

Hochautomatisierte Test- & Integrationspyramide von Softwaremodultest bis Gesamtfahrzeugintegration

Max Leicht

max.leicht@bmw.de

+49 151 601 22367

Sebastian Dietze

sebastian.dietze@tracetronic.de

+49 351 205 768 248

BMW AG

BMW AG, Petuelring 130, 80788 München

tracetronic GmbH

tracetronic GmbH, Stuttgarter Str. 3, 01189
Dresden

Abstract

Die Durchführung von Software-, Integrations- und Systemtests sind elementarer Bestandteil der Fahrzeugentwicklung. Die stetig steigende Komplexität erfordert einen durchgängigen, effizienten, automatisierten und nachvollziehbaren Absicherungsprozess. Derzeit sind die Testprozesse und die in den einzelnen Entwicklungsstufen zum Einsatz kommenden Tools nicht vereinheitlicht und die Schnittstellen kaum standardisiert.

Dafür wird ein Plattformansatz vorgestellt, welcher sowohl virtuelle als auch reale Testumgebungen abdeckt und auf allen Integrationsstufen variabel einsetzbar ist. Der eine Teil der Plattform (Software Validation Platform) ermöglicht die kontinuierliche Validierung von Software sowohl in virtualisierten Testumgebungen mit Cloud-Infrastruktur als auch an klassischen Testinstanzen von einfachen Aufbauten bis zu Systemprüfständen. Das Ergebnis ist die Freigabe für die nächsthöhere Integrationsstufe. Der andere Teil der Plattform (Continuous Testing and Integration Platform) umfasst die gesamte Integrationspyramide vom Softwaremodul bis zur Gesamtfahrzeugintegration, denn kontinuierlich muss mit jeder neuen Softwareversion auch das Zusammenspiel überprüft werden. Damit dies möglich ist, werden zum einen standardisierte Schnittstellen für die Nutzung der Plattform und zum anderen geeignete Testautomatisierungstools sowie damit verbundene Prozesse vorgestellt, welche die gestellten Anforderungen an die durchgängige Automatisierung ermöglichen.

Schwerpunktthema: Testprozesse und Technologie

Neuigkeitswert:

Der vorgestellte Plattformansatz im Zusammenspiel mit den eingesetzten Automatisierungstools ermöglicht die kontinuierliche, nachvollziehbare und automatisierte Durchführung von Software- und Integrationstests in allen Stadien der Entwicklung (virtuelle Komponente bis Gesamtfahrzeugintegration).

Automotive Software Engineering & Automotive Testing – Kein One-fits-it-All

Die Automobilindustrie steht vor einem grundlegenden Wandel durch den Aufstieg von Software-Defined Vehicles (SDVs), in denen Software zunehmend zur treibenden Kraft wird. Im Gegensatz zur klassischen Fahrzeugentwicklung, die vor allem auf mechanische Systeme fokussiert war, rückt die Software nun in den Mittelpunkt der Innovation. Dabei unterscheidet sich die Entwicklung von Automobilsoftware grundlegend von der klassischen Softwareentwicklung, wie beispielsweise für Webshops. Während Webanwendungen oft in schnellen Iterationszyklen und mit weniger strikten Sicherheitsanforderungen entwickelt werden, erfordert Automotive Software strengste Sicherheitsstandards und Echtzeitfähigkeit. Fehler im Automobilbereich können fatale Folgen haben, was umfangreiche Testverfahren und zertifizierte Entwicklungsprozesse notwendig macht. Zudem müssen SDVs in einem hochkomplexen, vernetzten Ökosystem operieren, in dem Sensoren, Steuergeräte und cloudbasierte Dienste nahtlos zusammenarbeiten. Die Anforderungen an Skalierbarkeit, Zuverlässigkeit und Langlebigkeit sind hier deutlich höher als in klassischen IT-Projekten. Diese Unterschiede machen deutlich: Es gibt im Automotive Software Engineering kein „One-fits-it-all“-Ansatz. Jeder Aspekt, von der Softwarearchitektur bis hin zu den Testmethoden, muss auf die spezifischen Anforderungen der Automobilbranche zugeschnitten sein.

