

# **WANTED** Neue Entwicklungsmethoden für autonomes Fahren

Das Rennen um das autonome Fahren ist in vollem Gange und beschäftigt die Entwicklungsabteilungen der Automobil- und der IT-Industrie gleichermaßen. In diesem Beitrag werden eine Reihe von Innovationen zur Entwicklung und Absicherung automatisierter Fahrfunktionen vorgestellt: Erweiterungen des sogenannten V-Modells, skalierte agile Methoden, Model-based Systems Engineering und der datengetriebene Entwicklungsprozess.

Von JENS KRÜGER

Elon Musk sagte 2014, dass die Technologie für komplett eigenständig fahrende Fahrzeuge in fünf bis sechs Jahren vorhanden sein werde, was eine Serienreife ab 2020 nahelegt. Heutiger Stand ist: Mit Adaptive Cruise Control, Parkassistenten oder Staupiloten sind mittlerweile erste Fahrzeuge verfügbar, die automatisiertes Fahren auf Level 3 möglich machen. Level 3 bedeutet: Das Auto übernimmt für bestimmte Situationen die Kontrolle, der Fahrer muss aber jederzeit eingreifen können. Das bedeutet jedoch nicht das Ende der Fahnenstange, denn die Automotive-OEMs arbeiten mit Hochdruck an der Entwicklung weiterer Funktionen für das autonome Fahren.

Fahrerassistenzsysteme (Advanced Driver Assistance Systems, ADAS) für das autonome Fahren erfordern die Integration mehrerer neuer Komponenten in das Fahrzeug – allen voran Sensoren: Die Augen und Ohren eines menschlichen Fahrers werden im selbstfahrenden Auto durch Sensorik ersetzt. Sensoren ermöglichen es, ein Bild der Umgebung zu generieren. Um auch bei Defekten und ungünstigen Umweltbedingungen wie Regen und Schnee eine Abbildung des Fahrzeugumfelds zu erhalten, ist eine redundante Auslegung der Sensoren erforderlich. Mit der Kombination mehrerer Sensortypen (Lidar, Radar, Kameras,

Ultraschall) und der Fusion der erzeugten Sensordaten wird es möglich, das Umfeld sehr zuverlässig zu erfassen und entsprechend das Fahrverhalten des autonomen Fahrzeugs abzuleiten.

Die Verarbeitung der Sensordaten umfasst neben der Datenfusion insbesondere das Erkennen der Umwelt und die Ableitung entsprechender Fahrentscheidungen. Dazu werden leistungsfähige Rechner mit vorab trainierten neuronalen Netzen genutzt (Stichwort: maschinelles Lernen).

Die Sensordaten können an ein IT-Backend-System übertragen, dort verarbeitet und dann wieder an das Fahrzeug zurückgesendet werden. Dies eröffnet neue Möglichkeiten, zum Beispiel die Zusammenführung der generierten Daten mit zusätzlichen, durch weitere Fahrzeuge in der Umgebung zeitgleich gesammelten Daten.

Eine weitere Komponente sind hochpräzise, dynamische Echtzeitkarten. Sie steuern das Fahrzeug mit einer Genauigkeit im Dezimeterbereich und sind unverzichtbar für das autonome Fahren, da sie eine Vorhersage der Steuerung über die Reichweite der Sensoren hinaus erlauben. Echtzeitkarten finden die bestmögliche Route und enthalten aktuelle Daten über Verkehrsregeln, Straßenschilder, mögliche Hindernisse oder Straßenverhältnisse.

