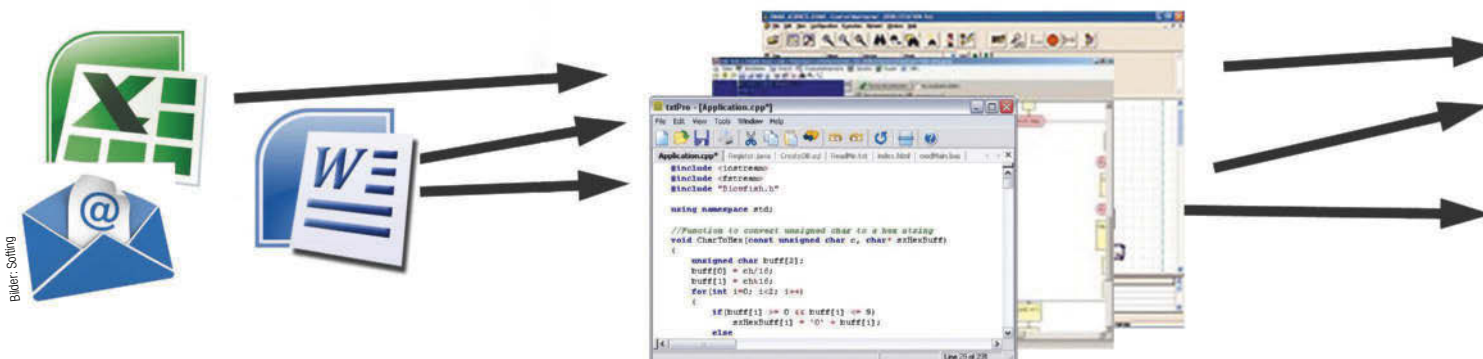


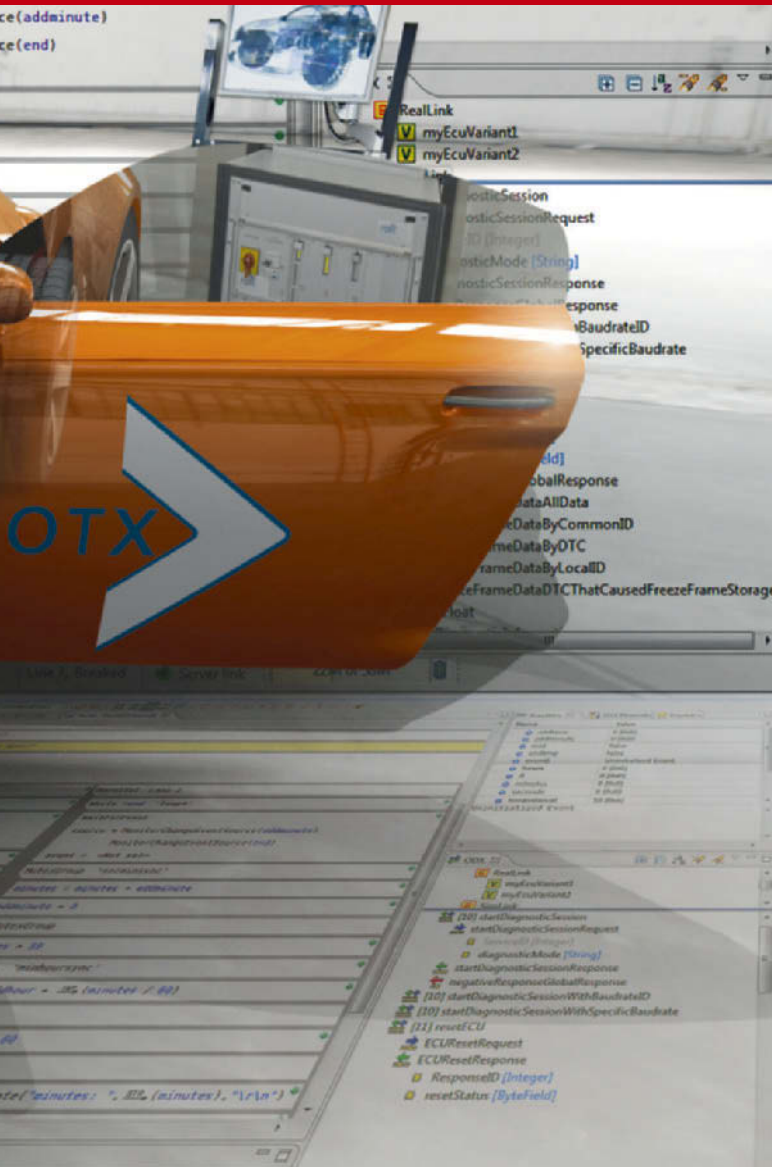
Diagnose mit OTX optimieren

Offboard-Diagnose mit Open Testsequence Exchange

Mit dem Standard OTX (Open Testsequence Exchange) hat die Automobilindustrie erstmals ein Sprachformat geschaffen, mit dem sich Diagnoseabläufe ohne Tool-Bruch spezifizieren und programmieren lassen. Das XML-basierte Format ermöglicht, validierte Abläufe über Abteilungs- und Prozessgrenzen hinweg auszutauschen und langfristig stabil abzulegen.

Autor: Matthias Meyer





Ablaufinformationen erstellt und teilt die Automobilindustrie heute typischerweise in frei formulierten und frei formatierten Dokumenten, die mit zahlreichen unterschiedlichen Werkzeugen bearbeitet werden. So spezifiziert ein Steuergeräteexperte die Prüfabläufe zunächst beispielsweise in Visio, Word oder Excel. Nach Fertigstellung der Spezifikation wird sie in ausgedruckter Form an einen Spezialisten für Testerprogrammierung weitergegeben. Dieser setzt die Testabläufe in die jeweilige Sprache seines gewählten Automatisierungswerkzeugs um. Es ist leicht einsehbar, dass ein derartiger Prozess ein hohes Fehlerrisiko enthält und dementsprechend teuer ist.

Hinzu kommt, dass auf diesem Wege erstellte Prüfabläufe nur bedingt wiederverwendbar sind. Aufgrund der Vielzahl der Spra-



Bild 1: Die Welt vor OTX: Verwendung vieler unterschiedlicher Formate zur Prüfabläuferstellung.

Auf einen Blick

Open Testsequence Exchange

Mit dem Einsatz von OTX lassen sich Prüfabläufe ohne Tool-Bruch zwischen vielen unterschiedlichen Bereichen der Wertschöpfungskette austauschen. Obwohl nicht ausschließlich für die Fahrzeugdiagnose konzipiert, enthält OTX alle im Umfeld der Diagnosekommunikation notwendigen Elemente für eine durchgängige Spezifikation und Implementierung von Abläufen.

