



Whitepaper

Breaking Silos: Wie branchenweite Zusammenarbeit die Skalierung Software-definierter Nutzfahrzeuge ermöglicht

Vorwort



Die Erwartungen an Nutzfahrzeuge steigen rapide, angetrieben durch operativen Druck und die digitale Disruption. Flottenbetreiber müssen die Uptime und die Total Cost of Ownership (TCO) verbessern, ohne dabei an operativer Flexibilität einzubüßen. Gleichzeitig schaffen Fortschritte in den Bereichen Elektrifizierung, Automatisierung, Konnektivität und künstliche Intelligenz (KI) neue Möglichkeiten, diese Erwartungen zu erfüllen.

Viele dieser Entwicklungen nahmen zwar zunächst im Pkw-Sektor an Fahrt auf, doch die Nutzfahrzeugbranche ist mittlerweile über die reine Sondierungsphase hinaus – der Fokus verschiebt sich von der Vision hin zur skalierten Umsetzung.

Der Weg in die Zukunft ist jedoch nicht nur eine Übertragung von Pkw-Konzepten auf Lkw und Busse. Nutzfahrzeuge unterliegen völlig anderen wirtschaftlichen Realitäten, haben deutlich längere Lebenszyklen und werden letztlich an anderen Erfolgskriterien wie den Kosten pro Tonnenkilometer gemessen.

Nutzfahrzeuge erfordern daher einen spezifischen Ansatz für das Software-Defined Vehicle (SDV).

Dieses Whitepaper beschreibt die Perspektive von Bosch in dieser Diskussion. Es beleuchtet die technologischen, architektonischen und organisatorischen Überlegungen, die unserer Ansicht nach entscheidend sind, um die nächste Generation von Nutzfahrzeugen zu entwickeln.

Viel Freude beim Lesen!

Timo Wenninger

Executive Vice President
Sales & Operations Commercial Vehicle Group
Bosch Mobility

Inhaltsverzeichnis

Executive Summary	04
--------------------------	----

01

Die Bedeutung von SDVs für Nutzfahrzeuge	05
---	----

02

Evolution der Architektur: Drei Wegbereiter für skalierbare SDVs	10
---	----

03

SDV-Plattform: Standardisierung schafft Skalierbarkeit	14
---	----

04

Organisatorische Transformation: Struktur, Geschwindigkeit und Denkweise in Einklang bringen	17
---	----

05

Fazit und Ausblick	18
---------------------------	----

Executive Summary

Der Übergang zum Software-definierten Fahrzeug (Software-Defined Vehicle, SDV) wird zu einer strategischen Priorität für Nutzfahrzeughersteller. Software-definierte Fähigkeiten versprechen eine höhere Flottenverfügbarkeit, eine höhere Effizienz und niedrigere Total Cost of Ownership (TCO). Gleichzeitig bringen der wachsende Softwareanteil, die zunehmende Fahrzeugvielfalt und die funktionale Komplexität bisherige E/E-Architekturen an ihre Grenzen.

Dieses Whitepaper skizziert einen pragmatischen Weg zu skalierbaren, Software-definierten Nutzfahrzeugen. Es konzentriert sich auf drei architektonische Wegbereiter: **Funktionale Verteilung** (Functional Distribution), **Skalierbare Rechenleistung** (Scalable Compute), **Zonale Abstraktion** (Zonal Abstraction).

Gemeinsam helfen sie Fahrzeugherstellern (OEMs) dabei, bestehende Architekturen weiterzuentwickeln, Software über verschiedene Fahrzeugprogramme hinweg wiederzuverwenden und die erforderliche Rechenleistung bereitzustellen, ohne ein disruptives Redesign der Architektur zu erzwingen.

Technologie allein reicht jedoch nicht aus. Eine skalierbare SDV-Implementierung erfordert zudem modulare Software-Assets, eine klare Plattformverantwortlichkeit (Platform Ownership), eine effektive Zusammenarbeit im Ökosystem und neue Betriebsmodelle. Diese Elemente sind unerlässlich, um schnelle Software-Innovationen mit der Sicherheit, Zuverlässigkeit und Langlebigkeit zu verbinden, die in Nutzfahrzeugen gefordert sind.