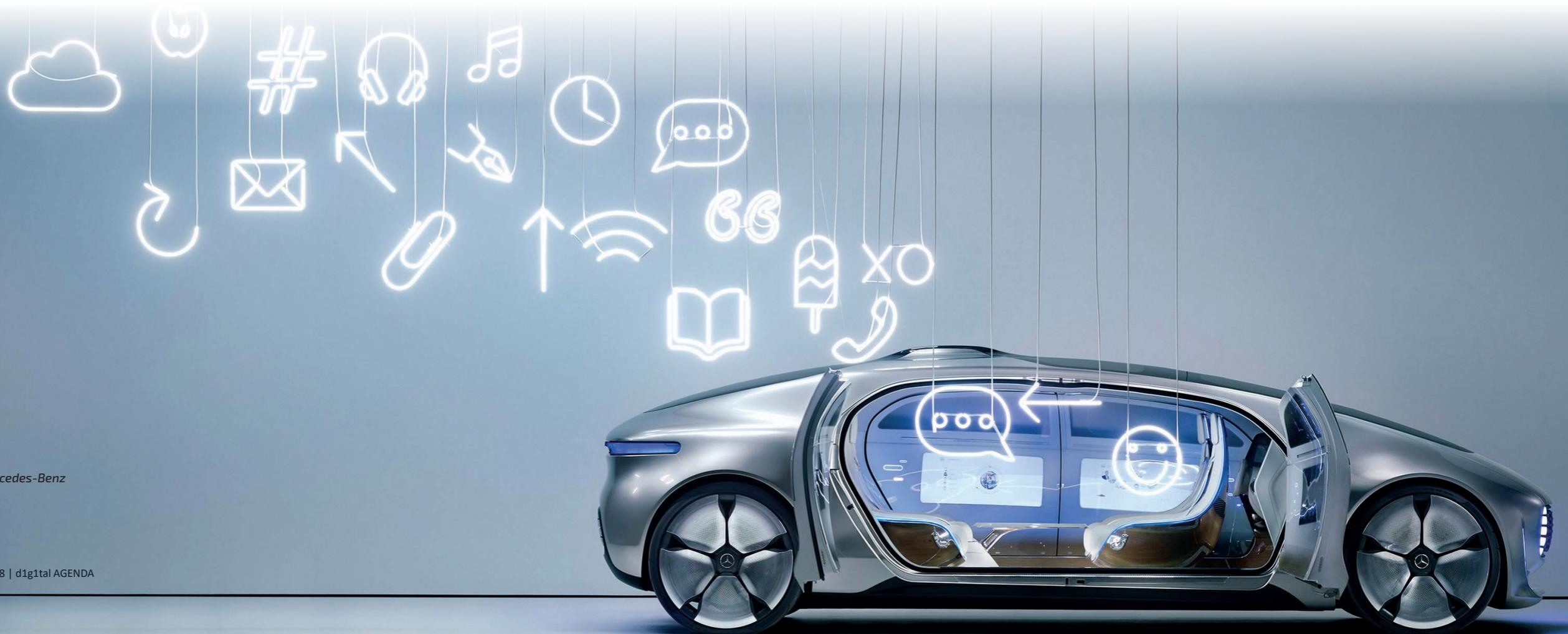
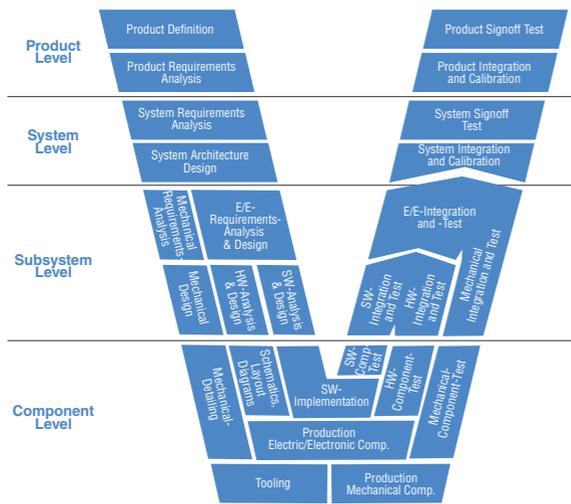


