

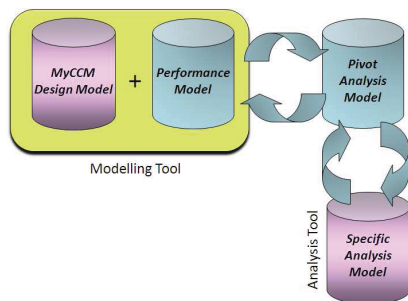
# TEMPO: Performance Viewpoint for Component-Based Design of Real-Time Systems

Rafik Henia  
THALES Research & Technology  
F-91767 Palaiseau – France  
Rafik.Henia@thalesgroup.com

Laurent Rioux  
THALES Research & Technology  
F-91767 Palaiseau – France  
Laurent.Rioux@thalesgroup.com

Nicolas Sordon  
THALES Research & Technology  
F-91767 Palaiseau – France  
Nicolas.Sordon@thalesgroup.com

The growing complexity of applications, combined with constant quality and time-to-market constraints, creates new challenges for performance engineering practices in the area of real-time embedded systems. It is namely expected that delivered products implement more and more complex features, while respecting strict real-time requirements. When developing such real-time systems according to a traditional application of the “V”-cycle, performance verification and validation activities start only when development and integration are completed. As a consequence, performance issues are not detected until the validation and verification stage starts. At this time, they are more difficult and expensive to fix. Thus, a reliable performance prediction at early design stages is essential to guarantee that the designed system meets its timing requirements before time and resources are invested for the system implementation. However, this is not an easy task due the lack of methodologies allowing to directly apply performance engineering activities to design models.



**Figure 1 Performance Viewpoint for MyCCM design**

At Thales, we are developing a performance Viewpoint, called TEMPO, for the integration and application of performance engineering activities as early as possible in the design process of real-time embedded systems, as a mean to shorten the design time and reduce risks of timing failures. The design of such systems is supported in Thales by a software framework, named MyCCM (Make your Component Container Model). MyCCM is a tailorable component based design approach that takes inspiration from the Lightweight Component Container Model (LwCCM) defined by the OMG. It implements the concept of functional components that encapsulate algorithms and are connected through communication ports to create a complete application. MyCCM models are described using UML modelers. The TEMPO Viewpoint for early performance estimation of MyCCM design models is represented in Figure 1. The first step consists in extending the MyCCM model with performance properties, i.e. timing and behavior characteristics of the application (e.g.

execution times and activation frequencies for threads, communication protocols between threads) and execution characteristics of the hardware platform (e.g. scheduling policy). The OMG standard MARTE is key technology for this purpose. However, due to the complexity of its syntax, it may result in very complex and confusing diagrams and models. We have therefore adapted the MARTE syntax based on the Thales designers’ feedbacks, thus allowing representing the performance properties in an easier and much more intuitive manner.

We have opted for scheduling analysis techniques for the performance estimation of the extended MyCCM models. These techniques are well adapted for this purpose, since they rely on an abstraction of the timing relevant characteristics and behaviors. From these characteristics, the scheduling analysis systematically derives worst-case scheduling scenarios, and timing equations safely bound the worst-case response times. However, we faced the problem that scheduling analysis is not directly applicable to extended MyCCM models due to the semantic mismatch between the later and the variety of analysis models known from the classical real time systems research and from the industrial scheduling analysis tools. For instance, in analysis models, a standard assumption is that a task writes its output data at the end of its execution. In contrast, in MyCCM models, a sender task may write data into the input FIFO of a receiver task and continue executing (case of asynchronous communication). In order to fill this large semantic gap and also to ensure a minimum of independence from modeling and analysis tools, we have decided to introduce a pivot analysis model in-between. For this purpose, we have specified a set of rule transforming extended MyCCM models into equivalent pivot analysis models and the later into the selected analysis tool models to which a dedicated MyCCM scheduling analysis, we have developed, is applicable. For instance, in the case of asynchronous communication mentioned above, the transformation consists in splitting the sender task in two tasks. Each of them inherits a subset of the sender task behavior: the first task terminates after writing data into the input FIFO of the receiver task, while the second task is activated immediately after the termination of the first task.

We are next planning to define rules adapting the analysis results back to the original MyCCM design model. This would allow a fully automated and to the user completely transparent scheduling analysis for MyCCM designs. Further details on our performance viewpoint for component-based design, including the model transformation rules and the MyCCM specific scheduling analysis will be addressed in more detail in the conference presentation.