Automotive DevOps

Die Kernthemen von tracetronic sind Tests und deren Automatisierung in der Softwareentwicklung der Automobildomäne. Gestartet mit der reinen Testautomatisierung an Hardware-in-the-Loop Prüfplätzen im Bereich des Antriebsstrangs entwickelten wir uns über die Automatisierung der Testautomatisierung, die seit einigen Jahren unter dem Bezeichner Continuous Testing bekannt ist, hin zu Automotive DevOps.

Automotive DevOps beschreibt die Anwendung von DevOps-Prinzipien in der Automobilindustrie, wobei die spezifischen Anforderungen an Sicherheit, Compliance und Echtzeit-Funktionalität berücksichtigt werden. Im Vergleich zur klassischen IT ist Automotive DevOps besonders anspruchsvoll, da Software in komplexe Hardware- und Steuersysteme integriert werden muss, die hohen Sicherheits- und Qualitätsstandards wie ISO 26262 oder Automotive SPICE entsprechen. Der Fokus liegt auf kontinuierlicher Integration und Lieferung (CI/CD), um schnelle und zuverlässige Software-Updates zu gewährleisten, etwa für autonome Fahrfunktionen oder Over-the-Air (OTA) Updates. Diese Methode ermöglicht es Automobilherstellern, Entwicklungszyklen zu beschleunigen, Fehler schneller zu beheben und die Sicherheit von Fahrzeugen durch regelmäßige Software-Verbesserungen zu gewährleisten [[EclipseCon](#)] [[Intellias](#)] [[DZone](#)].

Automotive DevOps adressiert drei zentrale Herausforderungen in der Fahrzeugentwicklung: die Verkürzung der Feedbackzeit, die Optimierung der Testressourcen und -kosten sowie die zielgerichtete Aufbereitung von Feedback für verschiedene Stakeholder.

1. **Verkürzung der Feedbackzeit:** Durch den Einsatz von kontinuierlicher Integration und kontinuierlicher Lieferung (CI/CD) werden Software-Updates schneller getestet und freigegeben. Dies reduziert die Zeit, die zwischen der Entdeckung eines Problems und dessen Behebung vergeht, erheblich. Entwickler erhalten schneller Rückmeldungen zu neuen Codeänderungen, was die Entwicklungszyklen verkürzt [[EclipseCon](#)].
2. **Optimierung der Testressourcen und -kosten:** Automotive DevOps nutzt automatisierte Tests, um die vorhandenen Testressourcen effizienter zu nutzen. Virtuelle Testumgebungen,

die auf Cloud-Infrastrukturen basieren, ermöglichen es, Tests in einer kostengünstigen, skalierbaren Umgebung durchzuführen. So kann auf teure Hardwaretests in frühen Entwicklungsphasen verzichtet werden, während die Qualität sichergestellt bleibt [[Intellias](#)].

3. **Adressatengerechte Aufbereitung des Feedbacks:** Ein weiteres wichtiges Ziel von Automotive DevOps ist die Bereitstellung von auf die jeweiligen Zielgruppen abgestimmten Rückmeldungen. Entwickler benötigen detaillierte technische Informationen, während Projektmanager und Stakeholder oft einen Überblick über den Fortschritt und potenzielle Risiken erhalten müssen. DevOps-Lösungen sorgen dafür, dass jeder Beteiligte die relevanten Informationen in einer für ihn verständlichen Form erhält, was die Kommunikation im gesamten Entwicklungsprozess verbessert [[DZone](#)].

Insgesamt trägt Automotive DevOps dazu bei, die Entwicklungsprozesse zu beschleunigen, die Kosten zu senken und die Effizienz durch zielgerichtete Kommunikation zu steigern. Dafür ist es notwendig im Bereich Continuous Integration das Bereitstellen der Testobjekte mit der Durchführung von Continuous Testing eng zu verzahnen.

