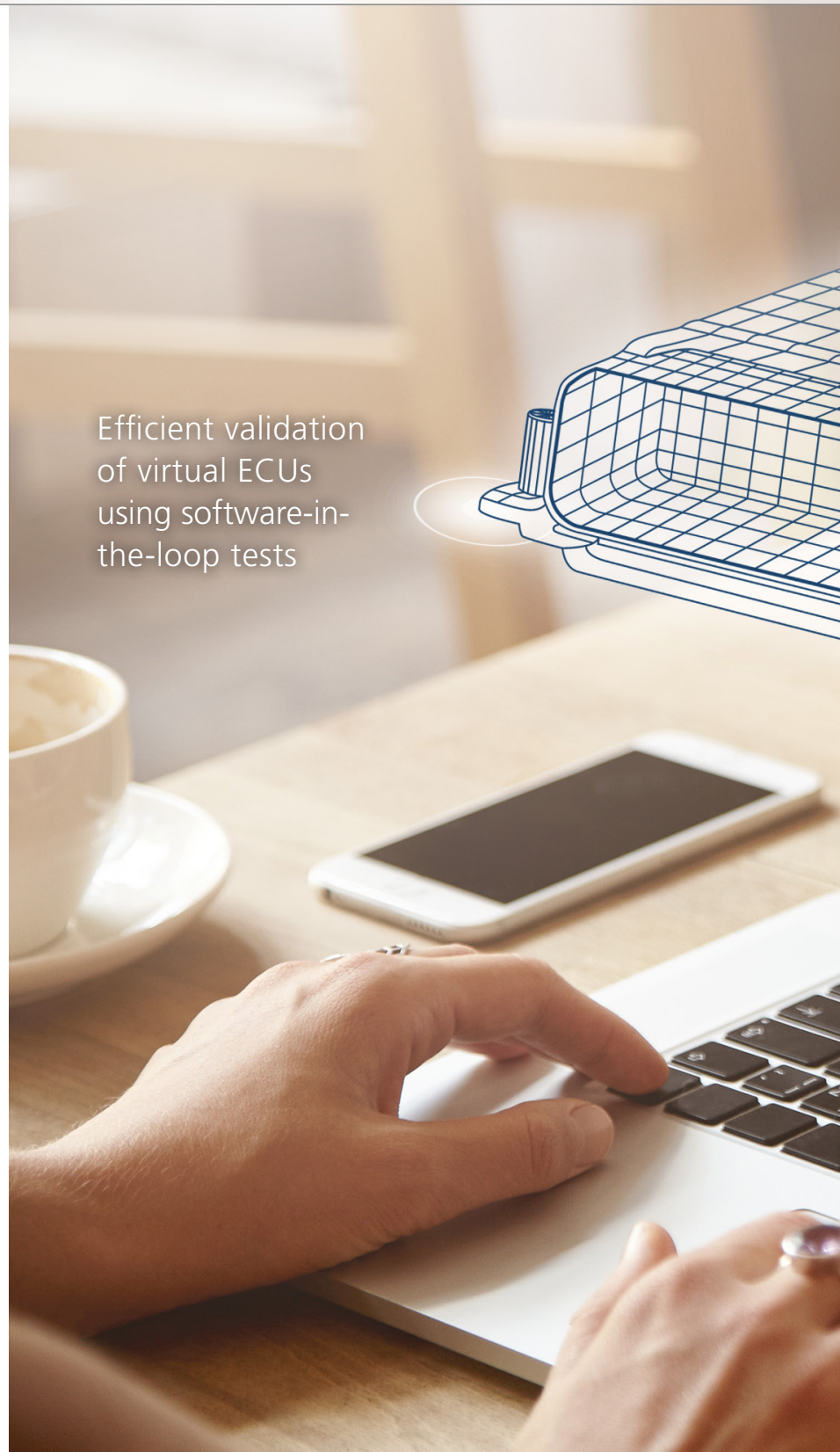To frontload tests for ECU functions and complete them as early as possible in the development process, new test strategies are required. Volkswagen (VW) has therefore decided to use software-in-the-loop (SIL) tests as an addition to the existing hardware-in-the-loop (HIL) tests. This requires virtual ECUs (V-ECUs) that are used at an early stage in place of hardware ECUs for realistic testing. An efficient approach was needed to create and simulate V-ECUs based on existing development artifacts. It had to be generic enough to be suitable for ECUs of different domains and for different vehicle types – from engine control units for internal combustion engines to high-voltage charge management for electric vehicles.

Merging Different Source Formats

Virtual ECUs are based on the same development data as the final ECU: ECU function code and an XML description file for the software architecture. This information is generally distributed across several development departments. Therefore, the first challenge was to identify all required files. Another challenge was to use code that was not based on AUTOSAR as the basis for virtual ECUs, because the software components of various ECUs were only partially developed according to the AUTOSAR standard.

