AUTOSAR and the Automotive Tool Chain

Stefan Voget
Continental Engineering Services GmbH,
Regensburg, Germany

ABSTRACT - This paper will present how the new concepts of the AUTOSAR system methodology influence the SW-development tool-chain landscape.

Keywords: AUTOSAR methodology, Tool Chains, ARTOP

I. INTRODUCTION

Today, a lot of series projects run with development activities based on the AUTOSAR standard. No sub-domain of the vehicle is excluded. The developments run in interior, power-train as well as in chassis although their requirements are quite different. In none of the projects one starts the development at zero. The reality shows, that there is not the one and only way to migrate existing solutions to AUTOSAR. But, several "best practices" can be derived by the experiences done in all automotive domains which already influence the processes and tool-chains.

In this article we will motivate the increased importance of the tool chain. This is due to the AUTOSAR methodology which shifts more complexity from an implementation focused way of work to a configuration process based way of work. This enables the introduction and use of more automatism in the software development. The new character of the AUTOSAR methodology enables and requires new tools.

Mainly two alternative approaches for the development of these tools can be observed. Some vendors enhance their existing tooling with export and import capabilities of the AUTOSAR format but do not adapt the workflow complete. Others use the standard as an opportunity to design new tool capabilities.

A lot of new tool innovations that realize the concepts around the AUTOSAR system methodology are on the way. One platform for these innovations is ARTOP (AUTOSAR Tool Platform). The initiative provides common base functionality for development tools used for designing and configuring AUTOSAR compliant systems and ECU's. It is an Eclipse-based AUTOSAR tool infrastructure platform (See www.artop.org).

Based on this platform we will show how the AUTOSAR concepts like system development, system configuration, timing analysis and code generation can be brought together into a seamless tool-chain.

II. AUTOSAR

The main concept of the AUTOSAR approach is to separate application and infrastructure software. On application level the AUTOSAR software components (SW-C) can be found. A SW-C can have a small, reusable but also complex automotive functionality.

Software components can be connected with each other through well defined ports. In terms of implementation, the AUTOSAR software component is independent from the infrastructure, i.e. it is independent from the type of the microcontroller and the type of electronic control units (ECU) the SW-C is mapped to.

This separation between function and infrastructure is important for the reusability of software components on different ECU's and is achieved during design time through the Virtual Functional Bus (VFB). The Virtual Functional Bus takes care of the communication between different components and between software components and the hardware. The VFB realizes the goal of being able to relocate software components and allows a virtual integration of AUTOSAR software components in an early development phase. AUTOSAR provides a methodology and proposes the use of tools for this purpose.

III. AUTOSAR METHODOLOGY

The AUTOSAR Methodology describes development steps which have to be executed during system development with AUTOSAR. It is a booklet to support the exchange of model data in early development steps. It is neither a process description nor a business model and therefore does not predefine a strict order in which the activities should be accomplished.

In the first step, system information is collected that will be used for a configuration of the system. These inputs are formal descriptions of software components, ECU hardware resources and system topology.

The first important step, the system configuration, takes these descriptions and performs a mapping of the software components to one or more ECU's. With the help of the VFB principle the application software had been modeled independent from the concrete hardware until this step. A second mapping is performed with the mapping of signals to bus frames.
After the system configuration step one knows the software components that are allocated to a single ECU. Now, the further steps operate on a single ECU. To be able to concentrate on a single ECU, the subset of information that is relevant for the further implementation of that ECU is extracted from the system configuration description.

Additionally, necessary information for the implementation is added in the ECU configuration activity. The output of this step is the ECU Configuration Description containing all information concerning one control unit. Using this description executable software for this ECU can be generated.

This step includes the generation of code, compiling code and connecting everything into an executable.

The description of an AUTOSAR system is performed using appropriate templates. Those templates define the AUTOSAR meta-model and allow a formal description of an AUTOSAR system. Based on these descriptions the AUTOSAR Methodology generates an ECU executable.