Bild: Mercedes-Benz



Konventionelles V-Modell

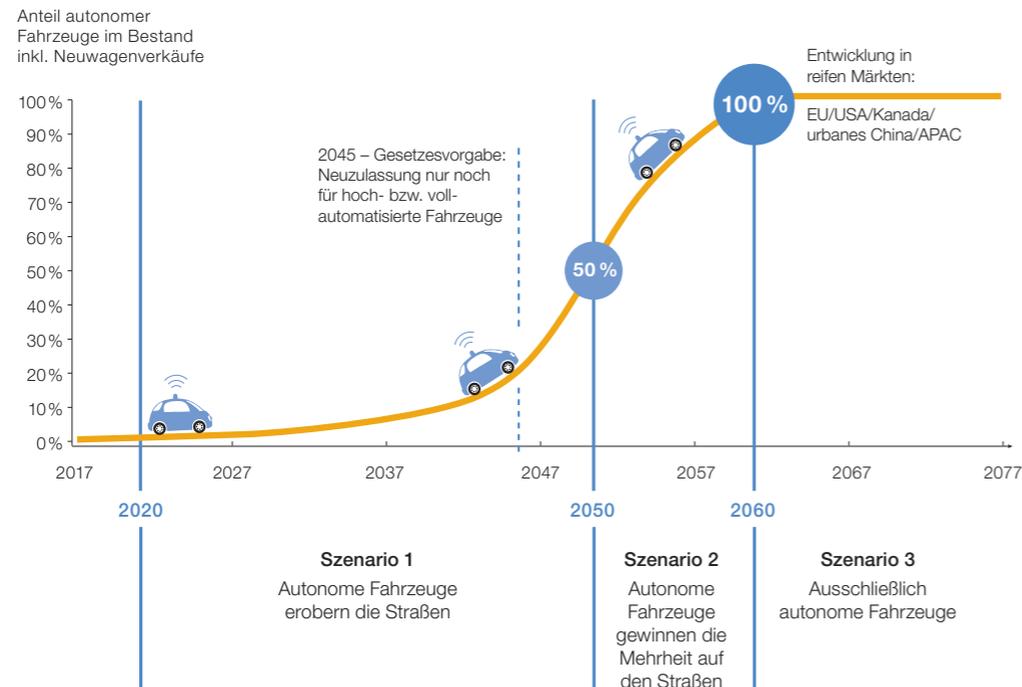
Quelle: prostep ivip / Smart Systems Engineering

Neuer Entwicklungsprozess für ADAS

Die Entwicklung von komplexen mechatronischen Systemen in der Automobilindustrie basiert normalerweise auf Methoden aus dem Systems Engineering und wird als V-Modell dargestellt. Die Entwicklung vernetzter, KI-basierender Systeme erfordert allerdings einige Änderungen und Erweiterungen im Entwicklungsprozess.

**Neues V-Modell.** Das konventionelle V-Modell gemäß VDI-Richtlinie 2 206 beschreibt im linken Ast die Entwicklungstätigkeiten von der Produkt- über die System- bis hin zur Komponentenebene. Für die Komponententwicklung gibt es die Disziplinen Mechanik, Elektrik/Elektronik und Software. Im rechten Ast finden sich die entsprechenden Integrations- und Verifikations-/Validierungstätigkeiten auf diesen drei Ebenen.

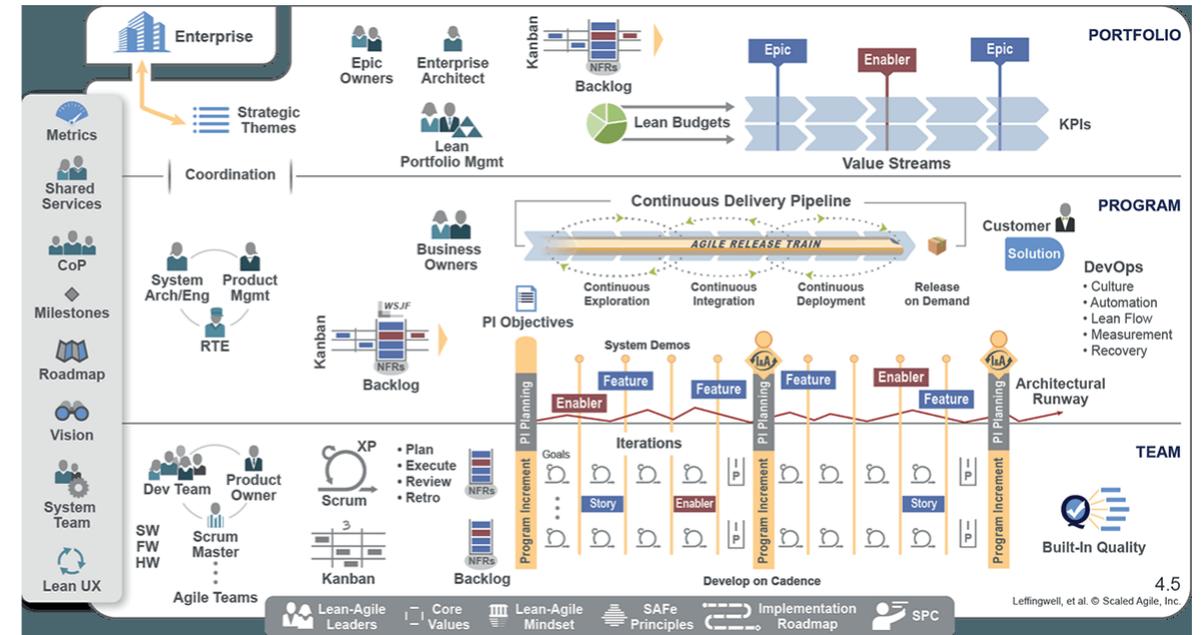
Bereits das Festlegen der Produktebene ist für die ADAS-Entwicklung nicht einfach: Hier geht es nicht nur um das einzelne Fahrzeug, sondern um ganze Fahrzeugflotten sowie die Vernetzung zwischen Fahrzeugen und mit dem Connected Services Backend.