Bosch trägt gemeinsam mit seinen Partnern aus dem gesamten Ökosystem zu dieser Transformation bei – mit komplementären Software-, Hardware- und Integrationskompetenzen. Dies hilft OEMs, die Implementierung von Software-definierten Nutzfahrzeugen zu beschleunigen und gleichzeitig Komplexität, Integrationsaufwand und Risiken im Lebenszyklus zu managen.

01

Die Bedeutung von SDVs für Nutzfahrzeuge

→ Beim SDV geht es nicht primär um Software. Es geht um den Geschäftswert.

Software ist zu einem primären Treiber für Wertschöpfung, Effizienz und Profitabilität in der Logistikbranche geworden. Während SDV-Strategien von der Konzeption zur

Implementierung übergehen, investieren Fahrzeughersteller, um die wachsende Komplexität zu bewältigen und die langfristige Wettbewerbsfähigkeit zu sichern.

1.1

Die SDV-Transformation ist bereits im Gang

Der Wandel hin zu Software-definierten Nutzfahrzeugen ist keine Zukunftsvision mehr. OEMs führen bereits heute SDV-Funktionen in aktuellen Fahrzeugprogrammen ein. Over-the-Air-Updates (OTA), zentralisierte Software-Plattformen und domänenübergreifende Rechenarchitekturen werden zunehmend Teil von Serienfahrzeugen.

Wie Abbildung 1 zeigt, basiert ein Großteil der Nutzfahrzeuge heute noch auf etablierten, verteilten E/E-Architekturen. Dieses Erbe ist nicht nur eine Einschränkung. Es spiegelt auch jahrzehntelanges Fahrzeug-Know-how, bewährte Engineering-Kompetenzen und ein tiefes Verständnis für Kundenanforderungen wider. Gleichzeitig erzeugt diese Transformation eine strategische Spannung. Neue Marktteilnehmer können einen „Clean-Field“-Ansatz verfolgen, während etablierte OEMs von unterschiedlichen Ausgangspunkten kommen und komplexe Fahrzeugportfolios, bestehende Plattformen

und etablierte Entwicklungsansätze weiterentwickeln müssen.

Die zentrale Herausforderung besteht daher nicht darin, ob SDV-Prinzipien übernommen werden sollen, sondern wie der Übergang von den heutigen OEM-spezifischen Architekturen zu zukünftigen Fahrzeuggenerationen effizient gelingt. Nutzfahrzeug-OEMs müssen auf ihren bestehenden Stärken aufbauen und gleichzeitig Fehler aus übermäßig ehrgeizigen Inhouse-Softwarestrategien vermeiden. Eine erfolgreiche SDV-Transformation erfordert daher eine pragmatische Balance zwischen interner Kompetenz, skalierbaren Plattformscheidungen und ausgewählten Partnerschaften.

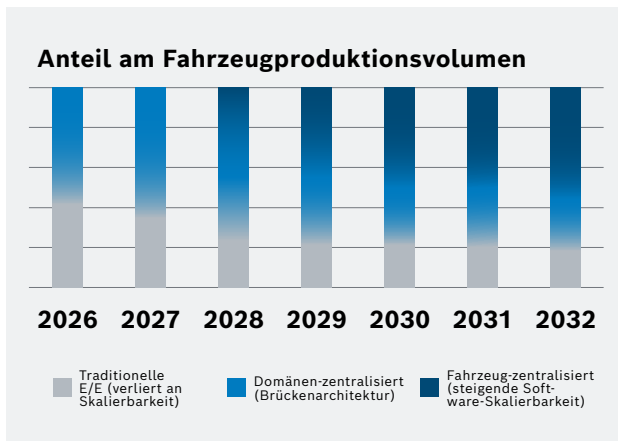


Abbildung 1: Die SDV-Transformation ist bei Nutzfahrzeugen bereits im Gange. Quelle: Interne Analyse von Bosch auf Basis ausgewählter europäischer, indischer und nord-amerikanischer Nutzfahrzeug-OEMs.

