



Whitepaper

Die nächste Generation an E/E-Architekturen

Die Automobilindustrie steht vor einer der größten Transformationen ihrer Geschichte. Neue Geschäftsmodelle versprechen zusätzliche Erlöse über die Lebensdauer eines Fahrzeugs. Endkunden erwarten zunehmend, dass das Auto in die Lebenswelten Wohnen, Freizeit und Arbeiten integriert wird. Hieraus ergeben sich neue Chancen, sich im Wettbewerb zu differenzieren. Diese sind jedoch insbesondere für die etablierten Hersteller mit großen Herausforderungen verbunden.

Vorwort

Heutige E/E-Fahrzeugarchitekturen sind limitiert in Bezug auf schnelle Updatebarkeit, Individualisierbarkeit und Konnektivität. Und auch die zugrundeliegenden Entwicklungsmethoden und -werkzeuge stellen ein Hindernis in der Erreichung der gewünschten hohen Entwicklungsgeschwindigkeit mittels neuer Softwarezentrischer Technologieplattformen dar. Um bei der Transformation zum Software-defined Vehicle (SdV) erfolgreich zu sein, benötigt es vielseitige Strategien basierend auf fundierten Analysen der notwendigen Veränderungen in den verschiedenen Technologiedomänen.

In diesem Whitepaper beleuchten wir das SdV aus Sicht der E/E-Architektur. Wir beschreiben die Ausprägung der SdV-Treiber in und zwischen den einzelnen funktionalen Fahrzeugdomänen und leiten daraus Technologiebedarfe und archetypische Referenzarchitekturen ab.

Die Implementierung des SdV ist eng verzahnt mit neuen fahrzeugzentrischen und zonalisierten E/E-Architekturen, da diese den technischen Unterbau für eine stark vereinfachte Funktionsentwicklung auf Basis performanter Zentralrechner und strukturvereinfachender Zonensteuergeräte bieten. Für eine kosteneffiziente Umsetzung ist es essenziell, den technischen Umbau pragmatisch unter Berücksichtigung der Bestandarchitekturen zu betreiben. Es gibt dabei keine One-size-fits-all-Lösung, weshalb wir mögliche Fusions-, Migrations- und Konvergenzscenarien, die auch schrittweise umgesetzt werden können, einbeziehen.

Als weltweit größter Tier 1 mit langjähriger Erfahrung in allen Fahrzeugdomänen liefern wir einen umfassenden Blick auf das disruptive Thema neuer Fahrzeugarchitekturen. Damit unterstützen wir unsere Kunden auf ihrem Weg in die nächste Fahrzeuggeneration.

Inhaltsverzeichnis

1

Einleitung 04

2

Das Software-defined Vehicle verändert die Fahrzeugdomänen, löst sie aber nicht auf 06

2.1 Verbindung von Informations-, Unterhaltungs- und Komfortfunktionen schafft Mehrwert 07

2.2 Fahrdynamik und -komfort beruhen auf den in der Fahrzeugproduktion verbauten Aktuatoren 09

2.3 Datengetriebene Entwicklung mit neuen Geschäftsoportunitäten prägen assistiertes/ automatisiertes Fahren 11

3

Die primäre Herausforderung in der SW-Architektur lautet Komplexitätsmanagement 13

4

Die Zonalisierung der E/E-Architektur pragmatisch umsetzen 15

5

Die nächsten Zentralisierungsschritte kommen in verschiedenen Varianten 17

6

Fazit 19

Impressum 20

1

Einleitung

Der Ruf nach neuen E/E-Fahrzeugarchitekturen hat in den vergangenen Jahren massiv zugenommen. Erste Fahrzeughersteller stellen bereits ihre Fahrzeuge auf zentralisierte und zonale Architekturen um. Jedoch gibt es weiterhin Kritik an den neuen Designs: zu teuer, schwer skalierbar und zu wenig auf einen schrittweisen, risikoarmen Migrationspfad für etablierte OEMs ausgerichtet heißt es in E/E-Architekturkreisen. Dabei helfen die neuen Architekturen, um neues Potential für Fahrzeughersteller zu heben. Ausgerichtet auf Vereinfachung, Updatebarkeit und softwarebasierte Erweiterungen bieten sie einen relevanten Wettbewerbsvorteil. Auch wenn es momentan teilweise zu Verzögerung kommt: Das Rennen um fundamental verbesserte Fahrzeugarchitekturen ist im vollen Gange.

Die neuen Fahrzeugarchitekturen sind besser an die sich verändernden Endkundenerwartungen angepasst, denn Motorleistung, Exterieur und Sicherheit sind schon lange nicht mehr ihre einzigen Entscheidungskriterien bei der Fahrzeugwahl.

Insbesondere jüngere Menschen kaufen ein Fahrzeug, weil es ihnen Unterhaltung, Komfort und eine enge Einbindung in die Online-Welt verspricht.

Sie suchen Produkte, die ihnen eine enge Vernetzung zwischen ihren Lebenswelten Transport, Wohnen und Produktivität bieten. Das Fahrzeug wird zur technologischen Bühne, auf der personalisierbare User Experience (UX) und kontinuierliche Weiterentwicklung im Vordergrund stehen. Dies ist eine Chance, insbesondere für etablierte Automobilhersteller, um neue Domänen und Erlösquellen zu erschließen.

Neue E/E-Architekturen stehen jedoch im Spannungsfeld zwischen erwarteter Kosteneffizienz in der Herstellung und den in diesen Fahrzeugen ermöglichten optionalen und nachträglichen Erweiterungen. Bisher einmalig beim Kauf konfigurierbare „Pakete“ werden um (auch temporär) nachbuchbare Fahrzeugfeatures ergänzt. Neue Features werden sogar erst nach Beginn der Fahrzeugproduktion entwickelt und anschließend zur Verfügung gestellt.

