>>> DIAGNOSTICS DEVELOPMENT PROCESS

OTX in practice

© emotive, adobe.stock.com/Kaspars Grinvalds

OTX is a standardized, domain-specific programming language (DSL) for the reliable and barrier-free exchange of testing knowledge in development, production, after-sales, and also within the vehicle. OTX plays a central role in resolving problems in diagnostics within a new electronics architecture.

We live in a networked world. Increasingly more demanding customer requirements and rising global competition lead to permanent pressure on all operating processes and procedures. Due to the resulting increase in complexity, companies face the challenge of continuously questioning and improving their operating processes. The quality of operating processes is revealed by the degree of „relaxed" mastery of complexity.

Here standards play a crucial role. Standards establish a uniform worldwide basis for the exchange of components. They are based on provable natural scientific arguments and pursue macroeconomic goals.

Germany in particular has established an outstanding position in worldwide standardization, especially in the field of automobile electrical/electronic systems. The large majority of new and innovative standards in this area was initiated in Germany and advanced with great dedication by experts in the respective specialized fields.

Figure 1. OTX libraries (Extensions). (© emotive)

OTX according to ISO 13209

The OTX standard is of special importance due to its integrative and harmonizing effect. OTX as a domain-specific programming language integrates seamlessly with existing standards for diagnostics communication, such as ODX and MVCI. Like hardly any other standard, OTX is also able to converge different, previously separate standards, even outside of traditional vehicle diagnostics. Examples include access to hardware-in-the-loop test stands (ASAM XIL), access to runtime systems for measuring and calibrating (ASAM MCD3-MC), and the description of tests in automated driving.

The goal of OTX is the reliable and barrier-free exchange and long-term availability of testing knowledge in development, production, and after-sales. OTX stands for „Open Test sequence eXchange" and is standardized in ISO 13209 as an exchange format. OTX is a meta-language for the description of executable test logic with verifiable quality. It is human and machine readable, and independent of technologies, service providers, and tool manufacturers. OTX is open, stable, and platform as well as technology-neutral. OTX strictly separates the test logic and runtime implementation. Active development of the standard is ongoing within ISO and ASAM, and it has broad-based tool support.

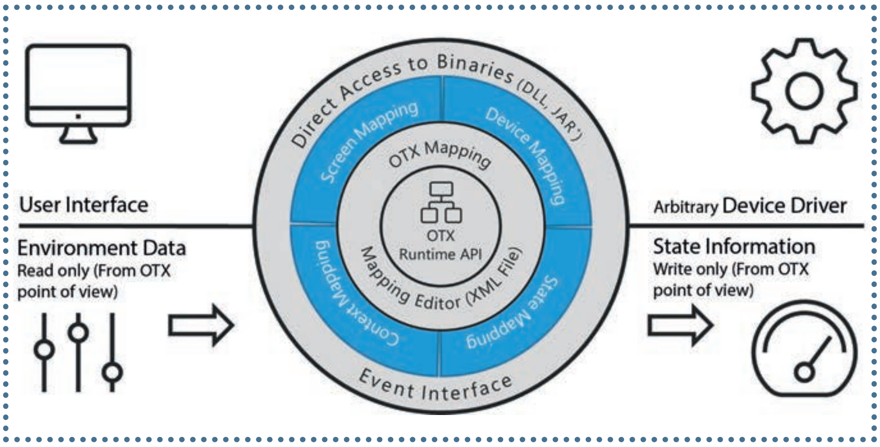

Figure 2. Linking external systems in OTX. (© emotive)

An OTX test sequence consists of one or more activities (also see Figure 3). All activities are thematically grouped in libraries, the OTX extensions. The core library contains all activities for the general test logic, such as procedure calls, assignments, branches, loops, activities for parallel execution, and error handling. All 36 OTX extensions add specific functions to the core, which is also capable of stand-alone execution (see Figure 1). In addition to extensions for vehicle diagnostics, HMI, and access to arbitrary external systems, there are numerous extensions that can map virtually any aspect for testing in the automotive industry. OTX is also readily expandable in compliance with standards.

Exchangeability beyond process limits

The exchangeability and long-term availability of testing knowledge are probably the most important features of OTX. Due to their widespread use, familiar programming languages (Java, C#, Python, etc.) have considerable advantages for specialized tasks on a few test stands, where exchangeability and long-term availability play a minor role. But when it comes to executing the same, quality-assured test logic in many different areas and a variety of environmental conditions, I believe OTX is indispensable. OTX handles the linking of all external systems in a very elegant way like hardly any other ISO standard.