Der Anwender ist damit in der Lage, schnell produktive Testanwendungen und Diagnose-abläufe zu erzeugen, die bei konsequenter Nutzung beträchtliche Kosteneinsparungen bewirken. Einheitliche Erstellung, langfristige Wiederverwendung und Austauschbarkeit validierter Abläufe sind damit sichergestellt.

 **infoDIREKT** www.all-electronics.de

324AEL0513

chen und Werkzeuge fehlt ein einheitliches, plattform- und technologieunabhängiges Format für Dokumentation und Ausführung von Abläufen. Daher lässt sich kaum sicherstellen, dass erstellte Prüfroutinen auch nach Jahren noch zuverlässig anwendbar sind.

Der neue OTX-Standard schließt diese Lücke in der ansonsten umfassend standardisierten Welt der Fahrzeugdiagnose: Bei Open Testsequence Exchange handelt es sich um ein XML-basiertes Austauschformat, mit dem Prüfabläufe spezifiziert, grafisch beschrieben, ausgetauscht, ausgeführt und archiviert werden können. OTX zielt also neben der einheitlichen Erstellung auch auf die langfristige Wiederverwendung und Austauschbarkeit erstellter Abläufe mit anderen Bereichen, Standorten oder Entwicklungspartnern ab.

Der OTX-Standard

OTX unterstützt den Anwender sowohl bei der Spezifikation seiner Abläufe als auch bei der nachgelagerten Implementierung. Als Spezifikationssprache erlaubt Open Testsequence Exchange das grafische Erstellen von Ablaufplänen mit frei formulierbaren Beschreibungstexten. Als Programmiersprache wird mit OTX diese Spezifikation in konkrete, ausführbare Einzelaktionen umgesetzt. Dafür sieht die OTX-Spezifikation Prozeduraufrufe, Aktionen, Verzweigungen und Schleifen vor, ebenso Elemente zur Fehlerbehandlung und nicht zuletzt Verarbeitungsmechanismen zur parallelen Ausführung von Abläufen.