Die Automobilindustrie beantwortet diese Herausforderungen mit dem Software-defined Vehicle, also durch eine Verschiebung des technologischen Lösungsraums von Hardware zu Software. Diese Refokussierung ermöglicht die gewünschte Flexibilität, bedarf jedoch nachhaltiger Veränderungen in der Gesamtfahrzeugarchitektur. Neben dem Ersatz vieler kleiner, funktional orientierter Steuergeräte durch wenige mächtige Fahrzeugrechner ist das „Vehicle OS“, ein auf Variantenreduktion in der Hardware- und Softwareentwicklung zielendes Konstrukt, Hauptthema der nächsten Generation von E/E-Architekturen. Statische Funktionen und zukaufbare Features werden in diesen neuen Architekturen um dynamische und verteilte Services ergänzt, zum Beispiel zur Nutzung von Clouddaten zur Optimierung der Fahrstrategie. Durch die starke Zunahme von Querabhängigkeiten verschwimmen die klassischen Domänengrenzen, jedoch mehr auf funktionaler als auf technischer Ebene. Dabei ist klar: Bestandsarchitekturen, Märkte und Fahrzeugsegment-Schwerpunkte bedingen eine Varianz im Lösungsraum.

Viele Funktionen und Ziele des SdV lassen sich jedoch bereits durch punktuelle Veränderungen umsetzen. So bleiben die Optimierungen heutiger Architekturen in ihrer zukünftigen Evolution erhalten.

In diesem Whitepaper beleuchten wir das SdV aus dem Blickwinkel der E/E-Architektur. Wir identifizieren den Einfluss von SdV auf die unterschiedlichen Domänen und zeigen auf, welche Aspekte bei der Weiterentwicklung bestehender E/E-Architekturen jeweils in Betracht gezogen werden müssen. Mit unserer Erfahrung als Anbieter von Gesamtlösungen betrachten wir die Weiterentwicklung und Ergänzung von Architekturelementen, wie bestehende Architekturelemente wie Steuergeräte, Sensoren, Aktuatoren und der Kabelbaum weiterentwickelt, ersetzt oder um neue Lösungen ergänzt werden müssen.

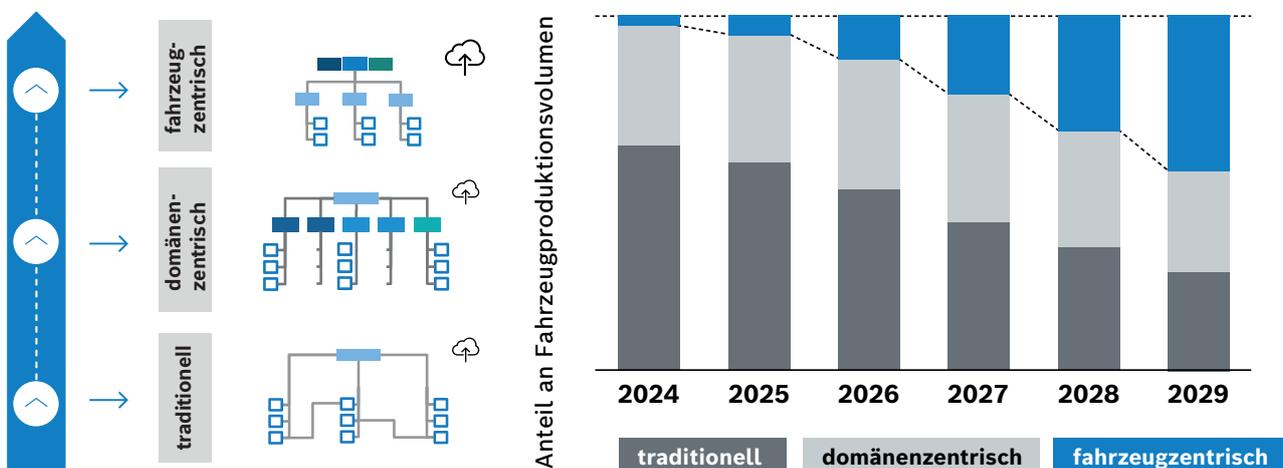


Abb. 1 Automobilhersteller bringen neue fahrzeugzentrische Architekturen in Serie, die zunehmend bisherige Architekturstrukturen ersetzen.

2

Das Software-defined Vehicle verändert die Fahrzeugdomänen, löst sie aber nicht auf

Mit dem Software-defined Vehicle sind große Erwartungen verbunden. Dabei fragt man sich: subskriptions- und servicebasierte Funktionalitäten wie Abomodelle für Verkehrsdaten gibt es auch schon heute: Warum muss dann das technische Rückgrat des SdV verändert werden?

Der Neuheitsgrad im SdV entsteht durch die starke Dynamik. Wo Features in der Vergangenheit nur einmalig definiert wurden, entwickeln sie sich im SdV über die gesamte Lebensdauer und Domänengrenzen hinweg. Das Fahrzeug wird zur lebenden Plattform, für die das Produktionsdatum nur noch sekundärer Indikator seiner Fähigkeiten ist. Um dies zu ermöglichen, bedarf es verschiedener Vorhalte.

In einem ersten Schritt benötigt das SdV Hardware mit entsprechender Performance-reserve oder einer Möglichkeit zum leichten Austausch über Lebensdauer. Das ist die offensichtlichste Veränderung. Das SdV lässt sich schwerlich in einer verteilten Bestandsarchitektur umsetzen.