Die Grafik aus dem „Whitepaper #01 Entwicklung des autonomen Fahrens“ von NTT Data zeigt Szenarien zur Verbreitung des autonomen Fahrens und erlaubt eine Analyse der möglichen Entwicklung des Ökosystems rund um selbstfahrende

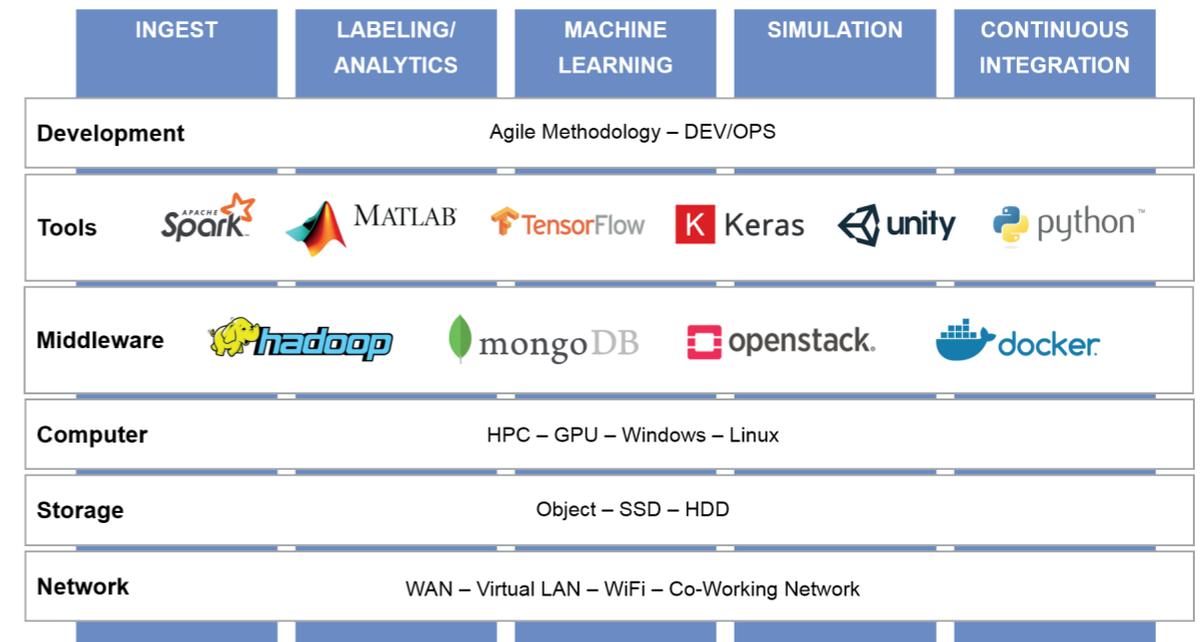
Autos

Quelle: NTT Data 2018



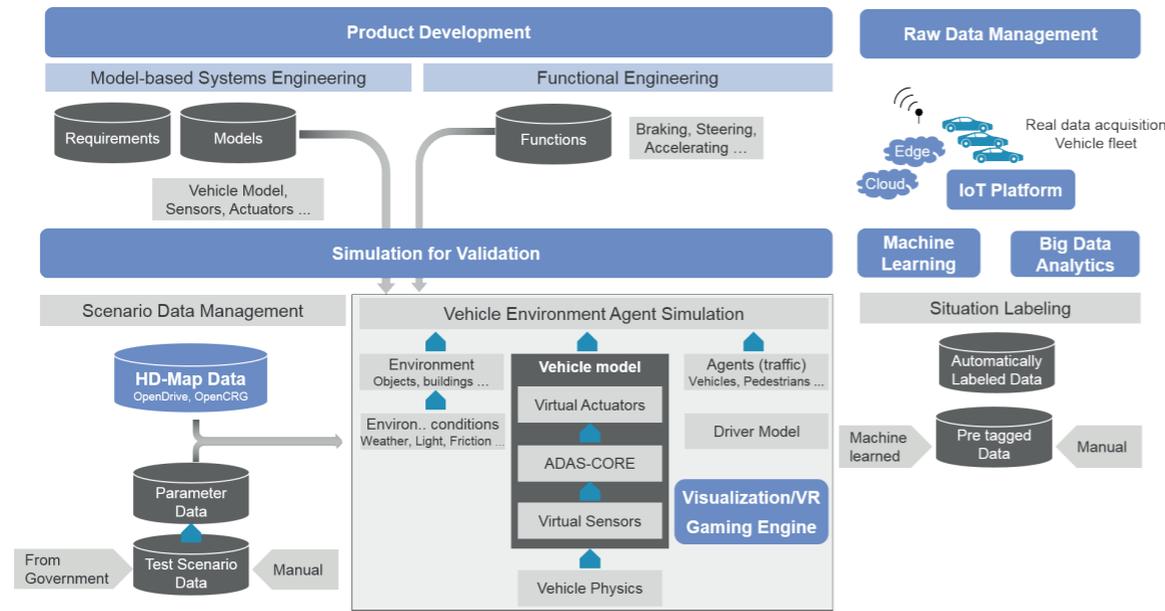
Scaled Agile Framework, siehe Text

Quelle: SAFe 2018



IT-Plattform zur Entwicklung und Absicherung automatisierter Fahrfunktionen

Quelle: Autor / NTT Data 2018



Schematische Darstellung der ADAS-Entwicklung

Quelle: Autor / NTT Data 2018

Entsprechend muss die Entwicklung der digitalen Services in den Entwicklungsprozess integriert werden, um deren Verfügbarkeit und Funktionalität bei der Markteinführung sicherzustellen. Das heißt konkret: Die Entwicklung von Services ist eine neue Disziplin im V-Modell.

**Agile Produktentwicklung.** In der Softwareentwicklung für Businessapplikationen werden mittlerweile überwiegend agile Methoden wie Scrum eingesetzt. Die Vorteile im Hinblick auf Flexibilität, Geschwindigkeit und auch Mitarbeitermotivation überwiegen mögliche Nachteile wie Kommunikationsaufwand und fehlende Projektplanung. Insbesondere die schnellen Zyklen von Spezifikation, Entwicklung und Test in den Sprints führen zu einem sichtbaren Projektfortschritt und bieten Gelegenheit, die Planung flexibel anzupassen. Entsprechende Überlegungen fließen auch in Verbesserungen des V-Modells ein. So gibt es zum Beispiel das W-Modell, in dem die Integration der digitalen Modelle aus den einzelnen Teildisziplinen schon während der Entwicklung erfolgt. Dadurch werden späte, teure Änderungen vermieden, die sonst in der Integrationsphase des konventionellen V-Modells zu erwarten sind.