### IV. CHALLENGES WITH AUTOSAR METHODOLOGY

SW architectures with several thousands of networked software components lead to a high complexity of today's distributed embedded systems in cars. To improve the distributed development, AUTOSAR reduces the usage of different data formats. Now, models are exchanged between the development partners based on a standardized exchange format.

But, the AUTOSAR meta-model is very complex. It is a new terminology with complex relationships between the defined objects. So far, only limited knowledge can be found in the market.

The models will be exchanged and reused between suppliers and customers during the life cycle of a software development. However, the AUTOSAR methodology does not define who is doing what and when in a software development. Procedures and the sharing of the different roles and tasks have to be identified and contracted between the OEM and their suppliers, using interoperable tools to manipulate these models.

The standard itself does not give support in the modeling process and just partial support in the data consistency. Concurrent configuration / modeling and parallel development are not considered by AUTOSAR but are a must for the usability of a tool-chain.

Furthermore, until now several AUTOSAR versions are in use, that are incompatible with each other.

The templates defined by AUTOSAR improve the interoperability of tools but do not completely solve the problem of usage of too many different tools. As each step of the methodology leads to a result, which is saved in an exchange format, the change of a tool can lead to loss of information and to errors.

Additionally to the AUTOSAR data, very often the tools need to store presentation data for the tool itself. Exporters, importers and translators are needed.

### V. ARTOP

A common framework, which enables the development of a continuous, lossless tool chain, is helpful to overcome the challenges.

ARTOP (AUTOSAR Tool Platform) is such a framework jointly developed by the ARTOP user group. The ARTOP user group is a group of users of the AUTOSAR standard with a special interest in AUTOSAR tools. The organization of this group is comparable to the Eclipse Foundation. This group had been founded by BMW Car-IT, Continental AG, PSA and Geensys. Until now further partners joined. ARTOP is open to all AUTOSAR members.

The ARTOP user group provides an infrastructure platform for the development of tools used for the design and configuration of AUTOSAR systems. ARTOP implements non-competitive base functionalities needed by each AUTOSAR tool.

The core component of ARTOP is the AUTOSAR meta-model implementation. It supports all available AUTOSAR meta-models like 2.0, 2.1, 3.0, 3.1 but also the newest 4.0. Furthermore ARTOP encloses AUTOSAR XML schema conformant serialization, rule-based validation, model refactoring, workspace management, example editors and further utilities.

Through ARTOP the interoperability of different tools that are used to support the methodology steps can be improved and commercial tools with better quality can be developed in less time, since only key functionalities have to be implemented.

Since the first release, published in 2008, more than 200 users had shown interest in ARTOP. In 2009 the first tools based on ARTOP arrived in the market. This shows the success of ARTOP. It seems that the automotive tool market awaited such an approach.
VI. ECLIPSE

ARTOP is an Eclipse-based infrastructure platform, i.e. it is build on top of Eclipse and uses Eclipse technologies like EMF. Mainly, the technologies provided from Eclipse are taken by ARTOP and are applied to the AUTOSAR meta-model specifics.

It is not the goal of ARTOP to develop new, generic tool utilities. Eclipse already provides a modeling framework and code generator to develop tools based on structured data models.

For the purpose of this paper it is sufficient to highlight three helpful technologies provided by Eclipse that are attractive for the development of AUTOSAR tool-chains.

1) The Plug-In mechanism from Eclipse provides well defined interfaces to plug different functionalities together to an enriched tooling. As Plug-in code is loaded only if necessary this mechanism enables well performed reaction times of the tools on user input.

2) The Extension Points mechanism contributes functionality to a specific plug-in, which defined the extension point. This enables the users of tools to extend the functionality in an easy way. Adaptations to the needs of the user can be operated by him. This is an often whished feature in the Automotive industry.

3) The Wizards utilities enable to guide the user through a defined sequence of steps to fulfill a specific task Eclipse provides support to easily create wizards.

