SystemDesk and TargetLink speed up AUTOSAR workflows
by exchanging component containers

Container
Swapping

The current versions of SystemDesk and TargetLink make AUTOSAR-compliant software development much easier by using containers to exchange software components (SWCs). This brand-new concept ensures transparency, efficiency and reliability in the AUTOSAR development process.

AUTOSAR in Practice

The efforts of the AUTOSAR development partnership to establish a unified software architecture and standardized exchange formats have achieved results in automotive production projects in numerous different ways. AUTOSAR-compliant application software is usually developed with a combination of a software architecture tool and a behavior modeling tool. The two tools exchange data in the AUTOSAR ARXML format in an iterative process called an AUTOSAR round trip. This exchange of data has proven to be critical and practical experience shows that well-coordinated tools and efficient workflows are vital for efficient, AUTOSAR-compliant work. Two of the typical issues that often come up in practical work are:

■ What is the most efficient way to handle the data and files in an AUTOSAR round trip for developing a software component (SWC)?
■ How can components be tested efficiently during interaction with each other and with the AUTOSAR Runtime Environment (RTE)?

SystemDesk® 3.0 and TargetLink® 3.2 have the perfect answer to these and other issues: a completely new concept based on the exchange of SWC containers.

The Concept of SWC Containers

SWC containers let users perform AUTOSAR round trips with just a few clicks, and also make the interaction between SystemDesk and TargetLink completely transparent and reliable (figure 1). SWC container exchange is based on the following principles:

■ All the files involved in developing a component (ARXML files, code files, A2L files and so on) are collected in a container and managed together by using a single container catalog file.
■ All AUTOSAR-specific data is partitioned into different ARXML files such that either the architects or the component developers have complete ownership of each file. This simplifies merge scenarios in AUTOSAR round trips and ensures clearly defined responsibilities.
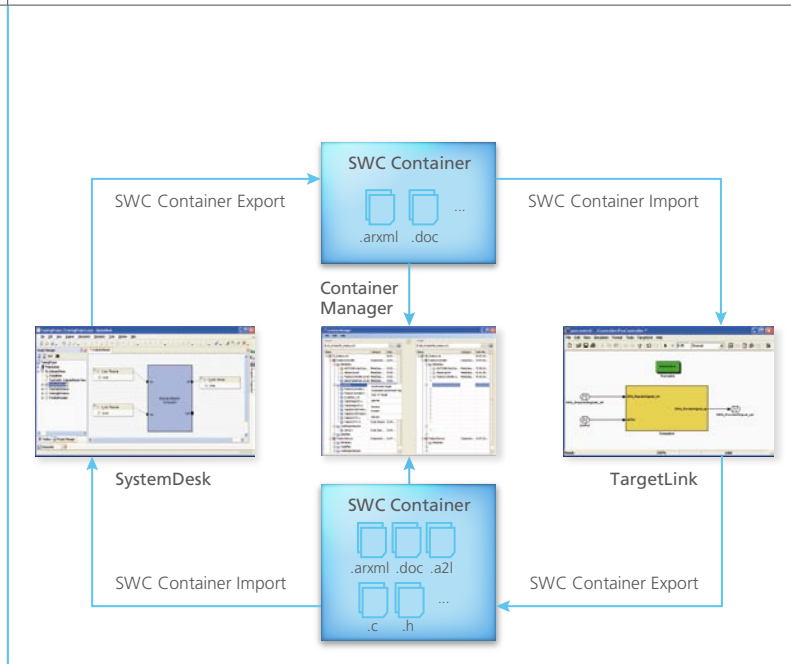■ Meta-information is placed in the

Figure 1: AUTOSAR round trips between SystemDesk and TargetLink by means of SWC containers.

the component developer, the architect exports an SWC container from SystemDesk. The container holds the necessary AUTOSAR data in ARXML files, and the architect can also add other specification documents (figure 1).

To perform component implementation, the TargetLink user imports the SWC container and uses the component's interface specifications to generate an AUTOSAR frame model. The design of the control function is then added to the frame model. Finally, the TargetLink user directly generates AUTOSAR-compliant code and executes it in the Simulink environment to test components in software-in-the-loop and processor-in-the-loop simulation.

