



We have a clear vision of the future: We want vehicles to reach their destination autonomously, without a driver. But even the first steps towards autonomy are challenging: While vehicles are already partially autonomous, their autonomy is restricted to certain situations.

The sophisticated functions of a (partially) autonomous system, such as a highway pilot, are not only difficult to develop, but the system must also guarantee safe behavior beyond its intended function at all times, regardless of the circumstances. In real traffic, vehicles must handle countless situations that have to be validated by tests. Not all the tests can be carried out on the road. Even simulation solutions that work directly with the ECU in real time are currently overloaded by the sheer volume of data.

The Solution

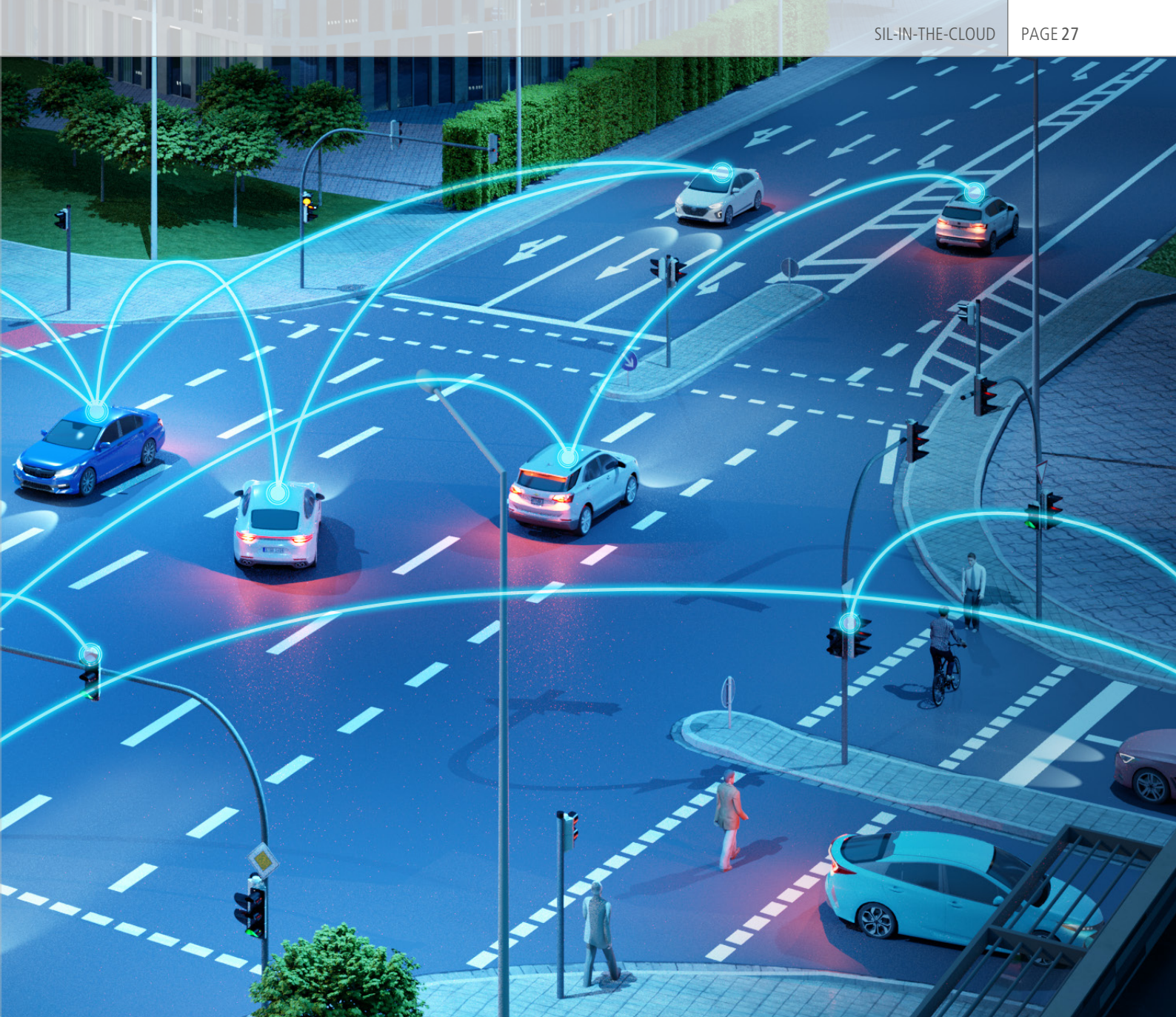
Our strategy for overcoming this issue rests on three pillars: the test objects, the simulation structure, and the test objectives.