Zweitens bedarf es einer schnellen Anbindung an eine Cloudplattform, über die Funktionen verwaltet werden und mit dem Fahrzeug kontinuierlich interagiert werden kann, z.B. für Telemetrie, Daten oder Services. Cloudplattform und Fahrzeug werden so zu einem eng verwobenen Ökosystem. Schließlich werden Updates zu einem fundamentalen Element des SdVs. Diese werden für verschiedene Zwecke eingesetzt:

01 Freshness: die Aktualisierung oder Erweiterung bestehender Funktionen, für ein verbessertes Nutzererlebnis

02 Bugfixing: die Beseitigung von nachträglich entdeckten Fehlern, bevor sie den einzelnen Kunden betreffen

03 Offerings: die Einbringung neuer Funktionen, um kurz- oder längerfristig das Fahrzeug weiter aufzuwerten

Die Fähigkeit zur Schaffung immer neuer Angebote setzt neue Systemschnitte und eine „Softwareifizierung“ voraus. Die Lösungen hierfür unterscheiden sich jedoch zwischen den einzelnen Bestandteilen des Fahrzeugs. Auch wenn in Zukunft alle Domänen gleichermaßen softwarezentrischer ausgelegt sein werden, so ist doch zu erwarten, dass sich das SdV zwischen den Domänen unterscheiden wird, bedingt durch individuelle Randbedingungen und Optimierungskriterien. Um die technischen Bedarfe des SdV besser zu verstehen, haben wir die neuen Treiber und Erwartungen für die jeweiligen Domänen gesammelt. Hierdurch entsteht eine Grundlage, um Architektur-entscheidungen und Variantenpunkte zu beleuchten. Wir beginnen im Folgenden mit den Infotainment- und Body/Comfort-Domänen, betrachten dann Funktionen zur Fahrzeugbewegungsregelung und schließen mit einer Analyse der Fahrerassistenz- und automatisierten Fahrfunktionen.

2.1

Verbindung von Informations-, Unterhaltungs- und Komfortfunktionen schafft Mehrwert

Neuartige SdV-Features erhalten ihren Automotive-spezifischen Charakter, weil sie ein domänenübergreifendes Kundenerlebnis erzeugen. Solche domänenübergreifenden Funktionen existieren bereits heute. Im Software-defined Vehicle werden die Übergänge zwischen Domänen jedoch nicht mehr featurespezifisch geschaffen, sondern generisch zur Verwendung in einer Vielzahl zukünftiger Applikationen. Wir gehen davon aus, dass die Infotainment- und Body-Domänen anfangs am stärksten hiervon betroffen sind, da sie gemeinsam die gewünschten physischen und digitalen Interaktionen mit Insassen und Fahrzeugfunktionen bereitstellen. Darüber hinaus ist die Infotainmentdomäne häufig auch der funktionale Endpunkt für die Cloudanbindung des Fahrzeugs und erweitert es um smartphonebasierte und Remote-services.

Um schnell neue, wertschaffende und für den Endkunden direkt erlebbare Services implementieren zu können, stellen die neuen Fahrzeugplattformen dafür unkompliziert den Zugang zu Sensorinformationen, Aktuatoren und Fahrzeuginformationen bereit. Dabei dürfen aber die Cybersecurity- und funktionalen Sicherheitsziele nicht aufs Spiel gesetzt werden. Insbesondere solche Fahrzeugfunktionen, die Licht, Türen und Sitze kontrollieren, können durch fehlerhafte Implementierung oder Fehlfunktion zu einem Sicherheitsrisiko werden. Weiterhin dürfen personenbezogene Daten nicht in falsche Hände gelangen.

Daher ist es absehbar, dass physische Funktionen und ihre hardwarenahe Steuerung technisch vom direkten Zugriff durch Infotainment-Apps separiert bleiben und stattdessen über logische Applikationsschnittstellen, sogenannte Vehicle APIs oder Hardwareabstraktionsschichten, zugegriffen wird. Aus praktischen Gründen, wie zum Beispiel Steckergrößen, wird es weiterhin sinnvoll sein, Bodyfunktionen zumindest teilweise auf dedizierten Steuergeräten (möglicherweise auch als Zusatzfunktion eines Zonencontrollers) zu belassen. Alternativ ist eine Verlagerung in eine dedizierte Ausführungsumgebung auf einem zentralen Controller sinnvoll. Hierdurch wird die notwendige technische Separierung der Funktionen vom Sicherheitsstandpunkt aus erreicht.

Jedoch beobachten wir verschiedene Allokationsstrategien der Fahrzeughersteller für die Steuerungslogik von Sensoren und Aktuatoren. So werden Logiken zentralisiert, lokal auf Bestandssteuergeräten oder auch verteilt zwischen Zonen-ECUs implementiert. Dies erfolgt häufig individuell für jede Funktion, in einigen Fällen gemäß einer generischen Strategie. Aufgrund stringenter Vorgaben bezüglich der Selbstdiagnose, des Aufstarts- und Fallbackverhaltens verschwinden sensor-/aktuator-spezifische Implementierungen jedoch nie vollständig aus den unteren Systemebenen.

Wie eingangs genannt, sind generische Applikationsschnittstellen Grundpfeiler einer vereinfachten domänenübergreifenden Funktionsimplementierung. SdV-Plattformen optimieren sich weiter, indem sie funktionale Sicherheitsziele durch generische, softwarebasierte Schutzmechanismen realisieren, z. B. sodass kritische Aktuatoren für die SdV-Funktionen im Fahrbetrieb gesperrt werden können. Hierdurch kann die Prozesslast für die SdV-bezogenen Implementierungen stark reduziert werden. Dieser Zentralansatz wird idealerweise mit einfachen, lokal im Aktuator realisierten Schutzmechanismen kombiniert.

Vehicle APIs sind eine Softwaremethodik die, mittels entsprechender Abstraktionen, eine nahezu freie Verschiebung technischer Endpunkte innerhalb der E/E-Architektur erlaubt. Für die unterliegenden Kommunikationsbusse gilt es allerdings, diese entstehende Dynamik zu unterstützen.

Daher werden Vehicle APIs mit solchen Transportprotokollen verbunden, welche einen dynamischen Verbindungsaufbau, Discoveryfunktionen und Quality-of-Service bieten. Beispiele hierfür sind SOME/IP und DDS über Ethernet, oft in Kombination mit zeitkritischen Netzwerkfähigkeiten (time-sensitive networking, TSN).