When these activities have been completed, an updated SWC container is exported from TargetLink. The updated container holds not only the ARXML files with added implementation information, but also additional artifacts such as code files and A2L files (figure 1). The container meta-information and the stored workflow ensure that the interfaces involved in the AUTOSAR

container catalog file as file categories and responsibilities. In combination with a user-configurable workflow description in XML format, the meta-information ensures that the different statuses of an SWC container are synchronized in a controlled, secure manner.

The SWC container concept implicitly uses ARXML files, so it is one hundred per cent AUTOSAR-compliant. It is also a useful extension to the AUTOSAR standard, since users

have to handle just one single container catalog file, which then implicitly manages all the other files.

AUTOSAR Round Trips Based on Component Containers
In a top-down workflow, the development of a software component as an element in a software architecture is begun by the software architect, who specifies the component, including its ports and interfaces, in SystemDesk. To make the specifications for a component available to

Figure 2: Comparing and synchronizing different project statuses with the Container Manager.

A
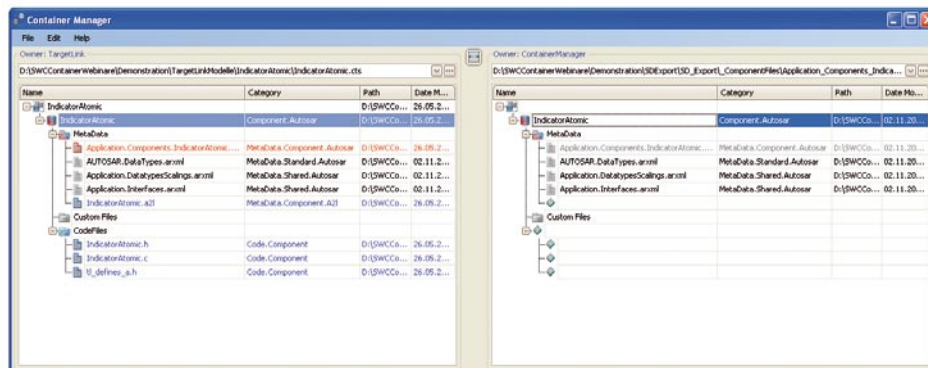
More transparency, efficiency and reliability in the AUTOSAR development process.

round trip cannot be modified unintentionally.

In addition, there is a specialized container handling tool, the Container Manager, which enables synchronization between containers and also provides a synchronized tree view for comparing them visually (figure 2). Thus, the user can control how to synchronize and merge the different project statuses that a container inevitably runs through during a round trip. For example, software architects can use the Container Manager to compare their version of the container with the version supplied by the TargetLink users, before reimporting the container in SystemDesk to complete the round trip shown in figure 1.

Simple Integration into Different Simulation Scenarios

Because the SWC containers are designed to hold not only AUTOSAR

ARXML files but also code files and A2L variable description files, container exchange gives TargetLink users an extremely convenient, direct connection to SystemDesk's simulation functionality (figure 3). The implementation files generated by TargetLink and placed in the SWC container are used directly in the simulation build process performed in System-Desk. The process compiles and links all the components against an RTE that is generated in SystemDesk. All this adds up to a system that covers numerous different simulation scenarios and tests.

- Users get immediate feedback on whether TargetLink components can be integrated into the software and whether a compile-and-link operation for the components and the RTE is possible.
- The interactions between an extremely large number of compo-

nents can be simulated and tested at "virtual function bus" level with push-button ease in SystemDesk.
- The A2L files in the containers also give simulation users direct access to internal component variables so that they can close current gaps in the AUTOSAR standard.
- In addition, the RTE can be precisely configured for more detailed simulation scenarios, including the scheduling and configuration of individual tasks, and simulated access to modules in the basic software.

SWC container exchange addresses all these scenarios directly and conveniently to simplify testing activities in the AUTOSAR development process. ∎

Figure 3: Simulating the interactions between software components in SystemDesk.