Definition: Das Software-definierte Nutzfahrzeug

Ein Software-definiertes Nutzfahrzeug (SDV) ist ein Fahrzeug, dessen Funktionalität zunehmend durch Software definiert wird. Sein primäres Ziel ist die Schaffung eines wirtschaftlichen Mehrwerts durch höhere Fahrzeugverfügbarkeit (Uptime) sowie eine verbesserte operative Effizienz und Entwicklungseffizienz.

1.2 SDV-Reifegrad: Jenseits vordefinierter Level

Die Fahrzeugindustrie beschreibt den SDV-Fortschritt oft anhand von Reifegraden. Obwohl sich die genauen Definitionen und Abgrenzungen zwischen den verschiedenen Frameworks der Branche unterscheiden, beschreiben die meisten

Modelle eine schrittweise Entwicklung von verteilten Fahrzeugarchitekturen hin zu softwarezentrierten Fahrzeug-Ökosystemen. Die in Abbildung 2 dargestellten Reifegrade dienen daher als anschauliches Referenzmodell für die Diskussion.

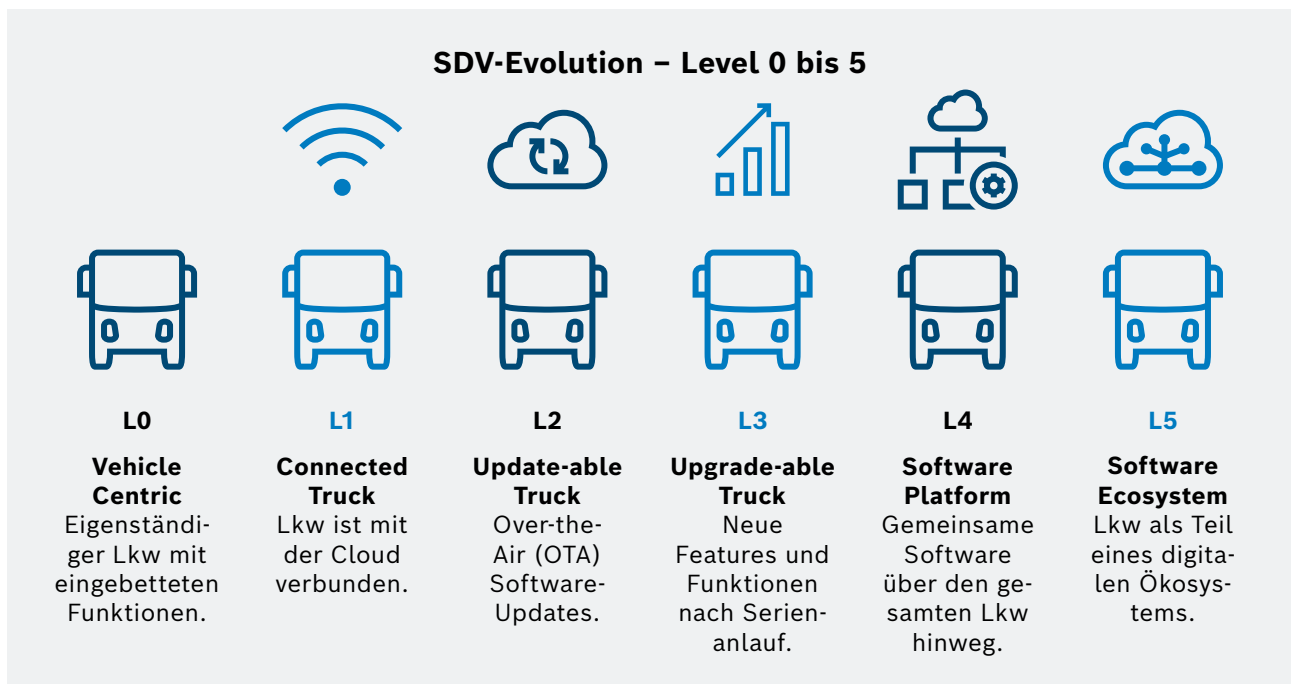


Abbildung 2: Illustratives SDV-Reifegradmodell für Nutzfahrzeuge, vergleiche [3].