Generating Virtual ECUs

VW chose the dSPACE SystemDesk architecture tool to efficiently create the V-ECUs. With SystemDesk, ECU description files can be imported, the files can be used to build AUTOSAR-compliant architectures, and the architectures can then be prepared for the existing function code. For this purpose, a software component (SWC) is first generated from the imported description file for each function. The code related to the function (C code or object code) serves as the implementation basis for the SWC. To use the non-AUTOSAR-based code, the functions are packed into AUTOSAR elements: For example, function calls and their timing in an SWC are defined in an internal behavior.

This encapsulates the implementation and thus the references to the code. Afterwards, the SWCs are connected to an ECU software architecture according to the description file. The last step is to create the

>>

Efficient validation of virtual ECUs using software-in-the-loop tests

Naturally Virtual
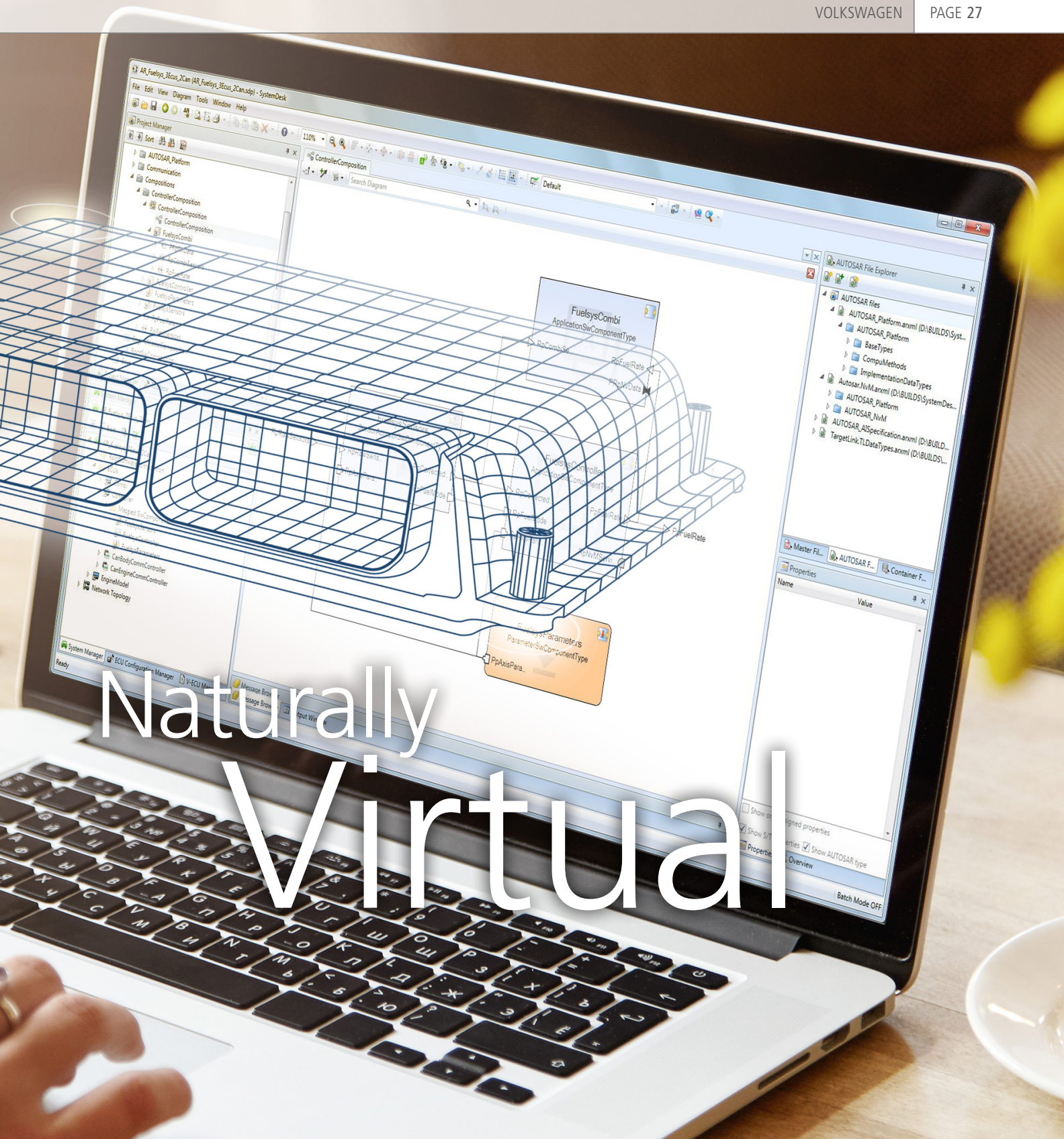
Testing new functions as soon as they become available – a task that requires flexibility and the frontloading of tests. For this purpose, Volkswagen relies on automatically generated virtual ECUs. The dSPACE tools SystemDesk and VEOS help developers generate and simulate the software.