Ein weiterer Aspekt des Systems Engineerings dieser zukünftigen Fahrzeugarchitekturen betrifft das Varianten- und Versionsmanagement. Zusätzlich zur Versionierung der SW-Komponenten werden zukünftig auch Applikationsschnittstellen versioniert. Letzteres vereinfacht die Verwaltung von Abhängigkeiten. Dieses technische Kompatibilitätsmanagement erfolgt aber immer unter Beachtung der nebengeltenden Regularien, z. B. zur Freigabe der Gesamtfunktion.

⊕ Datentelemetrie wird essentieller Bestandteil des Entwicklungsprozesses.

⊕ Nutzung von Cloud-daten und Funktionsauslagerung in allen Fahrzeug-domänen

⊕ Abobasierte Fahrassistentenfunktionen mit vorinstallierter Hardware für Fahrzeuge im höheren Marktsegment

⊕ Kontinuierliche Leistungsverbesserung durch Shadow Mode und kontinuierliche Systemaktualisierungen

⊕ Nahtlose Integration cloudbetriebener und digitaler Services in die physische Fahrzeug-erfahrung über Infotainment hinaus

⊕ Zunehmender Fokus von Endkunden auf Komfort- und Unterhaltungsfunktionen

⊕ Neue batterieelektrische Plattformen als Chance und Innovationsbasis für signifikante Umbauten der E/E-Architektur

⊕ Höhere Energieeffizienz durch verbesserte datengetriebene Betriebsstrategien

Abb. 2 Fahrzeugzentrische Architekturen sind eine Antwort auf die Erwartungen des C.A.S.E*-Paradigmas. Sie bedingen eine Technologieverschiebung in Richtung Software und Technologieplattformen, die schnelle Entwicklungsiterationen unterstützen.

*(connected, autonomous, sharing, electrified)

2.2

Fahrdynamik und -komfort beruhen auf den in der Fahrzeugproduktion verbauten Aktuatoren

Das manuelle Fahren wird auch mit Verfügbarkeit zunehmender Umfänge an assistierter und automatisierter Fahrfunktionen immer noch ein zentrales Thema in Bezug auf das Nutzererlebnis bleiben, ist es doch Teil des Markenkerns vieler Automobilhersteller. Es ist zu erwarten, dass sich das in Software implementierte Fahrverhalten, insbesondere im Regelfahrbetrieb, für den normalen Endkunden aufgrund hoher Ausgangsreife nur unwesentlich über die Fahrzeuglebensdauer ändern wird. Softwareupdates und Umbauten für das SdV finden primär „unter der Haube“ statt, in Form von Wartungserleichterungen und Performanceverbesserungen durch Zentralisierung und Betrieb bzw. Verschiebung nicht-kritischer Zusatzfunktionen in die Cloud.

Während die Eigenschaften neuer SdV-Architekturen in dieser Domäne also nicht unmittelbar für den Endkunden erlebbar sein dürften, profitieren Automobilhersteller dennoch von ihrer Einführung.

Sie können neue zentrale Kontrollsysteme nutzen, um die Leistung der dynamischen Längs- und Querkoordination ihrer Fahrzeuge in Bezug auf Sicherheit, Komfort und Effizienz weiter zu verbessern. Mit einer weitgehenden Zentralisierung von Lenkungs-, Brems-, Antrieb- und Dämpfungsfunktionen, wird eine gemeinsame Entwicklung in einer gemeinsamen Entwicklungsumgebung ermöglicht.

Trotz dieses Trends zur Zentralisierung sehen wir jedoch noch keinen größeren Schritt zu einer Auflösung der engen Verbindung zwischen enger Steuerung bzw. Regelung und den Sensoren und Aktuatoren der Domäne. Die vergleichsweise geringen Systemkostenvorteile, besonders hohe Sicherheitsanforderungen und hochgradig kosten- und leistungsoptimierte Einzelkomponenten stehen dem entgegen.

Für die oben genannten Subsysteme bringt SdV den Vorteil, dass sich, neben einer vereinfachten gemeinsamen Entwicklung, Updates ohne Werkstattbesuch ausrollen lassen. Über diesen Pfad können auch Verbesserungen der unterstützenden Funktionen für kritische Fahrsituationen, z. B. für Traktionskontrolle und Fahrdynamik, unterjährig eingebracht werden. Aus Gründen der Sicherheitsrelevanz und stringenterer Vorgaben bezüglich der Fahrzeug-homologation ist aber nicht mit den vergleichsweise hohen Updateraten anderer Domänen und einer Variantenvielfalt für die Funktionen an sich zu rechnen.

Ein Online-Kanal zu einem Automotive-Backend, tendenziell umgesetzt in einer von den direkten Fahrkoordinationsfunktionen separierten Ausführungsumgebung, bringt aber die Möglichkeit, das Fahrerlebnis durch eine cloudbasierte Funktionserweiterung noch sicherer zu machen wie z. B. durch eine vorausschauende Warnung bei schlechten Straßenbedingungen durch Schnee und Eis.

Da die Fahrfunktion aber auch ohne Remoteverbindung verfügbar bleiben muss, handelt es sich hier vorrangig um nicht-essenzielle Erweiterungen der Kernfunktionalität. Analog verhält es sich bei Funktionen zur Optimierung der Betriebsstrategie, die (je nach Ausbaustufe) in lokalen Umgebungen umgesetzt werden, die für dynamische Lasten optimiert wurden, beziehungsweise in Cloud-Umgebungen realisiert werden. Bezüglich ihrer Homologationsrelevanz kann es hier aber Einschränkungen in der freien Implementier- und Updatebarkeit geben.

Für den Fahrzeughersteller ergeben sich Chancen für neue indirekte Monetarisierungsmodelle: Über standardisierte Schnittstellen gesammelte Daten können für wertschöpfende Services wie prädiktive Wartungen oder zur Erstellung von Batterie-zertifikaten genutzt werden. Auch lassen sich Features nachträglich aktivieren.

