Secure Simulation in Collaborative Settings via a Functional Mockup Trust Center

Workshop on Simulation for European Space Programmes (SESP)
24-26 March 2015

ESA-ESTEC, Noordwijk, The Netherlands

M. Pfeil[1], C. Radermacher[1], J. Mezger[1], C.Kübler[1], V. Fäßler[1]

[1]TWT GmbH Science & Innovation
Ernsthaldenstr. 17
D-70565 Stuttgart
Email: markus.pfeil@twt-gmbh.de

INTRODUCTION

In complex technological fields such as in the aerospace domain, product development is often carried out in a collaborative manner, involving OEMs and possibly several suppliers or system integrators. In today's digital and simulation based development, especially in higly dynamic development environments like concurrent engineering, this usually implies the shared use of various component models, which are coupled in co-simulation frameworks and mutually exchange information such as parameters, state variables or other numerical data. The joint deployment of co-simulation represents a powerful tool for collaborative systems engineering to validate the product design and its operational performance during an early stage of the development process. The majority of the models represent cost-sensitive and confidential information on the part of the OEM or the supplier and safeguarding of intellectual property during model provision and co-simulation is indispensable.

In spacecraft engineering, a centralized model and simulator database would be highly conductive to simplify digital development. Yet, if the intellectual property, which is enclosed in the models and parameters, is not adequately protected, the collaborative development effort can be impeded or rendered impossible altogether, leading to non-acceptance of such an approach from industry side.

Another challenge of collaborative simulation settings and even exchange of system simulators between supplier and customer is the Product Life Cycle Management (PLM). Because the simulation setup proves to be complicated and involves a significant amount of different data sources, it can be difficult or even impossible to repeat a simulation or co-simulation after a considerable period of time. However, repeated simulations might be needed to develop successor products or to track down error sources for product failures during the life cycle. Even though versioning has been well established in software engineering, comparable approaches are not yet prevalent in functional mock-ups, i.e. for digital 3D representations of products which incorporate physical as well as functional properties.

The paper introduces the approach towards a common centralized spacecraft model and simulator database using a Functional Mock-Up Trust Centre (FMTC) that provides intellectual property safeguarding by protecting co-simulation models and system simulators against unprivileged access, unauthorized modification and unavailability without the need for modification of the models. This FMTC has been developed by TWT in the context of the ITEA2 Modelisar project and is proposed to be extended for use with SMP2 or its successor.

The versioned data exchange with PLM or model database systems is readily achieved and integration into the SMP Simulation Environment is targeted.

FMTC: EFFECTIVE PROTECTION OF INTELLECTUAL PROPERTY

The FMTC protects simulation models or simulators against unprivileged access. The basic idea of an FMTC is a cryptographic protection and signature for simulators. The FMTC is executed on a specific server with sealed hardware prohibiting unauthorized access to models that have been encrypted for simulation. Any protected simulation model can only be decrypted and executed inside a dedicated FMTC. If the models are decrypted for simulation, they stay isolated within the FMTC and are erased again when the simulation finishes. The FMTC provides the possibility for secure authentication and authorization of the model providers and users. Also, network communication is only allowed through secure protocols and interface-connectors.

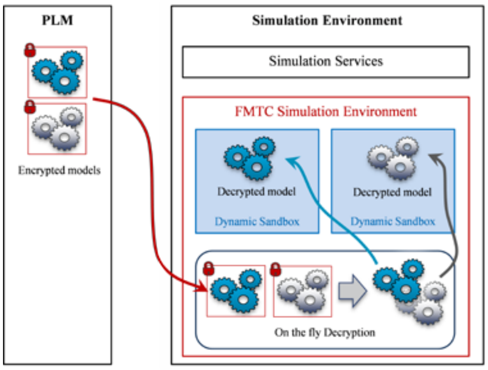

Figure 1: The FMTC acts as Native Simulation Environment and decrypts simulation models on the fly.

DATA MANAGEMENT AND EXCHANGE

The data management for the use with an FMTC is illustrated in Figure 1. The safeguarded models can only be decrypted inside a trusted FMTC and if certain conditions are met, i.e. all symmetric and asymmetric keys are available and usage licenses are approved. Hence, a potentially unsecure data management system, e.g. a PLM, can be used to store and version the simulation models. If the simulation assembly configuration contains the locations and versions of all needed models, the simulation can be set-up and processed in an automated manner.

With a simulation set-up, a stringent configuration control can be implemented, using dedicated access roles to the system to enable efficient change management with full traceability. Also, a full, secure, log of all simulations performed can be kept and directly linked to test and verification documentation. The same is true for the parameter sets that are input to the simulation and the output the simulations produce. Especially simulation runs that show errors or non-conforming results are logged and traceable, for example to the NCR-System.