Python script controls SystemDesk:
1. Description file > software component
2. Code > Implementation of each software component
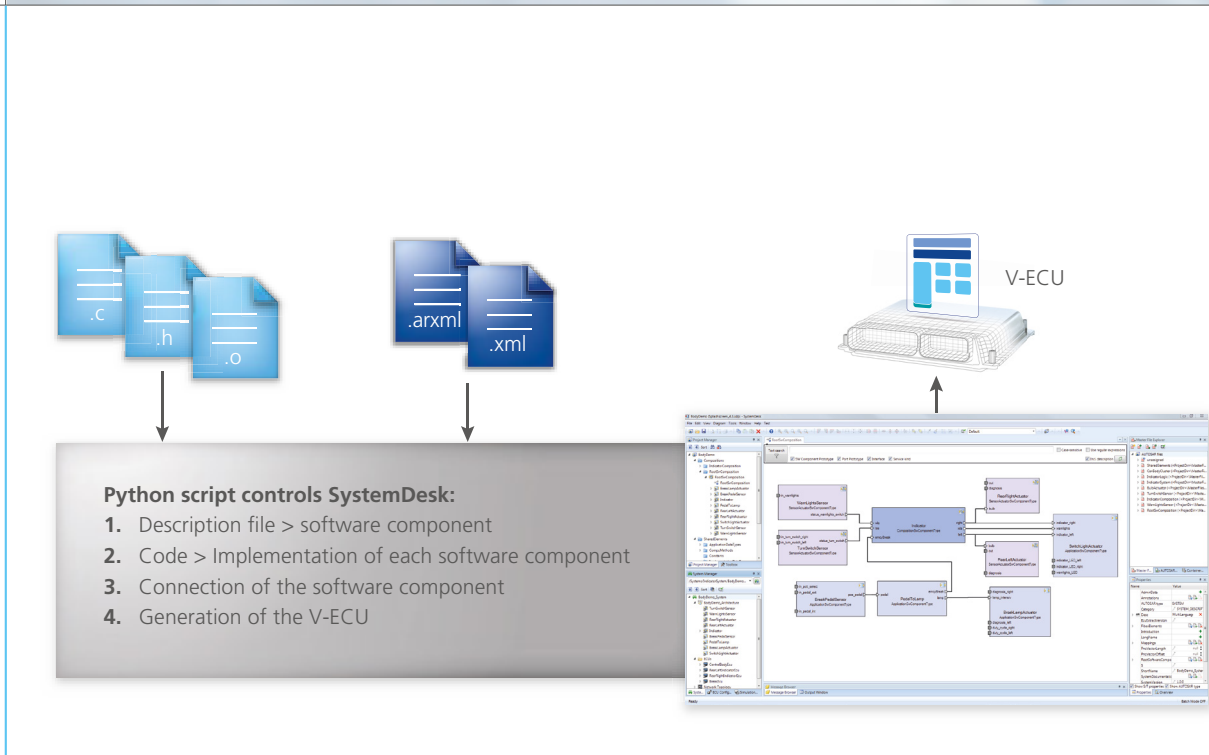3. Connection of the software component
4. Generation of the V-ECU

Figure 1: To create a virtual ECU, the existing code is integrated into AUTOSAR software components to generate AUTOSAR-compliant ECUs from non-AUTOSAR-based code.

> "The powerful SystemDesk functions and the very detailed documentation made it so much easier to automatically generate virtual ECUs."
>
> *Kirsten Pankratz, Volkswagen AG*

run-time environment (RTE) and integrate it with the basic software from the SystemDesk templates. Finally, a V-ECU can be generated using SystemDesk (figure 1).

Automated Generation via Script
Aside from the manual method,

Figure 2: ControlDesk provides an easy-to-understand overview of variable states.
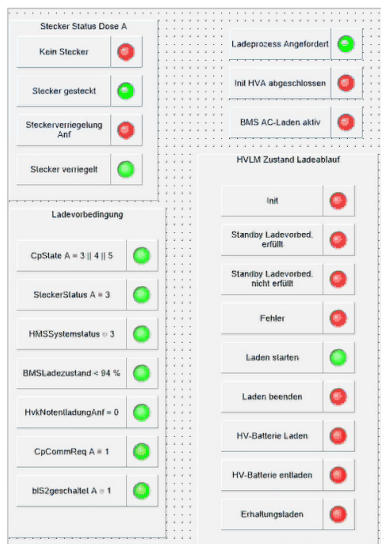
there is also an efficient option for automatically creating virtual ECUs. For this, the description files are analyzed using a Python script and the work steps are executed via the SystemDesk application programming interface (API). To prepare for automation, only the names of the software components of the desired virtual ECU and the file storage paths of the configuration files are required. If this data is available together with the description and code files, it only takes approximately ten minutes to create a virtual ECU via a script.