Recently, an automotive working group had been founded in Eclipse. This group wants to define an Eclipse target platform for the automotive industry. This target platform should be used to develop tools for the whole development process lifecycle. AUTOSAR influences parts of this development process. Therefore, ARTOP is of highly interest for the working group. Due to restrictions given by the AUTOSAR contracts, the AUTOSAR dependent parts of ARTOP cannot be handled by the Eclipse automotive workgroup. But the AUTOSAR independent parts can and are. ARTOP is the first concrete use case for the workgroup.

VII. ARTOP ENABLES SEAMLESS TOOL CHAIN

This chapter will present some challenges in the existing tool landscape and how ARTOP supports solutions.

A. AUTOSAR has boundaries

Having a look to the AUTOSAR methodology one recognizes that it does not cover the complete development process lifecycle. The influence of the methodology begins somewhere in the middle of the system architecture modeling step and ends directly before the compilation step. Topics like requirements management, hardware development or built management are not related with AUTOSAR.

One cannot expect that ARTOP provides a platform to enable a seamless tool chain from requirements to executable. But, as AUTOSAR influences directly a central part of the development process lifecycle, ARTOP is a good base for harmonization of the tool landscapes. It provides useful features to extend the platform itself – on editor level as well as on meta-model level. The extension point mechanism from Eclipse is incorporated.

B. Meta-model structure in-homogeneous

The structure of the AUTOSAR meta-model differs very much at system side from the one on ECU side. This reflects the differences in the use cases to be covered on both sides. The today's tool market follows these borders. Normally a tool works either on system level or on ECU level but not on both. The interoperability between both sides are usually done file based. ARTOP enables tool chains that do not require from the user to start several tools, to export from the one tool and to import to another tool. Everything is shifted from the file level to the meta-model level that enables a better data management consistency.

VIII. CASE STUDY – CESSAR-CT

On the one side the arguments for an approach like ARTOP are obvious. It works and one gets something for free.

On the other side one may ask the question about why tool vendors should use ARTOP, as they may give sensitive parts of a tool development into an open user group.

The development of an AUTOSAR configuration tooling within Continental Engineering Services (CES) can be taken as a case study for the successful usage of ARTOP. Together with the AUTOSAR standard basic software – also implemented
and put on the market - these software products enable the engineering activities of CES.

For the engineering business it is of main importance to have a flexible, extensible and customizable tool. Normally, these are not the attributes one finds in the tools available on the market. These are usually oriented on a fixed end user use case and not usable for an engineering expert that develops and adapts something for the customer.

---

**Figure 3** CESSAR-CT showing a SW-C editor

The main use cases supported by the tool are the basic software developer and the ECU integrator. Therefore, the focus is on the ECU part. But, with AUTOSAR also these use cases need information from the system part of the meta-model. E.g., Figure 3 presents a SW-C editor. It is a typical example for a situation, in which users normally have to open several tools in parallel. As CESSAR-CT is based on ARTOP the user gets a seamless AUTOSAR tooling.

It had been shown that the extensibility mechanisms provided by ARTOP are of main importance for CESSAR-CT. CESSAR-CT provides a Pluget mechanism such that the engineering expert or also the end user can extend the tool. Code generators of different technologies can be integrated and a form editor enables the easy extensibility and customization of the UI.

---

**Figure 4** One editor based on several layers

Figure 4 shows a typical example how the end product, the commercial tool, consists of. A basic explorer is already provided by Eclipse. ARTOP adds AUTOSAR standardized elements. Last but not least the tool supplier adds the use case specific features. This last step enables the usability of the tool for the end user.

As Continental had not been a tool vendor before it had been a challenge to act as a founding member of the user group ARTOP. After nearly two years with ARTOP it is accepted, that the approach enabled CES to develop a tool that can be sold on the market.

---

**IX. OUTLOOK**

The next phase in the development of AUTOSAR tool-chains will be driven by the new concepts of AUTOSAR release 4.0. The methodology and the meta-model had been enhanced in this release. "Functional safety" and "timing extensions" are only two concepts with major influence that have been incorporated. ARTOP will take these new concepts into account and provide platform support such that the idea of ARTOP can also be realized in future.

---

**X. REFERENCES**

[2] AUTOSAR Methodology; see [1].
[3] AUTOSAR Technical Overview; see [1].