In der Softwareentwicklung für sicherheitskritische Fahrfunktionen gibt es allerdings weitere Herausforderungen im Hinblick auf den Einsatz agiler Methoden,

nämlich die Sicherheitsnorm ISO 26 262 und die Skalierung der agilen Methoden auf Programmebene.

Die ISO 26 262 „Functional Safety“ ist eine Norm für sicherheitskritische E/E-Systeme und Softwarekomponenten in der Fahrzeugentwicklung. Sie nutzt ein Prozessmodell mit Phasen auf Basis des V-Modells, das zunächst im Widerspruch zu agilem Vorgehen steht. Mit einem Fokus auf die Arbeitsergebnisse wie Operational Situations, Hazards, Safety Goals und Safety Requirements sowie der Priorisierung dieser Themen im Backlog lassen sich auch agile Methoden für sicherheitskritische Themen nutzen.

Die optimale Größe von Arbeitsteams wird oft mit „7 +/- 2 Personen“ angegeben. Jedoch werden fürs autonome Fahren mit Level 3 bereits mehrere Hundert Millionen Zeilen Softwarecode benötigt, für Level 5 über eine Milliarde Zeilen Code! Es werden also Wege zur Skalierung agiler Methoden benötigt, mit denen Teams von mehreren Hundert Personen in einem Programm agil zusammenarbeiten können. Zweigängige Frameworks sind das Scaled Agile Framework (SAFe) und Large Scale Scrum (LeSS). Damit können mehrere agile Teams auf Teamebene effizient arbeiten und auf Programmebene effektiv synchronisiert werden.

