

Wenn Software Software schreibt

04/08/2021 Software wird immer mehr zum bestimmenden Faktor in der Automobilentwicklung. Marius Mihailovici, Geschäftsführer von Porsche Engineering Romania, wirft in seinem Beitrag einen Blick in die Zukunft der Softwareentwicklung und erklärt, warum sich der Job des Programmierers in den kommenden 20 Jahren komplett verändern könnte.

Autos sind schon heute rollende Computer. In ihnen arbeitet ein Netzwerk aus Dutzenden Recheneinheiten: Zwischen 70 und 100 dieser Electronic Control Units (ECUs) sind in modernen Fahrzeugen verbaut. ECUs steuern die Kraftstoffeinspritzung, regeln das Bremsverhalten oder überwachen die Klimaanlage. Der nächste Schritt werden HCPs (High Performance Computing Platforms) sein, die deutlich mehr Rechenleistung in einem Steuergerät ermöglichen.

Die höhere Rechenleistung und die Integration sind notwendig, da die Anzahl der Codezeilen und die Komplexität der Funktionen im Fahrzeug von Jahr zu Jahr ansteigen. Eine Zahl mag das verdeutlichen: 100 Millionen. So viele Zeilen Code stecken in einem modernen Auto. Zum Vergleich: Eine Boeing 787 Dreamliner bringt es gerade einmal auf 14 Millionen Zeilen. All diese unterschiedlichen Herausforderungen führen dazu, dass schnelles und flexibles Arbeiten gefragt ist. Ziele werden oft kurzfristig gesetzt und ändern sich schnell. Man kann sagen: Software wird ergebnis getrieben entwickelt.

In Cluj arbeiten unsere Softwareentwickler mit Tools, die sich seit 20 Jahren in der Softwareentwicklung bewahrt haben. So nutzen wir – wo immer möglich – agile Methoden. Sie beruhen auf kleinen Entwicklungsschritten, deren Ergebnisse in taglich stattfindenden Feedbackrunden überprüft werden. Die einzelnen Teams haben dabei große Freiheit und arbeiten direkt miteinander, um sich mit ihrer jeweiligen Kompetenz unterstützen zu können. Sie legen ihre Ziele taglich und flexibel selbst fest. Teamleiter haben hier oft nur noch moderierende Funktion und behalten das große Ganze im Auge.

Auch hinter dem Entertainmentsystem und der Navigation stehen viele Zeilen Code. Hinzu kommt die Möglichkeit, Smartphones und andere Geräte mit dem Auto zu verbinden, was ebenfalls nur durch komplexe Software möglich ist. Und dabei bleibt es nicht. Software übernimmt immer weitere wichtige Aufgaben im Auto. Zu den wichtigsten heutigen und künftigen Funktionen gehören der Datenaustausch mit anderen Verkehrsteilnehmern und der Infrastruktur, das Update der Fahrzeuge aus der Cloud und schließlich sogar das autonome Fahren.

Dabei findet die Entwicklung von Automotive-Software in einem besonderen Spannungsfeld statt. Es gilt, Sicherheitsbestimmungen und Kundenvorgaben in Form großer Lastenhefte zu erfüllen. Die klassischen Entwicklungsprozesse der Branche sind in der Regel zeitgetrieben: Es gibt einen vorgegebenen Zeitplan, der bestimmte Meilensteine definiert. Unsere Auftraggeber erwarten regelmäßige Ergebnisse zu vorher festgelegten Terminen. Hinzu kommen die behördlichen

Zulassungsprozesse, die sogenannte Homologation. Sie sieht zum Beispiel vor, dass Fahrzeuge erst auf den Markt kommen, wenn sie in einer bestimmten Stückzahl gebaut und abgenommen worden sind. Letztendlich geht es in der Branche immer darum, sich von einem „fertigen“ Zustand zum nächsten zu bewegen.

Ein weiteres aktuelles Paradigma der Softwareentwicklung ist „Continuous Integration“. Darunter versteht man ein stark automatisiertes Verfahren, mit dem Softwareelemente am Ende eines Arbeitstages auf ihre Lauffähigkeit überprüft und in das Gesamtsystem integriert werden. So können wir Fehler und Probleme schnell erkennen und beheben. Der Einsatz agiler Methoden und von Continuous Integration sorgt für mehr Effizienz und bietet auch unseren Kunden einen Mehrwert: Er erleichtert die Präsentation von Zwischenergebnissen und gibt ihnen die Möglichkeit für schnelles Feedback.

Für die kommenden fünf Jahre gehe ich von einem zunehmenden Einsatz der oben beschriebenen Methoden aus. Wir werden noch stärker ergebnis- und nicht zeitgetrieben arbeiten. Teamhierarchien dürften an Bedeutung verlieren, und ich sehe einen höheren Grad an flachen und selbstorganisierenden Teams mit klaren Verantwortlichkeiten voraus. Zudem wird es wohl selbstverständlich sein, flexibel von verschiedenen Orten aus zu arbeiten (gemäß dem Konzept: „Einstellen, wo die Kompetenz existiert“). In den nächsten Jahren werden wir außerdem eine zunehmende Automatisierung sehen: Immer häufiger wird Software von Software getestet und nicht mehr von Menschen. Manuelle Tests werden vollständig verschwinden.