1.3 Regionale Treiber prägen die SDV-Transformation

Der Wandel hin zu Software-definierten Nutzfahrzeugen wird von unterschiedlichen Prioritäten in den globalen Märkten angetrieben:

China

Elektrifizierung und digitale Innovation beschleunigen die Einführung Software-definierter Plattformen.

Nordamerika

Hochautomatisiertes Fahren treibt die Nachfrage nach skalierbarer Rechenleistung und fortschrittlichen Softwarefunktionen.

Europa

Cybersecurity, die Einhaltung von Vorschriften für Software-Updates und Sicherheitsbestimmungen sind wesentliche Treiber für architektonische Veränderungen.

Indien und Brasilien

Der Business Case für SDVs verlagert sich hin zu gesetzlichen Vorschriften und kostenoptimierten Hardware-Plattformen.

Daher müssen OEMs zunehmend unterschiedliche Anforderungen über Fahrzeugprogramme und Regionen hinweg unterstützen. Skalierbare Software-Plattformen bieten eine gemeinsame Grundlage, die eine regionale Differenzierung ermöglicht, ohne separate Fahrzeugarchitekturen zu schaffen. Obwohl die Treiber unterschiedlich sind, steigt in allen Regionen der Druck, die Uptime zu verbessern, die Betriebskosten zu senken und die Einführung neuer Funktionen zu beschleunigen.

1.4

SDV-Wertschöpfung im gesamten Ökosystem

Der Wert Software-definierter Nutzfahrzeuge reicht weit über das Fahrzeug selbst hinaus. Nutzfahrzeuge schaffen Wert für alle Stakeholder:

OEMs profitieren von wiederverwendbaren Software-Plattformen, digitalen Services über den gesamten Fahrzeuglebenszyklus und softwarebasierter Differenzierung. **Flottenbetreiber** profitieren von einer höheren Fahrzeugverfügbarkeit und niedrigeren Total Cost of Ownership (TCO). **Frachtkunden** erhalten mehr Transparenz und Rückverfolgbarkeit in den Logistikprozessen. **Fahrer** profitieren von kontinuierlich verbesserten Sicherheits- und Assistenzfunktionen.

Dieser Wert erstreckt sich auch auf das Zulieferer-Ökosystem. Tier-1- und Tier-2-Zulieferer können von klar definierten Softwareschnittstellen, wiederverwendbaren Komponenten und skalierbaren Integrationsmodellen profitieren, die es ihnen

ermöglichen, Fachwissen und spezialisierte Funktionen über mehrere OEM-Plattformen und Fahrzeugprogramme hinweg beizusteuern. Der Übergang zu Software-definierten Plattformen kann jedoch auch die Wertschöpfung, die Kompetenzen und das Risiko in der Wertschöpfungskette verschieben. OEMs, die bewusst mehr Verantwortung für Software und Integration übernehmen, können mehr Kontrolle erlangen, sehen sich aber auch mit höherer Komplexität, Kosten und Lieferrisiken konfrontiert.

Die effiziente Bereitstellung der Vorteile von SDV über verschiedene Fahrzeugklassen, regionale Märkte und Geschäftsmodelle hinweg erfordert skalierbare Software-Plattformen und eine starke Zusammenarbeit im Ökosystem. Dies ermöglicht es jedem Teilnehmer, sich auf seine Kernkompetenzen zu konzentrieren und gleichzeitig zu einem wirtschaftlicheren Software-definierten Fahrzeug beizutragen.