**Model-Based Systems Engineering (MBSE).** Methoden wie Anforderungs- und Konfigurationsma-

agement können zu umfangreichen Dokumenten mit Hunderten von Komponenten-Lastenheften führen, die in System-Lastenheften referenziert werden. Auch mit Best Practices aus dem Requirements Engineering wird eine Komplexität erreicht, die von Menschen nicht mehr beherrschbar ist. Daher geht man zur Nutzung formaler Modelle anstelle von Fließtextspezifikationen über. Solche Modelle lassen sich zum Beispiel mit der Modellierungssprache SysML beschreiben. Die Vorteile liegen in der möglichen Automatisierung, zum Beispiel der Validierung von Modellen gegen vorgegebene Regeln, der automatischen Optimierung von Modellen sowie der Generierung von ausführbarem Code für Steuergeräte.

**Datengetriebener Entwicklungsprozess.** Für die Fusion der Sensordaten, das Erkennen der Umwelt und anderer Verkehrsteilnehmer und die Steuerung des autonomen Fahrzeugs wird KI eingesetzt, insbesondere Deep-Learning-Verfahren. Dabei bestimmen die Menge und Qualität der Trainingsdaten die Qualität der daraus resultierenden neuronalen Netze. Daten sind in diesem Kontext ähnlich wie Anforderungen zu sehen. Zur Absicherung und Zulassung der ADAS müssen Millionen von Testscenarien und virtuellen Testkilometern simuliert werden.

Autonome Fahrzeuge generieren circa 5 Terabyte pro Tag. Diese Datenmengen sind effizient von Testfahrzeugen herunterzuladen und mit einer Speicherarchitektur zu verwalten, die für dreistellige Petabyte-Volumina ausgelegt ist. Die Kosten solcher IT-Plattformen lassen sich durch gemeinsame Nutzung einer Hybrid-Cloud-Architektur optimieren.

Neben teuren Messdaten aus Realfahrten kommen auch synthetisch generierte Trainingsdaten zum Einsatz. Umfangreiche Simulationsumgebungen können Straßen, Umgebung, Wetter, andere Verkehrsteilnehmer und die eigene Fahrzeugphysik simulieren. Durch Variation dieser Teilmodelle lassen sich nicht nur umfangreiche Trainingsdaten generieren. Es können auch die besonders interessanten Grenzfälle untersucht werden, die im realen Verkehr nur selten vorkommen. Ein weiterer Vorteil synthetischer Trainingsdaten ist: Der aufwendige Labeling-Prozess, das heißt, die Klassifizierung von Objekten in realen Sensordaten als Referenz für das Machine Learning, entfällt.

Nicht nur die IT-Infrastruktur, sondern auch konzeptionelle Probleme im Debugging von neuronalen Netzen sind eine Herausforderung in diesem Umfeld. Nachdem die Ergebnisse nicht mehr hauptsächlich auf beweisbaren Algorithmen, sondern auf umfangreichen, heterogenen Daten basieren, müssen neue Wege zur Absicherung gefunden werden. Gemäß ISO 26 262 werden derzeit Sicherheitskonzepte für Technik, Hardware und Software erstellt. Nach der aktuellen Diskussion ist die Entwicklung eines zusätzlichen Sicherheitskonzepts für Machine Learning erforderlich. Dieses kann nicht mehr auf einfachen Fehler-Ursache-Wirkungsketten beruhen, sondern muss mit Hypothesen und Wahrscheinlichkeiten in neuronalen Netzen arbeiten.

**Ausblick**

Vor einer breiten Markteinführung müssen nicht nur technische Hürden genommen, sondern auch weitere Fragen beantwortet werden, zum Beispiel von Gesetzgebern, Zulassungsbehörden und Versicherungsgesellschaften. Zentrale Voraussetzung für die gesellschaftliche Akzeptanz des autonomen Fahrens sind sicherere Systeme. Das bedeutet neben funktionaler Sicherheit der Fahrerassistenzsysteme auch Datensicherheit, sichere Softwareupdates und Cyber Security zum Schutz des Fahrzeugs gegen Angriffe von außen. Daher muss dieses Thema im Sinne von „Security by Design“ bei allen Verbesserungen des Produktentwicklungsprozesses durchgängig im Vordergrund stehen. Ob das bis 2020 zu schaffen ist?



Elon Musk (unser Bild) geht davon aus, dass komplett eigenständig fahrende Fahrzeuge ab 2020 serienreif sein sollten