CO-SIMULATION USE CASE

A co-simulation environment connects models for various parts of a system to a common simulation of the entire system. Since the involved models are often written and integrated by individual programmers and collaborators, common co-simulation environment setups often involve several machines with each of them running specific simulators in software or hardware. An exemplary set of simulators is sketched in Figure 2. While each of the simulators solves specific problems on its own, the co-simulation environment provides the integration of all the individual solutions into a common solution. Contrary to an integration of different models into a single simulation model, co-simulation relies on parallel execution of separate simulators with synchronous or asynchronous communication. Since providers may not be willing to share insights in the details of the simulator with the collaborators, the safeguarding of intellectual property is particularly challenging. The FMTC provides a method to embed a simulator to a co-simulation environment and use shared libraries, but provides a secure and encrypted environment for execution of the model. A Socket based bridge separates the FMTC sandbox from the communication layer of the co-simulation. A solution to embed the FMTC into existing service landscapes and engineering toolchains by a service-oriented architecture (SOA) was presented in [1].

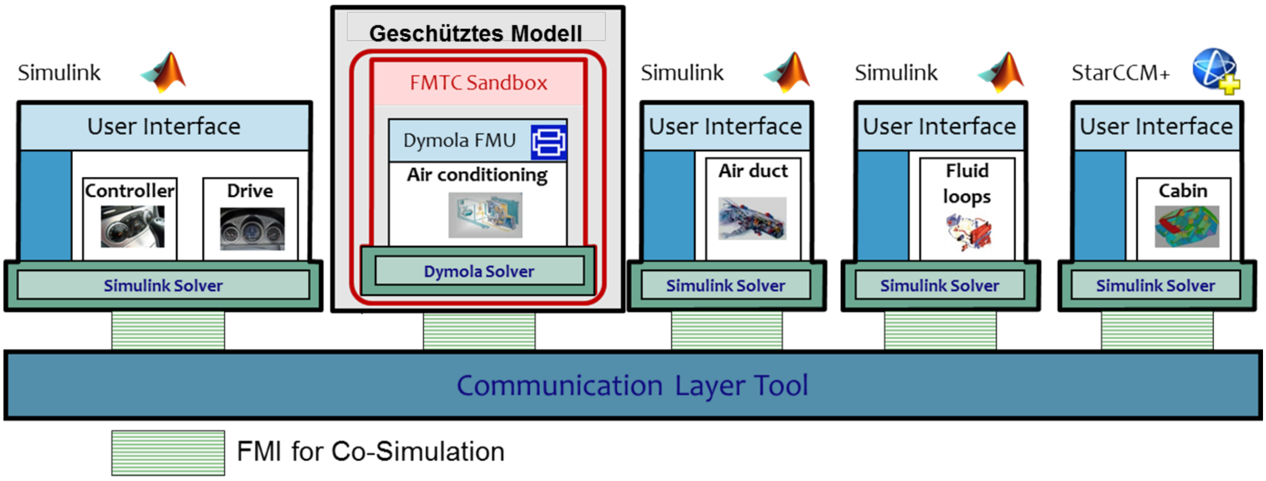

Figure 2: Usage of FMTC in a co-simulation environment.

SYSTEM SIMULATOR USE CASE

For the exchange and secure execution and management of system simulators the FMTC approach offers several benefits over traditional encryption approaches. As the system simulator can be stored and executed in the secure environment, it is easy to assign specific roles between supplier and the customer with regards to who can use the simulator and which parameters are open to change for him. Also, for configuration control and traceability all simulation runs and their settings can be logged and traced to the respective test reports. Furthermore, specific tools and programs needed for execution can be kept together with the simulator without the risk of accidentally upgrading or changing from a running configuration.

SMP ADAPTATION

The original FMTC has been developed for usage with FMI, which is an open standard for model exchange and Co-Simulation (incl. solvers) and was initially mostly used in the automotive domain. It is maintained by Modelica Association. In space applications, the SMP2 standard, which is an open standard for model exchange, is more common. In order to simulate safeguarded models inside an SMP Native Simulation Environment as depicted in Figure 1, the FMTC has to implement the necessary interfaces to the Simulation Services. Furthermore, in order to prevent different models from influencing each other, the FMTC has to offer interfaces between the models according to the XML Catalogue and Assembly files.

REFERENCES

[1] J. Mezger, M. Ditze, M. Keckeisen, C. Kübler, B. Relovsky, V. Fäßler: Protecting Know-How In Cross-Organisational Functional Mock-Up by a Service-Oriented Approach with Trust Centres, 9th IEEE International Conference on Industrial Informatics (INDIN), Lissabon 2011.