Test Objects and Test Distribution

There must be central, critical tests that are carried out directly with the control unit under test and the control unit networks, or on the road. In order to perform the required number of tests, a substantial part of the tests has to be performed on software-in-the-loop (SIL) systems, requiring a virtualized test object. The solution focuses on the subject under test (SUT), which is the actual code of the functions to be tested. There are several technical approaches to providing the virtual test object: For example, you can integrate the code into the simulation system as a complete and executable unit via appropriate interfaces. dSPACE is working on implementing this type

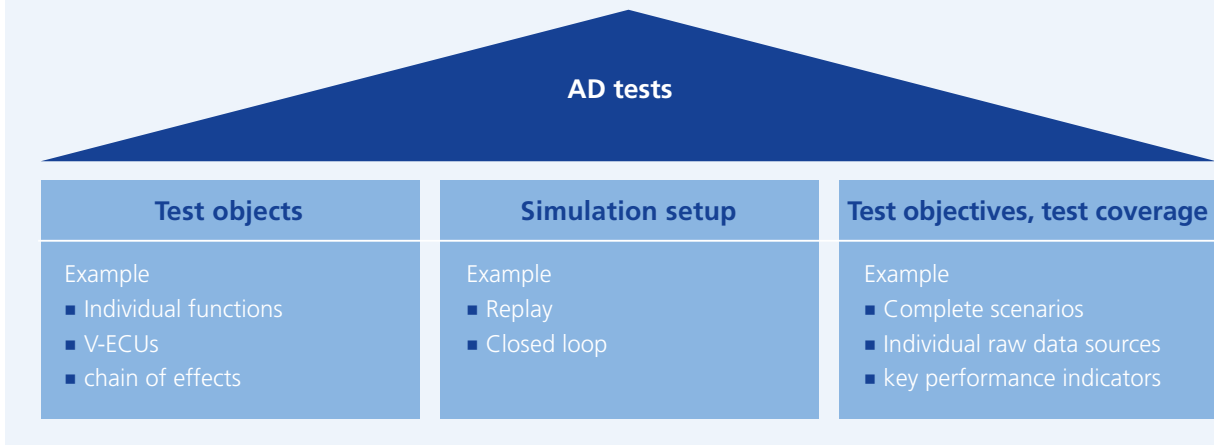


Did you know? In 2020, dSPACE tools will also operate on Linux systems



Taking Autonomous Driving in Stride

A story about scenario-based testing and SIL-in-the-cloud



The three pillars of the solution strategy for AD testing.

of integration with the help of container technology.

You can also integrate production code in particular as virtual ECUs (V-ECUs). dSPACE SystemDesk offers all the advantages of production code integration, from bus connection to operating system configuration. To determine how to virtualize the test object, the test scope must be precisely defined. Do you want to test individual functions, the complete ECU software, or a specific complete chain of effects? These factors ultimately determine the design of the SUT.

The Simulation Setup

The second pillar is the simulation setup and involves both the simulation system and its infrastructure. The use of SIL technology makes the simulation independent of real time and dedicated hardware. With VEOS, dSPACE offers a PC-based simulation and integration platform

that can serve as the basis for SIL tests. This versatility is particularly important for two reasons: First, the environment can be simulated with any degree of detail. In addition to motor or battery models, this particularly includes sensor models, which simulate all relevant levels of detail, from object lists to sensor-realistic data. Second, the goal is to easily multiply the test setup to achieve a high test throughput. Cloud systems, both public and customer data centers, use container and orchestration technologies to provide a platform for multiple parallel instantiation. dSPACE is working on seamlessly integrating its tools with cloud systems by providing preconfigured containers that contain, for example, a VEOS installation.