2.3

Datengetriebene Entwicklung mit neuen Geschäftsoportunitäten prägen assistiertes/automatisiertes Fahren

Die Entwicklung assistierter und automatisierter Fahrfunktionen ist ein großer Nutznießer der durchgängigen Konnektivität des SdV. Mittels eines konstanten Datenkanals für Szenen- und Telemetrie-daten können assistierende/automatisierte Fahrsysteme (ADAS) kontinuierlich weiterentwickelt werden. Diese Verbesserungen werden allen Kunden über iterative Updates zugänglich gemacht.

SdV-Technologieelemente wie die fahrzeug-internen Updatemechanismen und die Backend-Anbindung sind dazu prinzipiell notwendige Erweiterungen zur nachhaltigen Verbesserung der Produktqualität, können aber auch heute schon in bestehenden domänenzentrischen Architekturen umgesetzt werden. Im Sinne des Nutzererlebnisses ist hier also in erster Näherung kein direkter Bedarf für eine Veränderung der ADAS-Domäne erkennbar. Ist damit ADAS eine „nicht-SdV“-Domäne? Nein.

Durch das SdV ändert sich das Geschäftsmodell für ADAS, welches notwendige Veränderungen in der unterliegenden Architektur bedingt. Betrachtet man Bestandsarchitekturen, so basieren diese auf einem Kostenoptimierungsansatz, der auf optimale Auswahl des verbauten Sensorsatzes, der Rechenleistung und der Speicher zielt. Obwohl besonders kosteneffizient, verhindert dies Upgrades nach Produktion des Fahrzeugs.

Architekturen, die auf generelle SdV-Fähigkeit im ADAS-Bereich zielen, gehen hier einen gänzlich anderen Weg. Die gege-

benenfalls höheren Fahrzeugproduktionskosten werden den Erlösen durch Abo-Modelle oder Pay-per-Use-Optionen entgegengesetzt. Weiterhin werden Kostenersparnisse durch Reduktion der Varianten berücksichtigt. Dies erfolgt trotz einer potenziellen Überdimensionierung im Einzelfall.

Die SdV-Cloudanbindung dient hier zur Freischaltungsverwaltung und Abrechnung. Es ist offensichtlich, dass sich dies nur rechnet, wenn der Business Case aus Fahrzeugproduktionskosten, zusätzlichen Erlösen über Lebensdauer und der möglichen Variantenreduktion positiv ausfällt.

Nach momentanen Beobachtungen setzen bisher insbesondere Automobilhersteller mit einem Fokus auf das obere und Premiumfahrzeugsegment auf diesen Ansatz. Der Übergang von assistierenden ($\leq L2+$) zu automatisierten ($\geq L3$) Fahrfunktionen erhöht dabei signifikant den Sensorik-, Performance- und Redundanzbedarf. Damit sich das Zukaufkonzept auch jenseits des Premium-Segments rechnet, ist eine Variantenbildung zwischen Fahrzeugen mit und ohne zukaufbare/r $\geq L3$ Automatisierungsfähigkeit ein valides Unterscheidungskriterium in der Produktion. In neuen Fahrzeugarchitekturen lässt sich dies vorteilhaft durch rein additive Erweiterungen mit zusätzlichen Fahrzeug-ECUs und in Form zusätzlicher Sensoren und redundanter Stromversorgung ohne Bruch zu einer originären $\leq L2+$ -Architektur realisieren.

Durch die engere Cloudanbindung können im SdV weitere Datenquellen einbezogen werden, um die funktionale Performance der Assistenzfunktion (z. B. durch eine vorausschauende Planung mittels Onlinekarten mit erfassten Straßenbedingungen wie Glatteis) weiter zu verbessern. Bis zuverlässigere Drahtlostechniken flächendeckend verfügbar werden, gilt jedoch: Der robuste und universell verfügbare Kern der lokalen ADAS-Funktionsplattform wird hier um volatilere SdV-artige Elemente erweitert, aber nicht durch Cloudlösungen ersetzt.

Aus Gesamtarchitektursicht wird die Zentralisierung der ADAS-Funktionen für die Umsetzung des SdV weiter fokussiert. Dies ist insbesondere dann nötig, wenn eine Skalierung auf Basis von Software gewünscht ist.

ADAS-spezifische Sensoriken mit hohen Bandbreitenanforderungen (Kameras, Lidare und Rohdaten-Radare) bleiben dabei nach bisherigem Stand vorerst stern-topologisch über technologie-domänen-spezifische Breitbandschnittstellen (z.B. LVDS für Kameras) oder per Multi-Gbit-Ethernet angeschlossen. Eine Verschiebung in Richtung einer Zonalisierung erfolgt für niedrig-volumige Datenschnittstellen. Für Ultraschallsensoren mit niedrigeren Bandbreiten kann eine zonale Kopplung darüber hinaus geometrisch vorteilhaft sein.

3

Die primäre Herausforderung in der SW-Architektur lautet Komplexitätsmanagement

Im vorherigen Kapitel haben wir abgeleitet, welche Bedarfe die einzelnen funktionalen Domänen an SdV-E/E-Architekturen haben. Zentralisierung und Konsolidierung sind hier die Haupttreiber. Nun begeben wir uns auf einen Exkurs in die Softwaredomäne und betrachten technische Separationsbedarfe, da sie ein übergreifendes Thema bei der Bestimmung des möglichen Konsolidierungsgrades in den E/E-Architekturen sind.

Im SdV wird die Software zum Dreh- und Angelpunkt der technologischen Weiterentwicklung des Fahrzeugs. Damit der steigende Softwareanteil beherrschbar bleibt, bedarf es komplexitätsreduzierender und -kontrollierender Maßnahmen. Sowohl die SW-Plattformen des SdVs als auch das verwendete Tooling müssen entsprechend passend gewählt werden. Aber auch die Eignung der Hardware als Trägerplattformen muss geprüft werden. Hierzu ist eine ausreichende, hardwarebasierte Unterstützung von Separationsmaßnahmen vorzusehen. Wo vormals die Komplexität durch geschickte Platzierung von Funktionen auf unterschiedliche Steuergeräte reduziert wurde, sind es nun die Steuergeräteplattformen selbst, die die Basis einer Separationsargumentation liefern.