1.5

Balance zwischen Uptime, Fahrzeugvielfalt und Wirtschaftlichkeit

Die Implementierung Software-definierter Funktionen in Nutzfahrzeugen ist nicht nur eine technologische Herausforderung. OEMs müssen drei eng miteinander verbundene Ziele ausbalancieren: Uptime, Fahrzeugvielfalt und Fahrzeugwirtschaft-

lichkeit. Die Herausforderung besteht nicht darin, die Ziele einzeln zu optimieren, sondern alle drei bei einer Skalierung über Fahrzeugprogramme und Baureihen hinweg in Einklang zu bringen.



Abbildung 3: Die zentralen Ziele des Software-definierten Nutzfahrzeugs.

Uptime

Nutzfahrzeuge schaffen nur dann Wert, wenn sie in Betrieb sind. Fahrzeugarchitekturen müssen daher eine kontinuierliche Weiterentwicklung der Software, Over-the-Air-Updates und datengestützte Dienste unterstützen und gleichzeitig ungeplante Ausfallzeiten minimieren. Vorausschauende Wartung (Predictive Maintenance), Zustandsüberwachung (Condition Monitoring) und Ferndiagnose tragen dazu bei, die Verfügbarkeit über den gesamten Fahrzeuglebenszyklus zu verbessern.

Fahrzeugvielfalt

Fahrzeugvielfalt bezieht sich auf die große Vielfalt an regionalen Anforderungen, Fahrzeugkonfigurationen, Antriebssträngen, Radständen und Anpassungen durch Aufbauhersteller, die von einer gemeinsamen Fahrzeugplattform unterstützt werden müssen. Fahrzeugarchitekturen müssen daher diese Vielfalt beherrschen, ohne die Softwarekomplexität, den Validierungsaufwand oder die Integrationskosten kontinuierlich zu erhöhen.

Fahrzeugwirtschaftlichkeit

Fahrzeugwirtschaftlichkeit beschreibt die Fähigkeit von OEMs, Nutzfahrzeugplattformen wirtschaftlich über mehrere Fahrzeugprogramme, Regionen und Kundensegmente hinweg zu bauen und zu skalieren. Fahrzeugarchitekturen müssen daher die Wiederverwendung von Software, standardisierte Schnittstellen und skalierbare Plattformkomponenten ermöglichen und gleichzeitig doppelten Entwicklungsaufwand, Integrationskomplexität und die Kosten für die Wartung über den Lebenszyklus reduzieren.

Die TCO von Software-definierten Nutzfahrzeugen bleibt durch Investitions-, Betriebs- und Wartungskosten bestimmt. Die oben diskutierten Ziele beeinflussen jedes dieser Kostenelemente und damit den gesamten Business Case. Ihre relative Bedeutung hängt von Annahmen bezüglich Flottenauslastung, Fahrzeugverfügbarkeit, Software-Wiederverwendung und Lebenszykluskosten ab. Für Flottenbetreiber wiegt die Fahrzeugverfügbarkeit oft schwerer als inkrementelle Hardwarekosten, da selbst geringfügige Reduzierungen der Ausfallzeiten einen erheblichen Einfluss auf die Umsatzgenerierung und die betriebliche Effizienz haben können. Für OEMs werden Entwicklungsproduktivität, Software-Wiederverwendung und Wartungskosten über den Lebenszyklus immer wichtiger. Der wirtschaftliche Wert von Software-definierten Architekturen hängt daher nicht nur von der Reduzierung der Hardwarekosten ab, sondern auch von der Senkung des Integrationsaufwands, des Validierungsaufwands und der Kosten für die Wartung von Fahrzeugflotten über deren gesamten Lebenszyklus.

02

Evolution der Architektur: Drei Wegbereiter für skalierbare SDVs

→ Fähigkeiten skalieren, ohne die Komplexität zu erhöhen.

Das Ziel ist es, skalierbare, Software-definierte Nutzfahrzeuge zu ermöglichen. Dies wird nicht durch eine bestimmte Anzahl von zentralisierten Recheneinheiten (Compute Units) erreicht, sondern durch die Fähigkeit, Software zu entwickeln, zu integrieren, zu validieren und bereitzustellen und gleichzeitig deren Abhängigkeiten von der zugrunde liegenden Hardware zu steuern.