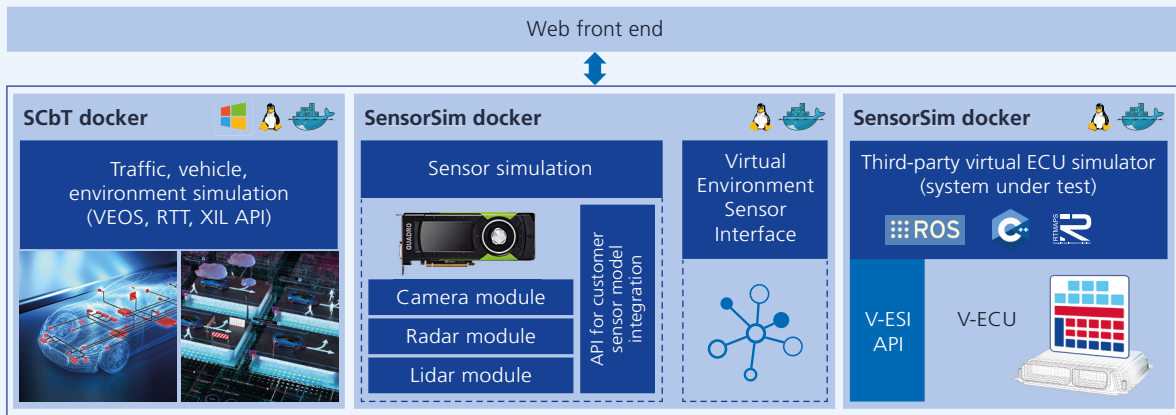
The Test Goals

This leaves the drastically increasing demand for testing. Ultimately, the

validation will be based on completing a range of specific traffic scenarios during simulation. This includes the simulation and variation of synthetic scenarios. But the replay of measurement data recorded during a test drive will also be a central part of the validation. First, the actual source of the tests or, based on the Pegasus method, the scenarios that a vehicle has to successfully complete must be identified. In principle, these are variations of a relatively small number of templates, called logical scenarios. For example, a certain avoidance situation in inner city traffic has to be tested in a wide variety of variants under different conditions, while the basic situation remains the same, for example, another vehicle unexpectedly changes lanes. Once a basic set of scenarios is available, it serves as a source for generating numerous specific test cases. Starting from the configuration of a log-

“The development of scenario databases, the origin of the scenarios, and their relevance to achieving the test objectives are questions that virtually all of our customers are currently asking.”

Karsten Krügel, Senior Product Manager Virtual Validation, dSPACE



SIL technology for sensor simulation.

ical scenario, an algorithm generates the final specific scenarios, which are then executed during simulation. Simple algorithms implement permutations, the simultaneous setting of individual parameters, or stochastic procedures. More advanced algorithms try to identify critical scenarios through optimization methods or artificial intelligence. Another aspect is often underestimated in the test setup: The entire process must be automated while enabling central configuration. The validation of the future will not focus on defining sophisticated test processes, but will hone in on the essentials, namely on measurable test case properties. These properties can be calculated from the measured values recorded for the simulation. The benefit: Their formulation is intuitive, for example, the relative speed of two vehicles driving behind one another can be derived directly from the respective speeds. These properties are calculated during or after the simulation, but have no effect on the actual test process. This makes it possible to always use a fixed test process in closed loop operation. It also eliminates the need for manual test step definition. ■

Did you know? dSPACE is also your partner for generating scenarios based on raw sensor data measured in real life.

Conclusion

The development of autonomous driving changes the basic models for cooperation between OEMs, classic suppliers, and platform providers. The focus is no longer on delivering a finished ECU, but on integrating distributed vehicle functions and validating them early on, possibly across companies. The basic principle is: The earlier you find an error, the cheaper it is to correct it. Access to shared simulation and test infrastructures enables new forms of collaboration, but ultimately represents only one of the many challenges of this collaboration. Validating the countless scenarios is one of the central challenges for

autonomous driving. Therefore, scenario-based testing at dSPACE is based on these three pillars. They consider many of the aforementioned aspects, but these can be very complex. Because dSPACE positions itself as a one-stop provider for validation, we offer solutions in each of these areas.

Our tool chain lets you complete the validation of functions or simulate ECU networks with just a few clicks, while ensuring high test coverage and scenario variation. It also lets you provide the developers with rapid feedback on their code quality at any time. This reduces the number of required test drivers and test drives to a manageable level.