Industrialized integration and validation of TargetLink models for series production

Continuous Software Production

The complexity of software systems in vehicles is increasing at the expense of the development of new innovative driver assistance functions. The EB Automotive Software Factory is the key to success because it implements recurring work processes as a production line for software. This means that integration and validation steps can be reproduced and traced at any time, in AUTOSAR-compliant or ISO 26262-compliant projects and also in conventional projects.

Increasing Complexity

When new technologies are integrated in vehicles, more of these technologies involve the actual software. The key task is to connect the growing number of individual systems from different suppliers with each other. The network of actuators and sensors is expanding. There are new application scenarios and the proportion of safety-related functions is also increasing. Complying with the required standards, such as ISO 26262 for the functional safety of road vehicles, is raising the quality requirements even more. The reality for premium vehicle manufacturers today is a variety of about 60 software features, consisting of several hundred software modules on around a dozen electronic control unit (ECU) platforms that are produced by a handful of suppliers and used in 25 series. This increases the complexity during development and during the management of new projects at a disproportionate rate. And today's development teams cannot just be re-scaled to meet this increase. Thus, a wide variety of challenges arise when innovations must be brought up to series quickly and successfully.

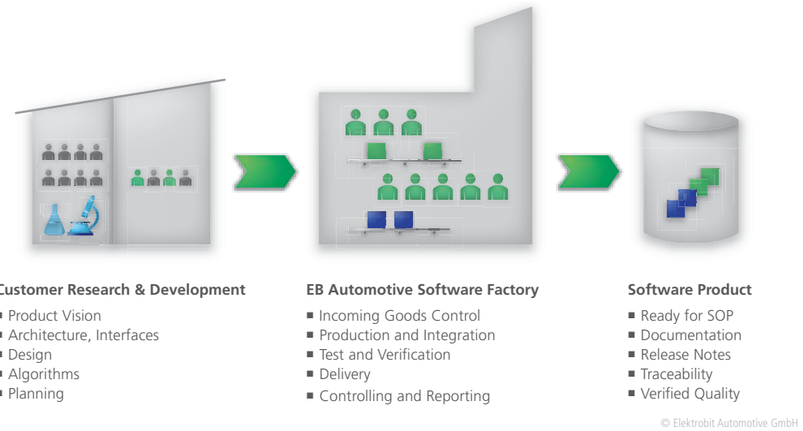
Software as a "Component"

Due to this complexity, in 2010 a German premium car manufacturer brought Elektrobit on board as a partner for developing the software of driver assistance systems. The goal was to let the car manufacturer's own development team focus on its core competencies, namely, developing innovative driver assistance functions in MATLAB®/Simulink®.

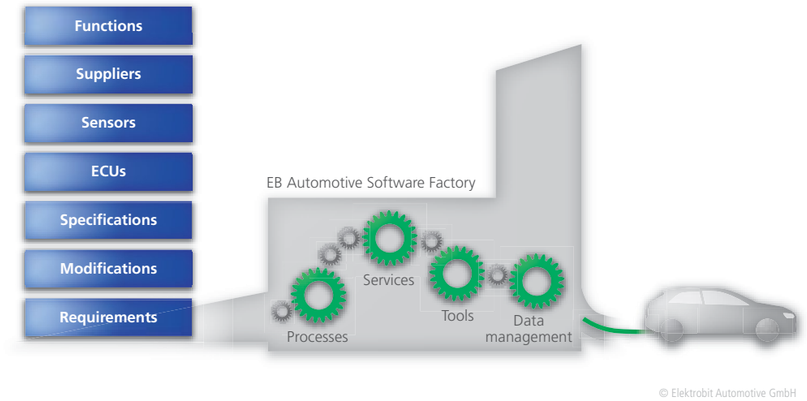
The software development experts at Elektrobit took on the task of implementing the functions in TargetLink® models on various ECUs and in several series and variants, integrating them with Tier 1 platforms and validating the functional safety. The business model that was agreed on handles the software products that Elektrobit integrated like "components". The products are given their own identification number and delivered to production lines all over the world. To take full advantage of the potential that this business model offers, Elektrobit set up the EB Automotive Software Factory for model-based tool chains, in analogy to vehicle production lines. This process generates executable code from models, checks to see if guidelines are being met, and tests the software.

A Virtual Project House

To operate the Software Factory and to be able to independently perform all the tasks being done at the EB locations, EB established a virtual project house and a stable network connection, according to the customer guidelines, to exchange deliveries and balance the project management systems. Elektrobit provide these specific development services for their customers' project teams in a private cloud.



The production lines in the EB Automotive Software Factory.



The concept of the EB Automotive Software Factory.