Mit dem Übergang zum SdV werden technische Schutzmaßnahmen wie Hypervisor oder Container zu Hauptlösungen. Für mikroprozessorbasierte Plattformen haben hier Lösungen aus anderen Industrien, zum Beispiel aus dem Datenzentren-Bereich, bereits ihren Weg und ihre Anpassungen für den Automobilbereich gefunden. So wird es möglich, Funktionen verschiedener Domänen sogar auf einem System-on-Chip (SoC) zusammenzulegen. Die im Automobilbereich eingesetzten Betriebssysteme beinhalten viele Funktionen, die dabei helfen eine vorteilhafte Trennung zu realisieren. Dies dient dem Wunsch, Software möglichst autark und regressionsfrei in kleinen Teams zu entwickeln, zu verifizieren und zu validieren. Die Evolution auto-mobiler Mikroprozessorplattformen ist aber noch nicht abgeschlossen, sondern wird weiter in Konsortien wie SOAFEE vorange-trieben. Nichtsdestotrotz ist im Bereich der Mikroprozessoren bereits ein hoher Reifegrad erreicht.

Betrachtet man Mikrocontrollerplattformen auf der anderen Seite, so fällt die meist unzureichende Unterstützung für die modulare Bereitstellung verteilt entwickelter Funktionen auf. Zum Teil ist dies in hardwarebedingten Beschränkungen begründet, zum Teil jedoch auch in den zugrundeliegenden Softwareframeworks, welche meist auf singuläre Integrationspunkte und monolithische Infeldbringung zielen. Die erreichbare Granularität endet zumeist auf der Ebene einzelner virtuallisierter Partitionen mit eingeschränkter Unabhängigkeit und Hardwareagnostik. Dies stellt eine Barriere bezüglich der agilen Entwicklung dar, und führt zur generellen Einschätzung, dass SdV-Funktionen eher auf Mikroprozessorplattformen entwickelt werden.

Insbesondere für die sicherheitsrelevanten und intern eng funktional gekoppelten Fahr- und Assistenzdomänen gilt allerdings: Trotz entkoppelter Entwicklung muss die Software als Funktionsumsetzung weiterhin im Ganzen bewältigt werden, um z. B. den Anforderungen der UN ECE R156 bezüglich der Freigabemechanismen für Software-

Updates gerecht zu werden. Technische Systemschnitte z. B. im Hinblick auf homologationsrelevante Anteile sind dabei weit aus relevanter als die Fähigkeit, Softwarekomponenten individuell installieren oder aktualisieren zu können. Dies bedingt vorteilhafte E/E-Architekturen, die zum Beispiel homologationsrelevante Softwareteile strikt trennen können.

4

Die Zonalisierung der E/E-Architektur pragmatisch umsetzen

Wie nun offensichtlich wird, ist eine Zentralisierung von Funktionen essenziell für das SdV. Diese wird jedoch durch Beschränkungen der Hardware und der verwendeten Softwareframeworks teilweise konterkariert. Um unser Architekturbild abzuschließen, schauen wir uns nun die unterliegenden Schichten der E/E-Architektur an. Fokus liegt auf den Zonen als Mittel zur funktionalen Entkopplung.

Die Vereinfachung des Kabelbaums durch Segmentierung und De-/Multiplexing wird häufig als Hauptargument für eine geometrische Fahrzeugpartitionierung mittels Zonen-ECUs angeführt. Aus Sicht des Kommunikationsnetzes bündeln sie Kommunikationskanäle niedrigerer Geschwindigkeit. Die Leistungsverteilung im Fahrzeug profitiert von einem feingranularen Sicherungs- und Steuerungskonzept, z. B. unter Verwendung elektronischer Sicherungen. Dies vereinfacht die Unterverteilung, welche vorteilhaft in Kombination mit einem zentralen Leistungsmanagement umgesetzt wird.

Sowohl für die Produktionskosten als auch für die Automatisierung der Kabelbaumfertigung und -verlegung kann eine Zonalisierung Vorteile bringen, die jedoch je nach Ausstattungsgrad des Fahrzeuges variieren kann. Insbesondere im kostenoptimierten niedrigen Fahrzeugsegment mit signifikant geringer Vernetzungsdichte amortisieren sich zusätzliche ECUs allein zur Kabelbaumvereinfachung häufig nicht.

Zonen-ECUs bringen weitere Kostenvorteile, wenn sie gleichzeitig zu einer merklichen Reduktion der vorhandenen Steuergerätezahl beitragen. Dies ist insbesondere im Body-Bereich sichtbar, wo viele Kleinststeuergeräte konsolidiert werden können.

Wie bereits im Titel dieses Kapitels beschrieben, wird die Aufgabe der Zonen hier aber pragmatisch entschieden – vom „verlängerten Arm“ der Zentralsteuergeräte bis zum weitestgehend unabhängigen geometrisch orientierten Body-Domaincontroller. Serviceorientierte Architekturen erlauben einen größeren Freiheitsgrad in der Allokationsentscheidung. Sie nutzen Zonencontroller auch als Infrastrukturelement für die oben beschriebenen abstrahierenden Vehicle APIs.

Die Skalierung der Zonen-ECUs ist durch Pinzahl und Entwärmungsgrenzen (insbesondere für die Leistungselektronik) begrenzt, weshalb der Verwendung sensibler Bauteile wie hochperformanter SoCs oder großer Speicher enge Grenzen gesetzt sind. Erst durch eine umfassendere Kühlungslösung mit entsprechenden Kostennachteilen gäbe es Möglichkeiten für performantere Zonen-ECUs, jedoch erscheint hier eine Zentralisierung in Richtung bereits aktiv gekühlter Zentralrechner generell günstiger.