Obwohl OTX viele Eigenschaften einer Programmiersprache mitbringt, handelt es sich nicht um eine klassische Programmiersprache mit eigener Syntax, denn OTX verwendet sowohl für die Spezifikation wie auch für die Implementierung ein standardisiertes Ablageformat. Hieraus ergibt sich der wesentliche Vorteil, dass dadurch eine Verarbeitung im PC deutlich einfacher zu realisieren ist. Wie auch andere Standard-Austauschformate nutzt OTX hierfür XML. Eine OTX-Laufzeitkomponente liest die Abläufe im XML-Format ein und kümmert sich um die Abarbeitung.

Zusätzlich zu den genannten Elementen zur Programmstrukturierung definiert der Standard in einem separaten Teil der Norm eine Reihe von Funktionsbibliotheken für bestimmte Einsatzfälle, die den Leistungsumfang der OTX-Ablaufsprache um wesentliche Elemente erweitern. Erst in diesem Teil des Standards kommt die Fahrzeugdiagnose ins Spiel. Das zentrale OTX-Datenmodell (OTX-Core) ist hiervon unabhängig, da OTX für die Verwendung in beliebigen Umgebungen entwickelt wurde.

Hier seien kurz einige wesentliche Standardbibliotheken genannt, die OTX mitbringt: Neben dem Zugriff auf die Fahrzeugdi-

agnose (einschließlich einer Bibliothek zum Flashen von Steuergeräten) enthält OTX Mittel zur grafischen Benutzerinteraktion (HMI-Library), eine Bibliothek mit mathematischen Funktionen, eine Erweiterung zum Umgang mit Mehrsprachigkeit (Internationalization, i18n) und Möglichkeiten zur Umrechnung von physikalischen Werten mit Maßeinheiten (Quantities).

Integration in die Fahrzeugdiagnose

Open Testsequence Exchange steht in enger Beziehung zu den bestehenden Standards ISO 22900 (MVCI) und ISO 22901 (ODX). Bild 2 zeigt, wie OTX sich in diese Standards integriert. Wie bereits angedeutet, heißt das nicht, dass OTX nur in diesem Kontext verwendet werden kann, aber es wird deutlich, dass der wichtigste Einsatzbereich von OTX die Fahrzeugdiagnose ist, um die eingangs aufgezeigte Lücke für die prozesssichere Beschreibung von Prüfabläufen zu schließen.

Der in der Bezeichnung ähnlich klingende Standard ODX (Open Diagnostic Data Exchange, ISO 22901) beschreibt die diagnose-

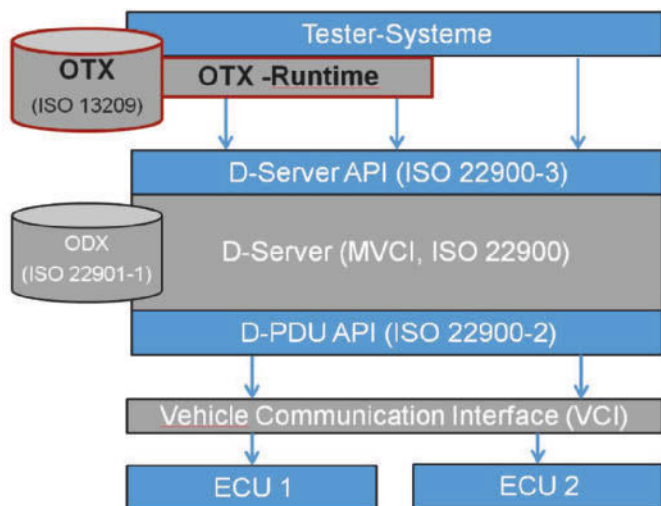


Bild 2: Integration von OTX in bestehende Standards zur Fahrzeugdiagnose.

relevanten Dienste und Daten eines Steuergerätes und verwendet hierfür ebenfalls ein Austauschformat in XML. Der MVCI-Standard (ISO 22900) beschreibt ein Diagnosesystem mit standardisierten Schnittstellen zur aufrufenden Applikation (D-Server API) und zur Protokollsoftware (D-PDU API).

Wird eine ODX-Beschreibung in den D-Server geladen, kann eine Testerapplikation über die D-Server-API darauf zugreifen. Der Anwender wählt den gewünschten Diagnosedienst aus, der als Kommunikationstelegramm an das Steuergerät gesendet wird. In ähnlicher Weise nutzt OTX die Kommunikation über die D-Server-API für die Verbindung zum Fahrzeug. Über die standardisierten Bibliotheken zur Fahrzeugdiagnose ist ein komfortabler Zugriff auf diese Schnittstelle möglich. Die Anbindung erfolgt dabei über die OTX-Runtime, die den D-Server aufruft und den Ablauf der Diagnosesequenzen steuert.