Die Grenzen zwischen Auto und Umwelt verschwimmen

Entwicklungen außerhalb der Automobilbranche werden uns ebenfalls zum Umdenken zwingen, denn Autos sind zunehmend ins digitale Leben ihrer Fahrer integriert. So verbindet sich zum Beispiel das Smartphone beim Einsteigen automatisch mit dem Fahrzeug. Mediennutzung, Navigation und Kommunikation gehen nahtlos ineinander über.

Es sind solche Nutzungsszenarien, die die Arbeit unserer Softwareentwickler inhaltlich verändern werden – einfach, weil die Grenzen zwischen der reinen „Autosoftware“ und anderen Anwendungen verschwimmen. Das erfordert übrigens auch ein bestimmtes Mindset bei unseren Entwicklern: Bei uns arbeiten Menschen, die selbst den digitalen Lifestyle leben. Sie wissen nicht nur, was unsere Auftraggeber fordern, sondern auch, was die Fahrzeugnutzer von ihrem Auto erwarten.

ECUs der Zukunft beeinflussen die Softwareentwicklung

Auch neue Steuergeräte-Architekturen verändern die Art, wie Automotive-Software entwickelt wird. Ich gehe davon aus, dass es im Auto der Zukunft einige wenige zentrale, sehr leistungsfähige Rechner vom HCP-Typ gibt, die zusammen mit nachgeordneten, einfacheren Einheiten das gesamte Fahrzeug steuern. Auf diesen zentralen Einheiten laufen auch alle Anwendungen, die über Basisfunktionen

hinausgehen, also zum Beispiel Entertainment, Datenverkehr oder die Kommunikationsanwendungen der Insassen. Diese zentralen Rechner mit einem echten Betriebssystem machen das Auto dann tatsächlich zu einem PC auf vier Radern.

Für Entwickler heißt das, dass sich die Methoden ihrer Arbeit gar nicht so sehr verändern. Was sich ändert, sind die Systeme, mit denen sie es zu tun haben: Sie sind hierarchischer, haben weniger Komponenten und werden von einer Gesamtsoftware gesteuert. Und diese wird – wie jede Software – regelmäßige Updates erhalten. Die Software eines Autos wird also nicht mehr einmal entwickelt und installiert, sondern ständig weiterentwickelt, auch wenn das Auto schon längst beim Kunden ist.

Eine Zukunft ohne Softwareentwickler?

Die Frage ist gar nicht so abwegig: Wird es in Zukunft überhaupt noch Softwareentwickler geben? Manche Experten gehen davon aus, dass Künstliche Intelligenz (KI) in den kommenden zwei Jahrzehnten die Softwareentwicklung komplett übernehmen wird – auch in der Automobilbranche. Allerdings gehen die Meinungen hierüber auseinander. Ich persönlich kann mir vorstellen, dass wir weiterhin die Rahmenbedingungen vorgeben und die KI diese dann umsetzt. Wir werden immer noch Software-Architekten, Anforderungsingenieure und Software-Ingenieure brauchen, um zu definieren, was KI beziehungsweise neuronale Netze tun werden: Softwarefunktionen auf der Grundlage der definierten Anforderung generieren, kontinuierlich automatisch testen und korrigieren, bis die Softwarequalität auf dem erwarteten Niveau ist.

Die Entwickler der Zukunft müssen darum in der Lage sein, in Gesamtsystemen zu denken. Sie müssen wissen, wie Endkunden ihre Fahrzeuge nutzen. Ihre Arbeit wird nicht nur von den Vorgaben der OEMs bestimmt, sondern immer stärker von den Konsumenten. Eines allerdings kann ich mir nicht vorstellen: eine Software ganz ohne „Bugs“. Fehler wird es immer geben. Wir müssen sie nur nicht mehr selbst finden und beheben: Das künftige neue System wird das erledigen, überwacht von menschlichen Softwareexperten.

Marius Mihailovici

Marius Mihailovici ist seit 2016 Geschäftsführer von Porsche Engineering Romania. Zuvor leitete er Forschung und Entwicklung bei Alcatel-Lucent S.A./Nokia Oyj, wo er auch als Software-Manager für 2G- und 3G-Projekte tätig war. Bei World Telecom arbeitete er als Ingenieur für Nachrichtentechnik.

Info

Text: Marius Mihailovici

Text erstmalig erschienen im Porsche Engineering Magazin, Nr. 1/2021

**MEDIA
ENQUIRIES****Inga Konen**

Head of Communications Porsche Schweiz AG

+41 (0) 41 / 487 914 3

inga.konen@porsche.ch

Consumption data**Taycan Turbo S**

Fuel consumption / Emissions

WLTP*

CO-Emissionen* kombiniert (WLTP) 0 g/km

Stromverbrauch* kombiniert (WLTP) 23,4 – 22,0 kWh/100 km

Elektrische Reichweite kombiniert (WLTP) 440 – 467 km

Elektrische Reichweite innerorts (WLTP) 524 – 570 km

*Further information on the official fuel consumption and the official specific CO emissions of new passenger cars can be found in the "Leitfaden über den Kraftstoffverbrauch, die CO-Emissionen und den Stromverbrauch neuer Personenkraftwagen" (Fuel Consumption, COEmissions and Electricity Consumption Guide for New Passenger Cars), which is available free of charge at all sales outlets and from DAT (Deutsche Automobil Treuhand GmbH, Helmuth-Hirth-Str. 1, 73760 Ostfildern-Scharnhausen, www.dat.de).

Link Collection

Link to this article

https://newsroom.porsche.com/de_CH/2021/innovation/porsche-engineering-wenn-software-software-schreibt-25354.html