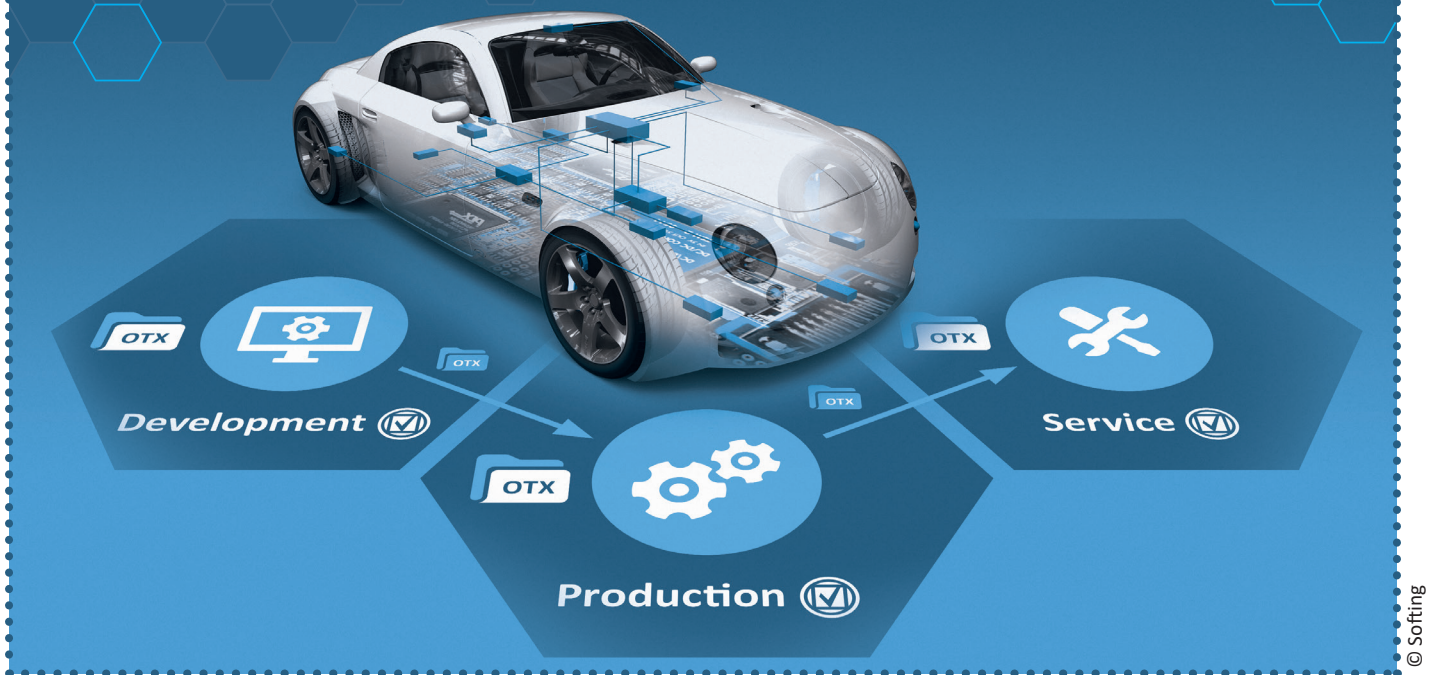


RE-USE OF SEQUENCES

optimize!
softing



© Softing

A Miracle of Diversity: OTX

In the automotive industry, diagnostics for ECUs and vehicle functions is constantly gaining in significance, after all, trends such as (semi-)autonomous driving demand increased predictability of the state of all components involved. An important basis for this is the programming language OTX (Open Test sequence eXchange) which, in the form of standard ISO 13209, describes diagnostic sequences for vehicles. One particular advantage is its suitability for a multitude of different use cases.

Today, there are numerous use cases for vehicle diagnostics: Particularly in the case of “smaller” ECUs, diagnostics is the first debug option in engineering; diagnostics is used to collect data in test systems; it enables ECU access in pre-production models; it is used for programming in all phases, and in manufacturing and after-sales service diagnostics is used to determine status as well as localize faults. However, to be able to use these opportunities, the diagnostic function must be developed and released in the ECU – as is the case for any other ECU and control function, too.

In many cases, release can still only be granted today after a series of manual tests. Several factors are responsible for this. For example, the fundamental variety of in-built ECUs often requires a different test structure to suit the particular variant. In this case, the effort involved in an automated test is very high. The continuous adaptation of equipment combinations necessitates the constant adaptations of test sequences: What may have been a reliable form of behavior yesterday, is no longer correct today. But diagnostics itself has changed considerably over the last decade. Whereas not many years ago, only a few ECUs were diagnostic- and update-capable, that is now normally the rule. While a few years ago di-

agnostics might have consisted of a single request which was sent and then answered by the ECU, today, the answers to a whole range of requests sent to the ECU have to be evaluated before the diagnostic statement can be said to be complete. In future, a further adaptation will be necessary, on the one hand because of the distribution of functions to different physical ECUs, on the other because of the integration of ECUs in high-performance domain systems.

