notations), thereby abstracting from the final platform-specific artifacts. However, such approaches typically require to reenact and pre-calculate the timing effects induced by the resource properties and to specify them as part of the timed behavioral models.

Thus, in order to enable early timing analyses already during RE and to relieve the timing analysts from the burden to pre-calculate and to specify the timing effects induced by the

¹ Chalmers | University of Gothenburg, Interaction Design & Software Engineering, Sweden jorg.holtmann@gu.se

² Universite Cote d'Azur, I3S/INRIA Kairos, France julien.deantoni@univ-cotedazur.fr

³ Fraunhofer IEM, Safe and Secure IoT Systems, Germany markus.fockel@iem.fraunhofer.de

platform resource properties as part of behavioral models, we extend a scenario-based requirements notation with allocation means to platform models and define operational semantics for the purpose of simulation-based, platform-aware timing analyses. Figure 1 sketches an overview of the approach and of the contributions.

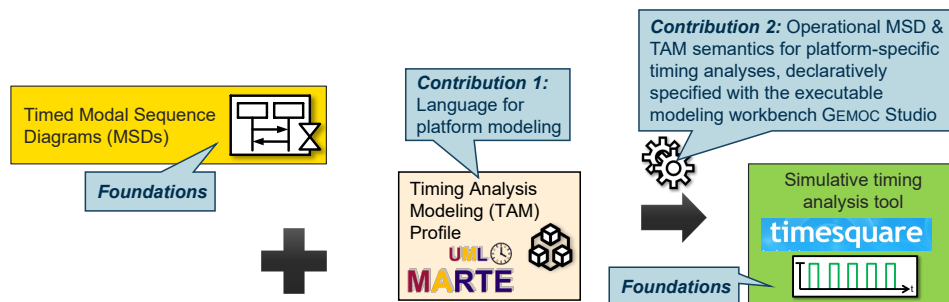


Fig. 1: Overview of the Approach and Contributions

Since the targeted real-time software-intensive systems strongly rely on message-based communication, we base the real-time requirements on our timed and component-based dialect of the scenario-based notation of Modal Sequence Diagrams (MSDs) (cf. left-hand side of Figure 1). Like the related work, the modeling and analysis means provided by our dialect enable specifying and validating real-time requirements but incorporate platform-specific aspects only insufficiently. To provide both an abstract specification of the execution platform with its particular resource properties and the allocation of MSD specifications to the execution platform, we furthermore extend platform modeling concepts of the real-time modeling UML profile MARTE (cf. middle of Figure 1). Based on the modeling languages mentioned above, we introduce as main contribution a new operational semantics for platform-aware MSDs dedicated to timing analyses (cf. arrow in the middle of Figure 1). This semantics encompasses an extended MSD message event handling semantics and particularly encapsulates the computation of the resource properties into platform-induced timing effects. To operationalize the semantics, we apply our workbench GEMOC Studio for the declarative specification of executable modeling languages. The overall approach then enables verifying the timing effects w.r.t. the real-time requirements specified by timed MSDs in platform-aware timing analyses through applying simulation and model checking in our tool suite TIMESQUARE (cf. right-hand side of Figure 1).

Data Availability The supplementary material encompasses the tooling for the actual approach, the tooling for parts of the evaluation, and evaluation data that is more detailed than in the published article. We published it at Zenodo (cf. reference in the article).

Bibliography

- [HDF22] Holtmann, Jörg; Deantoni, Julien; Fockel, Markus: Early Timing Analysis based on Scenario Requirements and Platform Models. *Software and Systems Modeling (Theme Section on Model-Driven Requirements Engineering)*, 21(6):2171–2211, 2022. J1st @ MODELS’22.