Eine weitere Verschiebung von Funktionen, insbesondere eine Sensorvorverarbeitung für die ADAS-Domäne, bringt keine klaren Vorteile. Stattdessen ist eine weitere Zentralisierung oder sensorinterne Vorverarbeitung auch im Sinne der Domänenunabhängigkeit und Technologiespezifität sinnvoller. So ist z. B. die Verarbeitungsalgorithmik von Radaren stark sensorspezifisch und hochgradig auf die lokalen Hardwarebeschleuniger optimiert. Die Verlagerung in eine Zonen-ECU bedarf gemeinsamer Komponentenentwicklung und ist eine zusätzliche Bürde für das Zonen-Design. Auch funktional ist eine zonenbasierte Vorverarbeitung nur begrenzt möglich, wie am Beispiel einer Kamerarundumsicht mit überlappenden (und zu fusionierenden) Bildbereichen schnell offensichtlich wird. Durch aktuell entstehende Standards und passende, kostengünstige Halbleiterbausteine könnten Kameras dennoch zu einem gewissen Zeitpunkt an Zonen angeschlossen werden. Zonen wären hier jedoch nur ein Multiplexer.

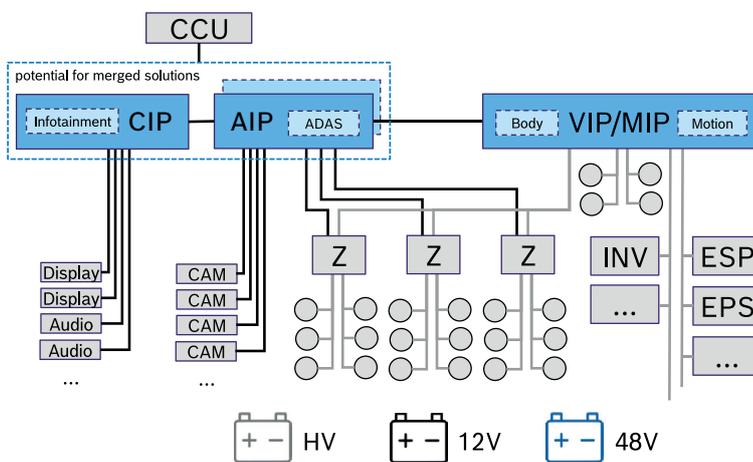
Abschließend bieten die auf Kommunikationsverteilung spezialisierten Zonen-ECUs das Potential, fahrzeuginterne Diagnose-Gatewayfunktionen zu übernehmen. Dies ist allerdings im Sinne des SdVs mit dem generellen Bedarf nach einer zentralen Datentelemetrie- und Update-Master-Funktion in Einklang zu bringen, die vorteilhaft auf der zentralen Compute-Ebene umgesetzt werden kann.

5

Die nächsten Zentralisierungsschritte kommen in verschiedenen Varianten

Die in den vorangegangenen Abschnitten dargelegte Sicht auf SdV-Architekturen aus Sicht der Domänen zeigt einen klaren Trend in Richtung Zentralisierung, mit konsolidierten Steuergeräten für ADAS, Infotainment, Fahr- und Body-Funktionen. Zwischen den „neuen“ Domänen und in Richtung der unterliegenden Zonen-ECUs werden stärker standardisierte Interfaces die domänenübergreifende Funktionsimplementierung erlauben. Die weiteren Integrationsschritte werden nach heutigem Erkenntnisstand auf Ebene der einzelnen Domänen und in unterschiedlichen technologischen Varianten erfolgen. So verschwinden die Domänensteuergeräte sukzessive aus der E/E-Architektur. Innerhalb der Steuergeräte werden die Domänengrenzen aus Gründen wie unterschiedlichen Anforderungen an Freigabeprozesse und geringen Vorteilen einer noch engeren Fusion aber vorerst erhalten bleiben.

Ein bereits heute durch einige Fahrzeughersteller umgesetzter Integrationsschritt ist der einer „Wohngemeinschaft“, bei der getrennte Compute-Einheiten in einem gemeinsamen Gehäuse untergebracht werden. Besonders die ADAS- und Infotainment-Domänen, die beide auf performante SoCs und hitzeempfindliche Speicherbausteine setzen, profitieren hier von einem gemeinsamen Kühlkonzept und gegebenenfalls weiterer geteilter Infrastruktur. Die Wohngemeinschaft kann dabei fix als Multi-PCB-Lösung ausgelegt, oder, mittels angepasster Modulkonzepte, freier konfektioniert sein. Diese erlauben auch eine günstigere Wartung und potenziellen Nachrüstung durch leistungstärkere HW-Versionen über die Fahrzeuglebensdauer.



- AIP = ADAS Integration Platform
- CAM = Camera Head
- CCU = Connectivity Control Unit
- CIP = Cockpit Integration Platform
- ECU = Electronic Control Unit
- ESP = Electronic Stability Program ECU
- EPS = Electric Power Steering ECU
- INV = Inverter ECU
- MIP = Motion Integration Platform
- VIP = Vehicle Integration Platform
- Z = Zone Controller ECU

Abb. 3 Erste fahrzeugzentrische Architekturen führen zentralisierte Fahrzeugcomputer und Zonen-ECUs ein. Wie aus diesem Beispiel ersichtlich, unterscheidet sich der Grad der domänenübergreifenden Integration und der Übergang zu einem zonalen Kommunikations-/Stromverteilungsnetz zwischen den Domänen.

Multi-SoC-Lösungen auf einem Einzel-PCB stellen einen starreren Integrationsgrad mit gleichzeitig erhöhten Synergien und Kostenvorteilen dar. Sie lassen sich auch für Skalierungsvarianten mit einem Modul-konzept verbinden.