»

Linking of external systems via OTX mapping

OTX fundamentally separates the test logic from the runtime implementation. Only one interface (signature) is described within the test logic. One of multiple runtime implementations can be assigned to this interface at the time of execution. EMOTIVE calls this process OTX mapping. It means linking the parameters of a signature to the property of an actually implemented class within a binary file in a graphical mapping editor. Thus OTX supports:

- Mapping to graphical interfaces (screen mapping), for example a WPF interface
- Mapping of arbitrary device drivers (device mapping), for example to query the ignition status via a DLL
- Mapping environmental data (context mapping), for example to determine the test location
- Mapping status information (state mapping), for example to output the flash progress (see Figure 2)

All required mapping data for a specific target environment are stored in an XML file. Solely by exchanging this file, the same test logic can be executed in various target environments. A nonexistent environment can also be simulated using what is called proxy mapping.

The highly simplified example shown in Figure 3 illustrates OTX mapping for various target systems based on a setup procedure for calibrating tire pressure sensors. First the sensor ID for various environments has to be input. In production this is automated via the test system, in development it is done manually using a barcode scanner. What exactly will be executed in the respective environment is parametrized in the context mapping. Next the input value is validated. If the value is plausible, a routine for calibrating the sensors is started. Subsequently the state of this routine is monitored and shown in parallel. The various environments are parametrized in the screen mapping. In production, representation is on a screen in the test system, while a generic OTX tester is used in development and the vehicle's infotainment system is used in the workshop. In the end the result is determined, logged, and output.

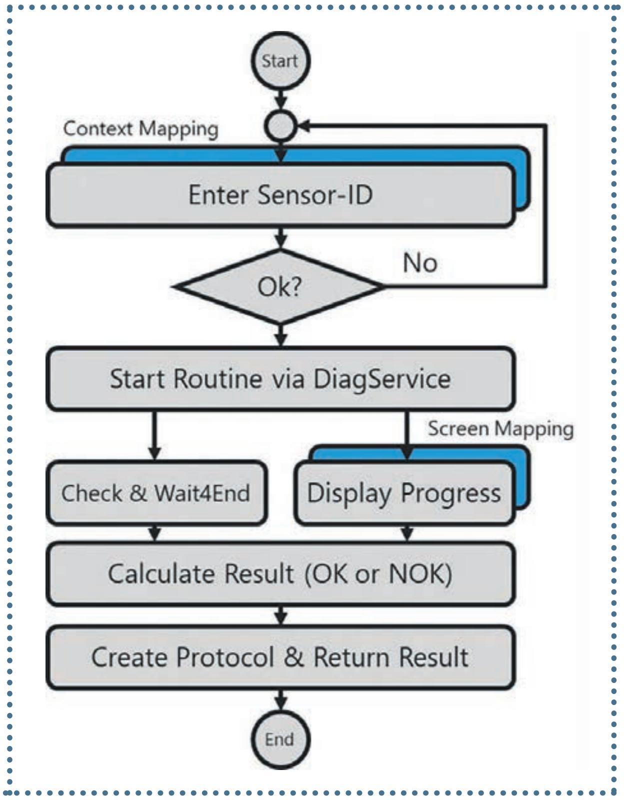

Figure 3. Application example: calibrating tire pressure sensors (simplified). (© emotive)

Figure 4 shows the process for the step-by-step development of this setup sequence. It begins with an abstract specification of the test logic, similar to Microsoft Visio. This results in an initial OTX document. In the realization phase, all specifications are implemented with the help of a test step library. The result is a complete, executable OTX document that is tested in the following step. This leads to a standardized PTX file containing all OTX documents, which can be shared with other departments. Each department now adds the environment-specific OTX mapping and generates what is called a PPX file. The PTX file is standardized and platform-neutral. The PPX file contains all PTX files and platform-specific information in the form of the OTX mapping. Subsequently the same PPX file can be executed in various target environments. In application 1, the sequence is executed within an existing