OTX – The Original Idea

But this increased complexity also has further effects. When employees from engineering / development, manufacturing and service get together to determine the diagnostic scope of a new vehicle, they often talk at cross purposes: Requirements in terms of variability, performance, setup times and long-term availability are often too different. The awareness of this problem was one of the main reasons for passing the OTX standard: the determination of a language which specifically addresses the needs of the vehicle electronics domain for direct execution in the test systems, but which thanks to its formal description also permits graphic representation. Particularly the fact that sequences can be displayed, for example, as flow charts, offers developers from various areas a common level of communication. The fact that they can be run in test systems

makes a large number of diagnostic use cases possible over the entire life cycle of a vehicle.

The pure OTX doctrine envisages the direct execution of the specified diagnostic sequences as well as the transfer thereof to other users. The at times complex sequences of today's most important diagnostic functions are thus supported:

- Each individual ECU is queried for faults in fault memory reading with environmental conditions. The relevant environmental conditions are then determined individually for every fault detected.
- A security protocol first has to be run for flash programming. The ECU then switches to programming mode and the bus also switches to a special environment. The actual programming only takes place once this has been concluded.

Both examples are required in development / engineering, manufacturing and after-sales service in slightly different versions and in different test systems. Re-using is also advisable for efficiency reasons.

Use as a Test Language

A basic principle of OTX is the strict separation of core language and extensions. The core language comprises the actual linguistic elements, in other words branches, loops, etc. The extensions describe function libraries, such as diagnostics, mathematical functions, file operations. An essential feature

of the extensions is their extendibility. This means that the range of standardized libraries can be supplemented with additional extensions. The following two examples explain this in more detail.

In practice it can often happen that the pure diagnostic access is not sufficient for the test application because, for example, a power supply has to be powered on and off. This is why an interface from the company National Instrument was

