Experiences During the Development of a Modelica Library in an Industrial Context on the Example of the Rexroth Simulation Library BRSL

Sven Bätzing[1] Rüdiger Kampfmann[1] Nils Menager[1] Niklas Worschech[1]

[1]Bosch Rexroth AG, Lohr a. Main, Germany {sven.baetzing, ruediger.kampfmann, nils.menager, niklas.worschech}@boschrexroth.de

Abstract

This contribution for the Industrial User Presentations session contains the experiences of Bosch Rexroth during the development of the Modelica library BRSL and discusses hurdles within encryption and licensing, long-term compatibility and conversion scripts and gives an overview about library development in the industry.

Keywords: BRSL, industrial library development, encryption, licensing, model compatibility

1 Introduction

Bosch Rexroth is a leading specialist in drive and control technology and mainly active in three different markets: mobile applications, machinery applications and factory automation. Typical applications for customers within these segments are excavators, hydraulic presses and injection molding machines. Common for all these applications is that it is necessary to deal with multiple physical domains to model and investigate the whole system dynamics. Hence, Modelica is a suitable solution to deal with these challenges and Bosch Rexroth has been actively developing and using Modelica models for more than a decade.

The use of system simulation is of particular importance in the field of hydraulics, since strong nonlinearities as well as complex interactions between components have to be taken into account there, which make it difficult to estimate the overall system performance without simulation. It has to be mentioned, that there are already hydraulics libraries available on the market, but these are mainly designed for the use in special tools. Following the Modelica idea, Bosch Rexroth therefore decided to develop its own library with the aim of compatibility with all relevant Modelica environments. Another argument for in-house development was the freedom in designing the library.

2 Rexroth BRSL

The BRSL developed by Bosch Rexroth is a 1D system simulation library with a focus on hydraulic and electric drive technology. It provides everything from a single source that is needed to create a representation of the static, dynamic and kinematic properties of an entire machine or system in the form of a digital twin. Finally, this digital twin can be used both for well-known and established use cases like the validation of the dynamic system behavior already in early phases of the development cycle, but also for future-relevant topics like the calculation of the energy consumption as well as the CO2 footprint.

The library contains a huge variety of generic components, but also fully parameterized and validated models of Bosch Rexroth components. The generic models have to be parameterized manually, e.g. on the basis of data sheets, and can therefore be used in a versatile and flexible manner. This allows usage of these models to represent the behavior of products from many different manufacturers.

The also available pre-parameterized models, however, require no further parameterization. All parameters are already stored inside the component model, so that the models are ready-to-use as soon as they are placed on the modeling area. On the one hand, this avoids possible errors during parameterization, on the other hand, the time required for modeling can be significantly reduced. Each Rexroth component has a unique type key that specifies it. A Rexroth component model can be easily integrated into the Modelica model by selecting its type key from a drop-down menu inside the parameter dialog of the Rexroth base component model. This allows a well-organized representation of the huge amount of available type keys and makes it easy for the user to insert special Rexroth components into his system model. This will be shown in detail in the presentation. The models of Rexroth products are generated automatically based on product databases. The implementation is done using the replaceable mechanism of Modelica.

3 Experiences during development

Within this presentation we will describe our experiences during the development and the usage of the BRSL library for industrial applications. The capabilities of the library are demonstrated on an industry-relevant example. Due to partially conflicting requirements such as know-how protection, maintainability, and ease of use a special library structure was necessary for the development of the BRSL.

Hereby, we had to deal with special challenges like know-how protection issues, performance issues and tool compatibility.

3.1 Industrial library development process

The presentation will focus on some interesting aspects of the industrial development of a Modelica library as well. The BRSL development follows state-of-the-art principles and includes all elements of DevOps starting with requirements management, through implementation up to monitoring and testing. The development team works in an agile development process. They define the model depth together with the corresponding departments responsible for the hardware product development, deal with the complexity and choose suitable modeling approaches in order to fit the requirements of system simulation. Using a self-developed test environment, the generated models and their test cases can be tested automatically with various Modelica compilers (OpenModelica, SimulationX, Dymola, Modelon Compiler) - either via GUI or via Build server API. The BRSL currently contains about 1500 test cases which are simulated every night and checked for deviations in the results.

3.2 Encryption and licensing

For industrial applications, in contrast to academic applications it is necessary to deal with encryption and licensing. This is not only necessary to protect know-how such as special equations representing the behavior of Rexroth components or characteristic fields containing sensitive data, which is usually already prescribed in the product development process of the components. It is also a mandatory requirement for the implementation of business models for the monetization of models. As of today, the BRSL is separately encrypted and licensed for each tool since the encryption is not yet standardized. This is not only very time consuming but also does not follow the Modelica idea of a tool independent modeling language. We currently support the BRSL for SimulationX and Dymola and encrypt and license the BRSL for these two tools separately. Compatibility with OpenModelica as well as the Modelon compiler is currently being worked on.

Hence, we strongly demand standardized encryption and licensing of Modelica libraries. This can contribute to realizing Modelica's theoretical advantages of developing tool-independent libraries in practice. There is a protocol and implementation proposal for this from Modelon, SEMLA, which is now also included in the GitHub repository of the Modelica Association. SEMLA is already supported by OMEdit. In this presentation, we will also introduce the integration of SEMLA in the BRSL.

Furthermore, it is currently only possible to generate node-locked licenses for a defined MAC address. However, floating licenses are state-of-the-art, and our own corporate guidelines require to integrate other licensing mechanisms. Both requirements are not supported in the latest version of Dymola and SimulationX by default. A satisfactory solution from our point of view is SEMLA. This framework has already been integrated to our licensing mechanisms and proven in first applications. Therefore, it is desirable that all Modelica tools support an appropriate standard.

3.3 Long-term model compatibility and conversion scripts

In the industrial environment, longer-term compatibility of models is also important. In this context, the user experience is not yet satisfactory in all tools and should be optimized. It is not uncommon that models have to be reopened after several years, either to reproduce results, but also to make adjustments for follow-up simulations and using the model as basis for the new simulation, so that the entire model does not have to be rebuilt. Modelica already offers conversion scripts, but in our experience these are not implemented equally in all tools, which turns out to be a big disadvantage in practice. From our point of view, there is also a small need for adaptations of the specification, but most of all there is a need for a consistent support across all tools as this is an important issue. The expectation of the customers that his models are seamlessly executable also in new versions of the BRSL library has in our eyes to be understood as an important requirement. Examples of problems with automatic conversion will be given during the presentation.