Abläufe spezifizieren und programmieren

OTX unterstützt einen fließenden Entwicklungsprozess. Zu einem frühen Zeitpunkt, wenn noch nicht alle Ablaufdetails bekannt sind, kann in der sogenannten Spezifikationsphase (Specification Stage) der künftige Prüfablauf bereits initial erstellt werden. Dabei wird er in einer freitextlichen Formulierung festgehalten, bereits im Zielformat XML gespeichert und ist austauschbar. Nach Abschluss der Spezifikationsphase beginnt mit einem geeigneten Edi-

tor die Befüllung der freitextlich definierten Abläufe mit Anweisungen, die durch das Laufzeitsystem (OTX-Runtime) verarbeitet werden können. Das Skript ist in diesem teilimplementierten Status (Intermediate Stage) bereits ausführbar. Nach Abschluss der Zwischenphase sind alle in der Spezifikationsphase formulierten Prüfschritte durch ausführbare Anweisungen ergänzt. Das fertige Skript ist nun vollumfänglich implementiert und vollständig ausführbar (Realisation Stage).

Da der Diagnoseexperte selbst seine Abläufe erstellen kann, verringert sich der Spezifikationsaufwand deutlich. In der Vergangenheit erforderliche Abstimmungsaufwände mit dem Experten für die Umsetzung in ein ausführbares Format fallen deutlich geringer aus oder sind gar nicht mehr erforderlich. Statt aufwändiger Erstellung von automatisierten Diagnoseabläufen in Programmiersprachen, die ein hohes Maß an Expertenwissen erfordern, lassen sich OTX-Skripte mit geringem Einarbeitungsaufwand deutlich kostengünstiger und zeitsparender erstellen.

Review

Ein Review oder eine Fortentwicklung von Prüfabläufen kann mit OTX einfacher und schneller erfolgen. Änderungen in unterschiedlichen Versionsständen der Prüfabläufe können leicht nachvollzogen und dokumentiert werden. Darüber hinaus ist die Spezifikationsansicht als Flussdiagramm ein geeignetes Medium zur Abstimmung unter den Beteiligten bei der Erstellung und Änderung von OTX-Skripten. Durch die hohe Verfügbarkeit von OTX-Abläufen lassen sich in der Praxis auch kleine Verbesserungspotenziale nutzen, da Skripte schnell anpassbar sind und rasch wieder in den produktiven Einsatz zurückgeführt werden können. In Summe sind durch diese Flexibilität signifikante Kosteneinsparungen möglich, da erkanntes Optimierungspotenzial schnell genutzt werden kann.

Als weiterer wichtiger Punkt ist auch zu sehen, dass die Spezifikation und die Implementierung in einem einzigen Dokument vereinigt sind. Dadurch wird die Festhaltung des fachlichen Wissens, das im Rahmen der Spezifikationsphase in die Ablaufferstellung einfließt, zu einem integralen Bestandteil des Entwicklungsprozesses bei der Erstellung der Diagnoseabläufe. Der OTX-Ablauf ist damit ohne zusätzlichen Aufwand für weitere Verwendungen lesbar dokumentiert.

OTX Studio

Um die Vorteile des OTX-Standards geeignet zu nutzen, ist eine passende Toolunterstützung erforderlich. OTX Studio von Softing ist eine Entwicklungsumgebung mit umfangreichem Funktionsumfang. Erstellte OTX-Abläufe werden mit allen zugehörigen Daten in Projekten zusammengefasst. Bei der Fehlersuche erlauben Debugging-Funktionen wie Haltepunkte, Einzelschrittverarbeitung und Variablen-Monitoring ein komfortables Arbeiten.

Je nach Anwendungsbereich besteht die Möglichkeit, zwischen verschiedenen Ansichten umzuschalten. Innerhalb dieser Ansichten kann der Anwender häufig benötigte Standardabläufe als eigene Bibliotheken ablegen. Ein integrierter OTX-Checker prüft zudem die Einhaltung der ISO-Konformität der erstellten Skripte und ermöglicht damit den reibungslosen Austausch von OTX-Abläufen. (av) ■



Der Autor: Matthias Meyer ist Product Manager bei Softing Automotive in Haar.