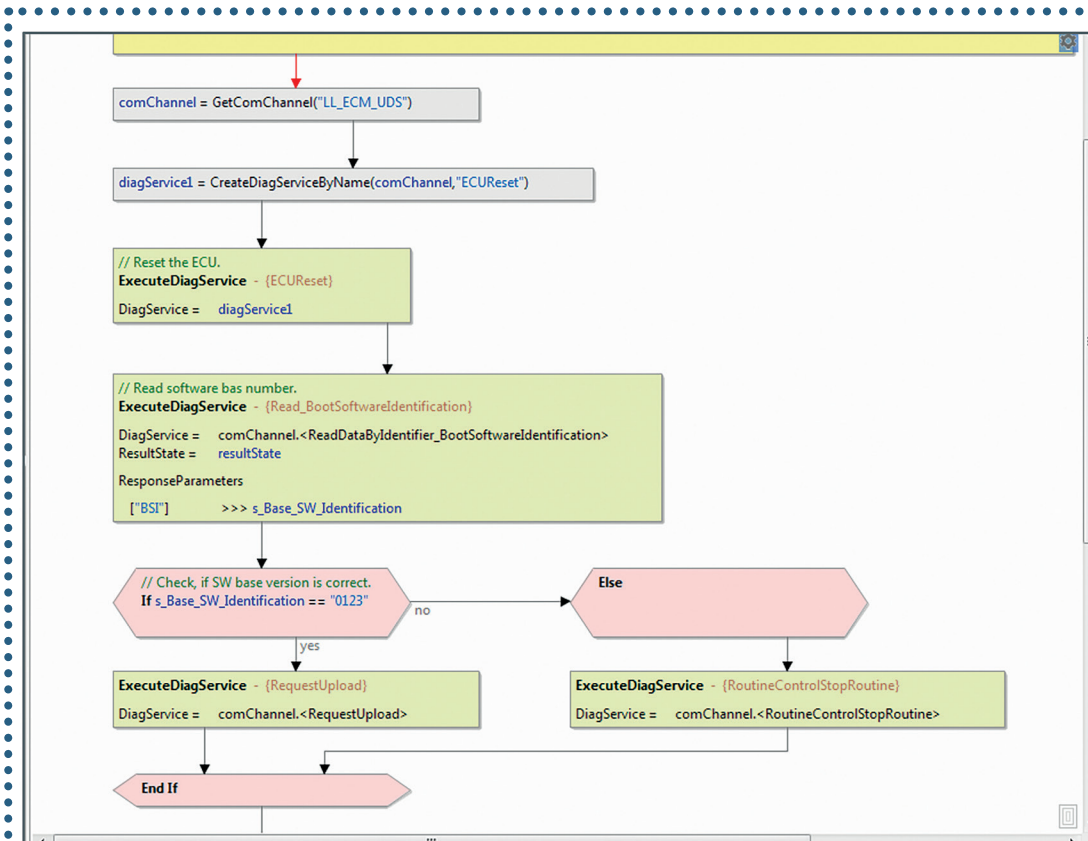


Figure 1: Graphic representation of sequences in Softing OTX.studio

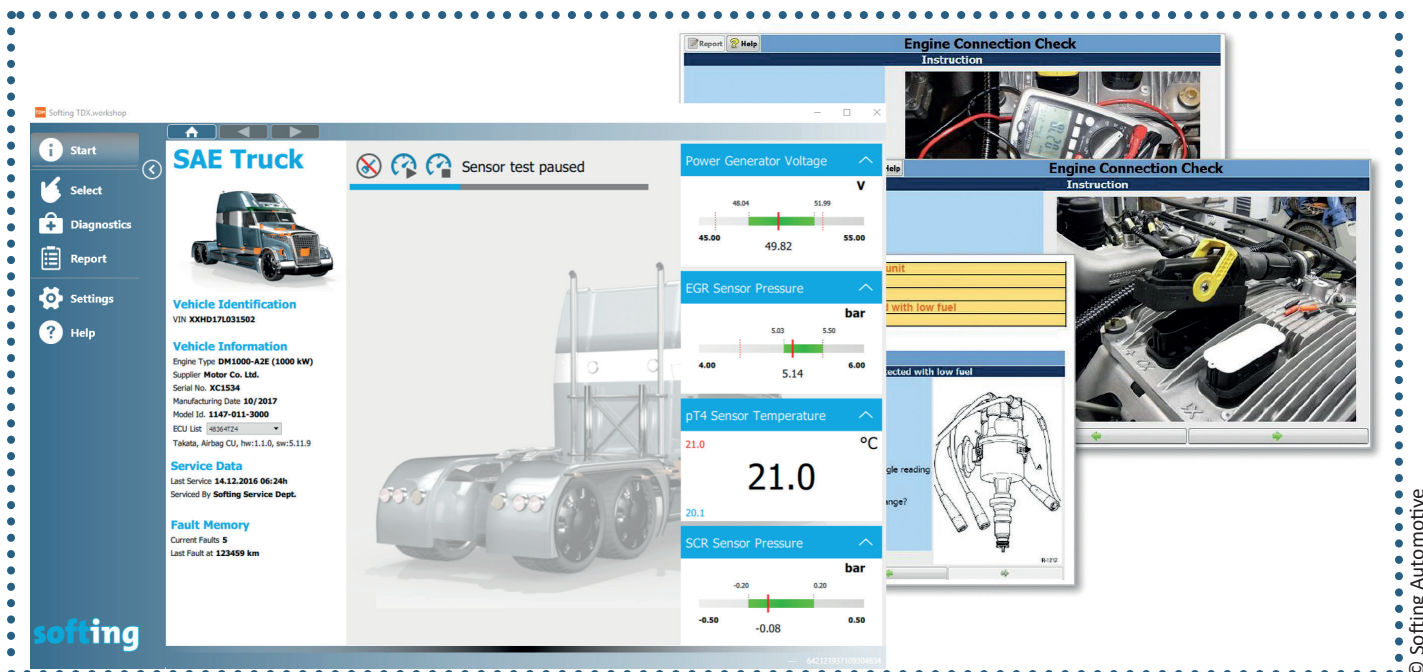


Figure 2: A comparison of rich GUI and base GUI

incorporated in Softing OTX.studio for accessing a large number of NI components. This makes it possible to represent a large number of scenarios which contain not only I/O boards but also adjustable voltage sources and measurement amplifiers. An ECU can thus be stimulated in different ways and the correct response to these stimuli can be verified, for example whether a fault memory entry takes place at the right time.

Integrating a suitable graphic library also opens up numerous possible application cases. Particularly in the case of tests which cannot be run fully automatically, user interaction can be integrated directly in the test sequence. This is often the case in tests on mechatronic systems when, for example, a control element in mechatronics has to be used before the test can be continued. But guided fault search in diagnostics also takes place in this way and the tester is guided step for step along the variables found to the cause of the fault. It is also simple to realize pure state monitoring in which only the status or a few measurement values are reported without interrupting operations using this kind of graphic. Even without additional extensions, OTX is perfect for use in diagnostic tests: A complete test of all diagnostic services with the possible parameterizations is possible with on-board aids and can be generated at the push of a button in relevant tools.

OTX in practice

Today, a tool such as Softing OTX.studio makes it possible for users to cover all kinds of different application cases: First of all the basic sequences can be determined graphically in a specification phase. These are then implemented in entirety by the sequence programmers. This means the diagnostic sequences only have to be created once. They can then be reused – also in different forms. The integrated GUI editors make it possible to extend the sequences at a later date with various user interactions, depending on the particular phase. Furthermore, libraries and integrated tools can be used in all application cases for automated diagnostic tests.

>> automotive.softing.com



Markus Steffelbauer heads up Product Management and Marketing at Softing Automotive and is a committed member of standardization bodies.