Anstatt ein komplettes Redesign zu fordern, benötigen OEMs einen pragmatischen Weg, der verschiedene Fahrzeugprogramme und eine langfristige Flottenrentabilität unterstützt. Drei Wegbereiter unterstützen diese Transformation:

2.1

Funktionale Verteilung Functional Distribution

Nutzfahrzeuge verfügen über eine breite Palette von Funktionen, von grundlegenden Fahrzeugfunktionen bis hin zu fortschrittlicher Fahrerassistenz (ADAS). Das Hinzufügen neuer Funktionen durch dedizierte Steuergeräte (ECUs) und eng gekoppelte Software-Stacks würde die Softwarefragmentierung, den Integrationsaufwand und die Lebenszykluskosten erhöhen. Die funktionale Verteilung befasst sich mit der architektonischen Frage, wo Funktionen innerhalb des Fahrzeugs ausgeführt werden sollen, wobei zwischen lokaler Ausführung, zentraler Verarbeitung und systemweiter Koordination abgewogen werden muss.

Fahrzeugfunktionen basieren zunehmend auf funktionalen Ketten, die sich über traditionelle Domänengrenzen hinaus

erstrecken. Sensorik, Verarbeitung, Entscheidungsfindung und Steuerung können in verschiedenen Teilen des Fahrzeugs ablaufen oder von vernetzten Backend-Systemen unterstützt werden. Bei der funktionalen Verteilung geht es daher nicht darum, jede Funktion zu zentralisieren. Es geht darum zu entscheiden, wo jede Funktion basierend auf Sicherheit, Leistung, Datenverfügbarkeit und Kosten ausgeführt werden soll.

Bestehende Funktionszuweisungen müssen nicht sofort geändert werden. OEMs können zunächst Funktionen neu zuweisen, bei denen gemeinsam genutzte Ressourcen und eine schnellere Funktionsentwicklung über den Fahrzeuglebenszyklus hinweg einen klaren Vorteil bieten. Stabile Schnittstellen und gemeinsame Software-

Plattformen helfen, die Kopplung an die Ausführungsumgebung zu reduzieren. Dies begrenzt das Transformationsrisiko und unterstützt gleichzeitig die Wiederverwendung über Fahrzeugprogramme und Generationen hinweg. Der erreichbare Grad

der Wiederverwendung bleibt jedoch funktionsspezifisch. Sicherheitskritische oder sensorintensive Funktionen können eng an bestimmte Sensoren, Rechenressourcen und Echtzeitanforderungen gebunden bleiben.

2.2

Skalierbare Rechenleistung

Scalable Compute

Skalierbare Rechenleistung ist eine Voraussetzung für das Software-definierte Nutzfahrzeug. Verschiedene Fahrzeugklassen und Anwendungen benötigen unterschiedliche Rechen-Topologien, was einen universellen Architekturansatz (One-Size-Fits-All) unpraktikabel macht. Eine Plattform, die für den anspruchsvollsten Anwendungsfall konzipiert ist, kann für Einstiegssegmente unwirtschaftlich sein, während eine kostenoptimierte Lösung möglicherweise nicht genügend Leistung für fortschrittliche Anwendungen bietet.

Skalierbare Rechenleistung bietet die für jedes Fahrzeugprogramm erforderliche Leistung durch unterschiedliche Hardware-Konfigurationen. Verarbeitungskapazität, Speicher und Kommunikationsbandbreite können an die Anwendung angepasst werden. Datenintensive Funk-

tionen wie videobasierte Umfelderkennung oder fortschrittliche Assistenzsysteme können eine höhere Rechenklasse mit dedizierter Hardware-Beschleunigung erfordern. Skalierbare Rechenleistung ermöglicht es OEMs, diese Leistung dort bereitzustellen, wo sie benötigt wird, ohne jede Fahrzeugvariante überdimensionieren zu müssen.