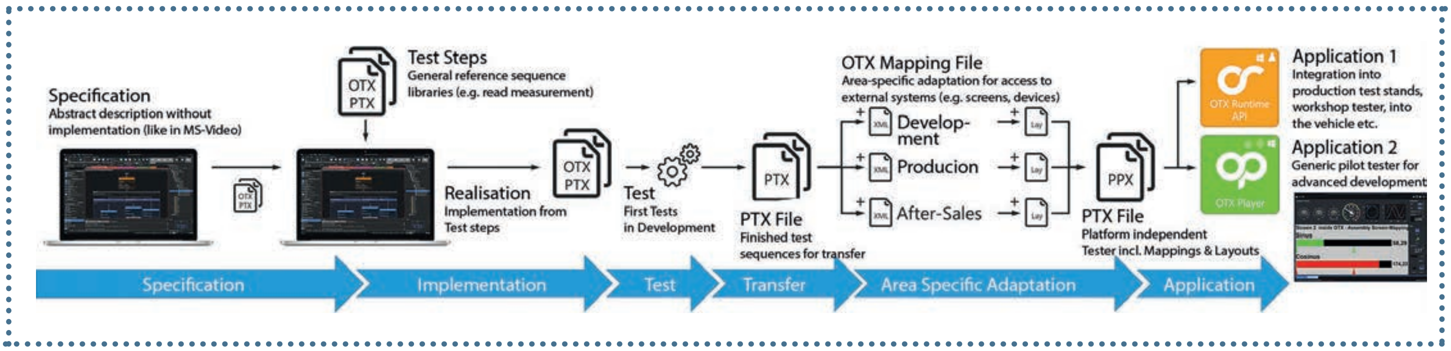

Figure 4. Step-by-step development of a setup sequence. (© emotive)

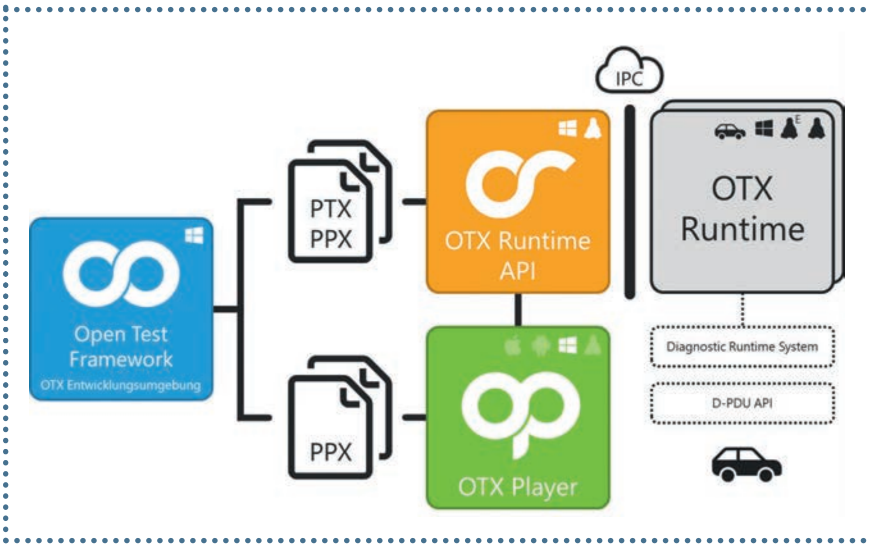

Figure 5: OTX toolchain by EMOTIVE. (© emotive)

system in production or after-sales. Application 2 executes the same PPX within a generic pilot tester in development.

Tool support

Numerous established manufacturers are offering software tools for OTX in the course of the nearly completed standardization process. The OTX toolchain from EMOTIVE shown in Figure 5 essentially consists of the OTX development environment „Open Test Framework" (Windows), the OTX runtime environment „OTX Runtime" (Windows and Linux), and the generic OTX tester „Open Test Player" (Windows, iOS and Android planned). It is currently in productive use by several vehicle manufacturers in development, production, after-sales, and also within the vehicle.

Conclusion

Like hardly any other standard in the automotive environment and also within traditional vehicle diagnostics, OTX is able to exchange quality-assured testing knowledge across process boundaries, with unchanged execution under a wide variety of environmental conditions. It makes the diagnostics development process more reliable and productive with the support of suitable software tools. ■ (oe)

» **www.emotive.de**

Dr.-Ing. Jörg Supke is the founder and CEO of emotive GmbH & Co. KG. He leads the German ISO working group for ODX, OTX, and MVCI at the VDA, is a member of the ASAM Technical Steering Committee, and leads various working groups for the standardization of OTX.