Besonders im kostensensitiven Segment stellen Lösungen auf Basis von Fusion-SoCs eine valide Integrationslösung dar. Hier wird die Separation auf Ebene der Software, z. B. mit Hypervisoren und Containerframeworks, umgesetzt, immer gepaart mit starken hardwareunterstützten Schutz- und Quality-of-Service-Mechanismen. Dieser Integrationsgrad ist auch der komplexeste im Sinne der Darstellung funktionaler Interferenzfreiheit.

Welche Variante der Co-Integration sich anbietet, ist nicht allein technisch zu beantworten. Gerade Übernahmen aus Bestandsarchitekturen lassen aber einen gestuften Ansatz von der Wohngemeinschaft zum Fusion-SoC als Methode der Risikominimierung vorteilhaft erscheinen.

Neben der vor allem mikroprozessor-zentrischen Integrationsplattform für ADAS und Infotainment bietet sich ein vergleichbarer mikrocontroller-zentrischer Integrations-schritt für Motion, Gateway und Bodyfunktionen an. In diesen Plattformen dominieren weitaus stärkere Separationsmechanismen, die vorrangig in Hardware ausgelegt werden. Nur so können die höheren Determinismus-anforderungen und begleitende Themen wie eine Entkopplung homologations-relevanter Funktionen realisieren. Damit einher gehen eher statische Softwareframeworks, die jedoch auf etablierte Standards wie AUTOSAR Classic aufsetzen. Dafür bieten diese Plattformen niedrigere Ruhestromverbräuche, ein schnelles Aufstarten und leichtere Umsetzung von Funktionen bis zu maximalen Sicherheitsstufen.

6 Fazit

Die nächste Generation der E/E-Fahrzeugarchitekturen stellt Nutzererlebnis und Software in den Mittelpunkt. Getrieben durch eine veränderte Kundenwahrnehmung erlaubt sie Automobilherstellern die Erschließung neuer Business-Modelle auf Basis einer engen, datenzentrischen Verzahnung des Fahrzeugs mit cloudbasierten Services und durch die Öffnung von funktionalen Domänengrenzen in nutzerzentrischen Bereichen.

Abstraktion auf Hardware- wie Fahrzeugebene und eine größere Freiheit und Entkoppelung bei der Softwareentwicklung sind notwendige Voraussetzungen, um der immer schwieriger zu beherrschenden Komplexität des Fahrzeugsystems gerecht zu werden. Dabei helfen angepasste E/E-Architekturen, die geometrisch wie zentralisierend durch Konvergenz von Steuergeräten die Entwicklung von Fahrzeugfunktionen essenziell vereinfachen. Trotz einer zwischenzeitlichen Verlangsamung ist der Trend für neue Fahrzeugarchitekturen, der sich in Zonen-ECUs und Hochleistungsrechner manifestiert, ungebrochen.

Damit dieser Umbau im Rahmen der Kosten-, Zeit- und Qualitätsziele gelingen kann, bedarf es eines ganzheitlichen Ansatzes, der die spezifischen technischen und funktionalen Belange der Fahrzeug-

domänen berücksichtigt und sinnvolle Brüche in den oft dogmatisch gedachten fahrzeugzentrischen Architekturen vorsieht, wo Evolution, Kosten- oder Skalierungsgründe diese notwendig machen. Dies ist besonders relevant, um die E/E-Architekturen schrittweise aus Bestandsarchitekturen kommend und mit dem Ziel einer Abdeckung auch niedrigerer Marktsegmente weiterentwickeln zu können.

Der neue softwarezentrische Ansatz kann jedoch nur unzureichend im bisherigen Ökosystem skalieren. Neue Initiativen wie SOAFEE, COVESA und Eclipse.SDV ermöglichen den Zugang zu SW-Ressourcen und -Entwicklern in bisher unbekanntem Umfang. Die Erreichung der Qualitätsanforderungen und -werte der Automobilindustrie bedarf jedoch vieler weiterer gemeinsamer Anstrengungen, auch auf Hardware-Seite. Es bedarf neuer Partnerschaften, um gemeinsame Synergien zu erzeugen. Dabei sind den Besonderheiten der stark regulierten und auf Sicherheit ausgelegten Automobilindustrie in Verbindung mit hohem Kostendruck Rechnung zu tragen. Der Mangel an gemeinsamen Prozessen, Methoden und Tools für notwendige Zusammenarbeit in Projekten der Produktentwicklung stellt eine zu bewältigende Herausforderung dar.

Die Erschaffung eines Software-defined Vehicles wird gelingen, wenn die Anforderungen an Updatebarkeit, cloudbasierter Serviceintegration sowie Varianten- und Konfigurationsmanagement klar auf technische Vorzugslösungen und Funktionsansichten abgebildet sind. Bosch beschäftigt sich mit diesen Fragestellungen und ist als domänenübergreifender Systemanbieter mit Expertise im klassischen Automotive-Geschäft wie auch im Bereich des SdV unterwegs. Als weltweiter Player bringen wir unsere umfangreiche Expertise in Projekten und Studien ein, um tragfähige Architekturweiterentwicklungen wie in diesem Whitepaper skizziert zu ermöglichen.

Impressum

Warum Bosch?

Bosch ist mit seinen Produkten und Dienstleistungen über den gesamten Technologie-Stack hinweg ihr Partner für innovative Hardware-, Software- und Systemlösungen. Dabei stützt sich unser Expertennetzwerk auf jahrelange Erfahrung, um unsere Kunden bei der Realisierung von kosteneffizienten, skalierbaren und sicheren Lösungen über alle Automotive-Domänen hinweg bestmöglich zu unterstützen.

Autoren:

Dr. Thorsten Huck

Dr. Andreas Achtzehn

Andreas Deberling

Erik Goerres

Dr. Karsten Wehefritz

**Cross-Domain Computing Solutions
Engineering System Architecture**

Kontaktieren Sie uns jetzt

Sprechen Sie mit unserem Expertenteam. Wir freuen uns sie bei den nächsten Schritten zur Realisierung ihrer E/E-Architektur kompetent unterstützen zu können.

 thorsten.huck@de.bosch.com

 +49 173 3175436