EB Automotive Software Factory – The Big Picture

The Software Factory is a coordinated multi-step workflow made of modular build steps that ensure continuous integration and delivery. The workflow is used for different types of releases. For example, intermediate results are generated for each modification to give developers feedback as quickly as possible on whether the modifications are still able to be integrated. The production releases are based on a selected software configuration, started manually and released.
The following measures enhance this concept of continuous delivery:

- Testing the delivered components to verify that they match the software quality requirements and to ensure their integratability
- Continuous monitoring of the metrics (for software quality, requirements or test coverage, number of issues, etc.)
- Continuous monitoring of the Software Factory itself (CPU utilization, length of build queues, etc.)
- Traceability, such as between the release and the version of the integrated software
- Automatic generation of a complete documentation
- Automatic provisioning of build machines to scale the computing capacity as desired

Essential Requirements on the Factory

There are some features that are essential for setting up a Software Factory and integrating complex systems:

- The software architecture must reduce integration costs and enable meaningful and efficient tests
- Configuration management must deliver consistent versions from build environments, software, tests, etc.
- Variant management must reduce the number of versions to be integrated and verified
- Practical error prevention methods must reduce the effort for error detection

During the starting phase, the manufacturer's/OEM's tool and process landscape was analyzed to see which elements can be used and to create the interfaces necessary for a comprehensive, homogeneous software development process.

> "In our software development process for safety-related functions according to ISO 26262, we rely on dSPACE's production code generator TargetLink."
>
> *Robert Holzwarth, Elektrobit*

Tool Landscape

The resulting tool chain consists of the following components:

- MATLAB/Simulink/Stateflow®, with other toolboxes
- dSPACE TargetLink as the code generator
- MES Model Examiner® for static model verification
- Polyspace® for static code analysis
- MTest and TPT for model module tests
- Mini HIL systems and tools from the vehicle manufacturer/OEM
- Jenkins as the continuous integration system
- PTC Integrity for configuration, change and release management at the OEM
- Subversion and JIRA for version and change management at Elektrobit
- pure::variants for variant management
- Further scripts and tools for
  - Model integration and system integration
  - Variant management
  - Tier 1 specific adaptations
  - Generating AUTOSAR interfaces
  - Interface tests

Mapping on Virtual Machines

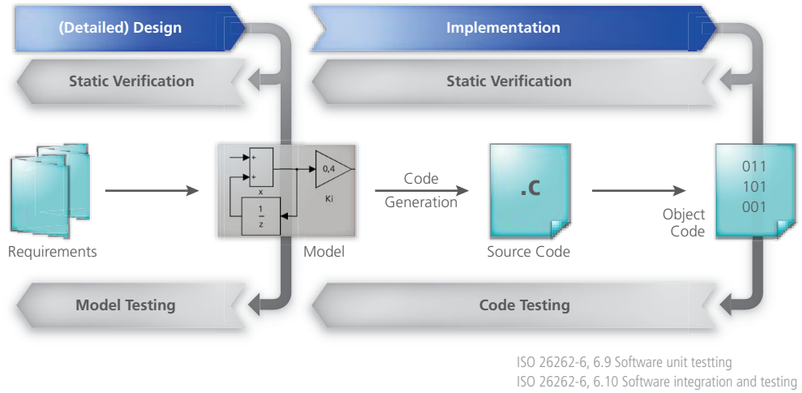
The actual build environment was mapped to virtual machines (VMs) to keep the installation and especially the configuration of the tools under control.
Different versions and variants, such as MATLAB R2007 and TargetLink 3.0 or MATLAB 2011b and the 64-bit TargetLink 3.4, were mapped on their own instances. The central continuous integration systems are based on these VMs. Local instances are provided as the development environment for several applications. For example, Factory 2 Go is used to generate software versions in a consistent build environment during test drives without any online connections.
The advantages of virtual machines are also used for the long-term archiving of complete projects, since only files need to be archived, not any hardware.

Build Process

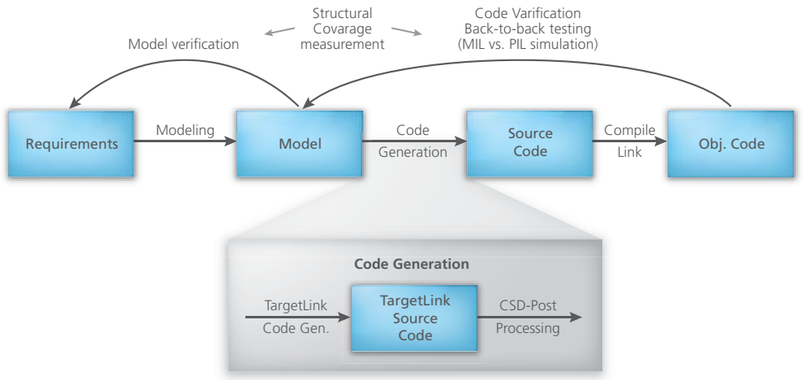
The designed process is based on automating and modularizing each step with clear input and output artifacts.
For example, at the interface between the OEM and Elektrobit, there is an automatic test of incoming goods set up for MATLAB/Simulink models and C code. This test involves the following tasks:

- measure the complexity of the modules
- check whether the relevant modeling rules and design guidelines are met
- check the consistency of the internal and external interfaces with respect to the specified design
- generate reports on the findings and metrics

The model integration step requires the models and components and the variant model of the OEM as the input artifacts. The model modules specified in the variants are used in a generated interface framework within the overall model and connected or replaced by empty modules. Code generation with TargetLink is fully automatic. During post-processing, hook functions are used to perform Tier 1 specific modifications in the resulting C code. Measurement and parameter files are



The TargetLink reference workflow provides guidance on how to fulfill functional safety requirements with model-based development methods and tools.



How the reference workflow enhances validation.

generated, and the layout of the generated header files is controlled by style sheets and parameterized by the TargetLink Data Dictionary. The tool created by the vehicle manufacturer/OEM for AUTOSAR frame generation and post-processing was simplified considerably in cooperation with dSPACE and reflected in the new features of TargetLink 3.4.

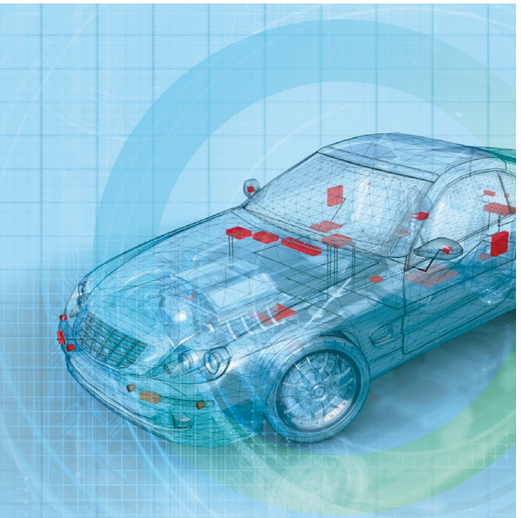
The TargetLink Reference Workflow

ISO 26262 requires that the use of software tools for developing safety-related software meet the requirements of Part 8, Section 11. This means that the tool's reliability must be evaluated in the concrete project environment and that the tool must be given a quality level that reflects the result.
The production code generator TargetLink itself was certified by TÜV SÜD for the development of safety-related systems. The certificate confirms that TargetLink is suitable for software development according to ISO 26262 up to ASIL-D, IEC 61508 up to SIL 3, and the derivative standards. dSPACE also provides a reference workflow document on the model-based software development of safety-related systems with TargetLink. This document was also examined by TÜV SÜD and acts as a guideline on how to meet the specific safety standard requirements in safety-critical projects. Thus, when the reference workflow is used, no further qualification measures are required for TargetLink. For the self-developed post-processing procedure, the TargetLink Reference Workflow was examined in more detail to evaluate to the process step that was downstreamed by Elektrobit. The evaluation showed that the recommended measures, such as back-to-back tests, also cover this downstreamed process step. This means that here too, no further tool qualification measures are required when the TargetLink Reference Workflow is used.
dSPACE and Elektrobit work together hand in hand to implement the reference workflow and especially to perform the recommended back-to-back tests. ■

Robert Holzwarth,
Elektrobit (EB) Automotive GmbH

"The production code generator TargetLink makes complex processes such as AUTOSAR-compliant code generation extremely easier."

Robert Holzwarth, Elektrobit



Conclusion

Elektrobit successfully took on the production of production software for several product lines and generations at an OEM. "Software components" are created in a customer-specific private cloud and delivered to the production teams. The continuous integration of the software is enhanced by measures to ensure process and software quality. Elektrobit was able to use the TargetLink Reference Workflow successfully and expand it on their own in-house tools. Elektrobit's experience and expertise in their EB Automotive Software Factory are key factors that help manufacturers be successful when they perform complex projects to develop innovative software for series production. By automating software production and delivery, Elektrobit has delivered more than 230 releases to its customer on time since 2010, with over 75 individual production releases.
Elektrobit continues to expand the concepts of the EB Automotive Software Factory. For example, they are making the supply chain processes more transparent and thus easier to model. The automotive, commercial vehicle, and component supplier branches can access project-specific implementations of the Software Factory worldwide in a private cloud.

Robert Holzwarth
Robert Holzwarth is Head of Technology & Innovation Software Integration and Services at Elektrobit Automotive in Erlangen, Germany.