Während im Bereich der klassischen Softwareentwicklung mit GitLab und GitHub (und Azure DevOps) ein ganz klarer und einheitlicher Plattformansatz verfolgt wird, ist die Wahl einer einheitlichen Plattform oder Toolkette in der Automobilsoftwareentwicklung nicht einfach. Diverse Programmierumgebungen, Architekturen, Kommunikationskanäle und die damit verbundenen Anforderungen und Möglichkeiten machen eine einheitliche Toolentscheidung schwer bis unmöglich. Während in softwaregeprägten Bereichen wie dem Infotainment cloud-basierte state-of-the-art Lösungen nahezu anpassungslos übernommen werden können, hat die Absicherung des Antriebsstrangs auch heute noch starke Abhängigkeiten zur verwendeten Zielhardware. Dieser Spagat ist in den vorhandenen Marktlösungen meist nur schwer möglich. Tools, die aus der Automobilwelt kommen, sind zumeist nicht für die enormen Testdurchsätze optimiert. Tools aus der klassischen Softwareentwicklung dagegen sind mit der Vielzahl an Hardwareabhängigkeiten, Echtzeit- sowie Securityanforderungen überfordert.

Für die Entwicklung einer Automotive DevOps Plattform ergibt sich als essentielle Anforderung die Unterstützung einer Vielzahl an notwendigen Tools. Es bedarf eines Single-Point-Of-Truth mit einer Oberfläche die alle Informationen gemeinsam aufbereiten kann und einen guten Trade-off aus Generik und Spezialisierung anbietet, gesucht wird eine Möglichkeit eine Richtung vorzugeben, ohne dabei die notwendige Individualität einzuschränken.

Über vereinheitlichte Konzepte soll es ermöglicht werden, Nachvollziehbarkeit und Identifizierbarkeit von allen Objekten und Artefakten im Entwicklungs- und Testprozess zu erreichen. Zentraler Bestandteil dafür ist ein gemeinsames Build- und Abhängigkeitsmanagement. Dieses System muss im weiteren Verlauf lückenlos in die Nachfolgeprozesse integriert werden, um bspw. im Testprozess aufgetretene Fehler, auch nach dem Kompilieren und Verpacken, während der Verwendung auf einem Steuergerät, noch der richtigen Software-Unit in der richtigen Version zuordnen zu können. Dies ist eine elementare Anforderung, um hochautomatisierte Testprozesse über mehrere Teststufen zu ermöglichen. Außerdem muss ein Weg gefunden werden, dass verwendete Tools und Technologien auch nach der initialen Versionsdefinition noch aktualisiert werden können, um keine Sicherheitsrisiken zu erzeugen und auch nach dem ersten Release der Fahrzeugsoftware noch eine exzellente Entwicklungsumgebung zu schaffen.

Solange grundlegende Konzepte vorhanden sind, Nachvollziehbarkeit, Identifizierbarkeit und Reproduzierbarkeit sichergestellt werden kann und eine grundsätzlich gemeinsam verwendete

Umgebung bereitgestellt wird, kann von einer Automotive DevOps Plattform gesprochen werden. Der Anwender ist dadurch in der Lage seinen Fokus nicht auf Themen wie das Kompilieren oder Verteilen von Softwarefragmenten zu legen, sondern direkt an der Wertschöpfung und damit an der Modellierung oder Implementierung von Fahrzeugsoftware zu arbeiten. So entstehen schnell neue Endnutzerfunktionen – aber sicher.

Fallbeispiele

Im Rahmen des Vortrags wird nicht nur theoretisch beleuchtet, welche Anforderungen an eine Automotive DevOps Plattform für die Entwicklung und Absicherung moderner Automobilsoftware gestellt werden. Im Vortrag gibt es zusätzlich Einblicke in Konzepte, Automatisierungsmöglichkeiten und allgemein aus dem Testalltag. Stay Tuned.

Literaturverzeichnis

- [[EclipseCon](#)] <https://www.eclipsecon.org/2023/sessions/orchestration-automotive-devops-pipelines-chances-and-constraints-open-source>
- [[Intellias](#)] <https://intellias.com/devops-cloud-computing-automotive/>
- [[DZone](#)] <https://dzone.com/articles/the-scene-of-devops-in-the-automotive-industry>