First Project Experience: Engine Control
The first project implemented at VW was focused on the virtual validation of engine ECUs. Specifically, this involved crankshaft-angle-synchronized function calls, which are decisive for

precise engine control. A VW-programmed function that evaluates the current crankshaft angle implemented the call of the crankshaft-angle-synchronized raster in the 1 ms raster. Depending on the current position of the crankshaft, the function triggers the call of the SWC implementations. The first setup of the virtual ECU was still done manually to become familiar with the software and the necessary work steps. Based on the acquired knowledge, VW then created the Python script. Thanks to the detailed documentation for SystemDesk, its implementation went smoothly.

Virtual and Real ECU Compared
The dSPACE VEOS simulation platform, which only requires a standard PC as hardware, was used to evaluate the functionality and quality of the generated virtual ECUs. For the test, a V-ECU and a Functional Mockup Unit (FMU) were connected with VEOS to form a system. Real data
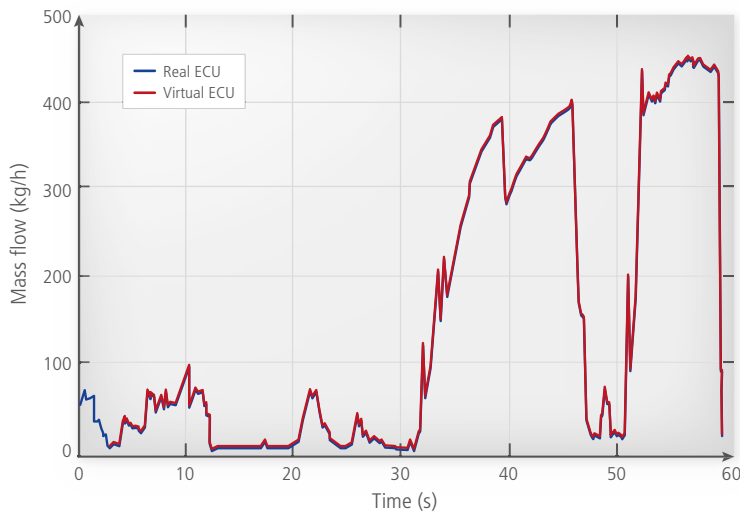
Figure 3: The comparison between the test data of a virtual ECU and that of a production ECU illustrates the very realistic behavior of the virtual ECU.

from vehicle measurements was used as stimulus data. The data was loaded into Simulink and integrated via the FMU to stimulate the virtual ECUs in VEOS. The output values of the virtual ECUs were measured with dSPACE ControlDesk. The input and output variables of both the virtual and the real ECU were then graphically prepared and compared (figure 3). The test results matched almost completely, which illustrates the high quality of the virtual ECUs and makes them a full-fledged supplement to the existing tests. Another advantage is that virtual ECUs can be used in parallel in multiple development projects. This eliminates any dependency on the availability of hardware prototypes.

Experience with dSPACE SystemDesk

SystemDesk has proven to be a powerful, convenient tool for creating virtual ECUs. SystemDesk enables the simple modeling of AUTOSAR architectures and can generate virtual ECUs, including the RTE and basic software. Due to the comprehensive, intelligible documentation, all work steps can be easily automated via an

API using a script. It took only a few weeks to go from the first script prototype to the first virtual ECU. The clear graphical representation of the ECU architecture in SystemDesk provides a good overview of the existing functions, their interfaces, and the functional relationships.

Since the functions are usually provided by different teams, SystemDesk also supports an efficient collaboration between developers.

Summary and Outlook

The first project (validation of EECUs) has shown that virtual control units have the maturity level required for developing and testing production projects. Another project dealt with creating a virtual ECU for the high-voltage charging manager that controls the communication between the electric vehicle and the charging station. This project showed that the design process for virtual ECUs covers the entire motor and engine range, from the combustion engine to the electric motor. The virtual ECUs are now being used in various development groups at Volkswagen for software-in-the-loop testing. For this, existing HIL tests and configurations are reused, which noticeably reduces the effort required for test creation. ∎

Kirsten Pankratz, Dominik Ott, Volkswagen

> "The clear overall representation of the ECU architecture in SystemDesk supports an efficient collaboration between development teams."
>
> *Dominik Ott, Volkswagen AG*

Kirsten Pankratz
Kirsten Pankratz is responsible for virtual validation in the integration environment at Volkswagen in Wolfsburg, Germany.

Dominik Ott
Dominik Ott is responsible for virtual validation in the integration environment at Volkswagen in Wolfsburg, Germany.