Hardware kann nicht als vollständig austauschbar behandelt werden. Echtzeitverhalten, Sicherheit, thermische Grenzen und Hardware-Beschleuniger beeinflussen weiterhin die Softwareausführung. Klare Abstraktionsschichten und kontrollierte Abhängigkeiten helfen, unnötige Hardware-Kopplungen zu reduzieren und gleichzeitig den Zugriff auf spezifische Fähigkeiten dort sicherzustellen, wo es erforderlich ist.

2.3

Zonale Abstraktion Zonal Abstraction

Nutzfahrzeuge weisen eine hohe Vielfalt bei Antriebssträngen, Radständen, Achskonfigurationen und Anwendungen von Aufbauherstellern auf. Diese Unterschiede wirken sich auf physische Schnittstellen,

die Verkabelung und die Fahrzeugintegration aus. Wenn diese Unterschiede direkt in der zentralen Rechenplattform abgebildet werden, führt jedes Fahrzeugprogramm zu erhöhter Integrationskom-

plexität und neuem Validierungsaufwand. Variantenspezifische I/O- und lokale Steuerungsfunktionen sollten daher durch stabile Fahrzeugschnittstellen abstrahiert werden.

Zonale Controller aggregieren diese Schnittstellen und schirmen die zentrale Rechenhardware und -software von fahrzeugspezifischen Implementierungen ab. Physische Änderungen können dann lokal gehandhabt werden, ohne höhere Softwareebenen zu beeinträchtigen. Dies erfordert konsistente Daten-, Steuerungs- und Serviceschnittstellen, die eine stabile digitale Abbildung für die zentrale Plattform bieten und gleichzeitig die Implementierungsflexibilität auf lokaler Ebene erhalten. Zonale Abstraktion ist daher mehr als nur die Verlagerung von

Steuergeräten. Ihr Zweck ist es, digitale Konsistenz oberhalb der zonalen Ebene zu schaffen.

Darüber hinaus kann die Abstraktion dieser Schnittstellen mit zonalen Controllern die Komplexität und das Gewicht des Kabelbaums reduzieren. Der Grad der Zonalisierung entwickelt sich über Fahrzeugprogramme und Generationen hinweg. Bestehende Schnittstellen und lokale Steuerungsfunktionen müssen nicht sofort ersetzt werden. OEMs können eine zonale Abstraktion dort einführen, wo physische Varianz oder wiederholter Integrationsaufwand einen klaren Vorteil schaffen. Dies schützt die zentrale Rechenplattform und hält gleichzeitig die fahrzeugspezifische Integration nahe am Ort des Geschehens.

Wesentliche Vorteile der Architekturevolution

Funktionale Verteilung

Funktionen können dort zugewiesen werden, wo sie die technischen und wirtschaftlichen Anforderungen am besten erfüllen.

Skalierbare Rechenleistung

Skalierbare Rechenleistung vermeidet unnötige Hardwarekosten und unterstützt gleichzeitig anspruchsvolle Anwendungen.

Zonale Abstraktion

Stabile Schnittstellen schirmen zentrale Rechenplattformen von fahrzeugspezifischen Implementierungen ab.

2.4 Einführung in Phasen: Entwicklung ohne Disruption

Diese drei architektonischen Wegbereiter können schrittweise und in unterschiedlichen Geschwindigkeiten eingeführt

werden. OEMs können mit ausgewählten Fahrzeugprogrammen und Funktionen beginnen, bei denen der technische oder

wirtschaftliche Mehrwert besonders deutlich ist. Die Reihenfolge und der Umfang hängen von bestehenden Architekturen, Produktstrategien und Investitionsprioritäten ab.

Phase 1: Etablierung einer skalierbaren Basis

Phase 1 konsolidiert ausgewählte Fahrzeugfunktionen und etabliert skalierbare Rechenleistung, gemeinsame Software-Grundlagen und stabile Schnittstellen. Dies erhöht die Wiederverwendung von Software und begrenzt gleichzeitig das Transformationsrisiko.

Phase 2: Erweiterung der domänenübergreifenden Integration und Abstraktion

Phase 2 erweitert die Integration von Fahrzeugfunktionen über traditionelle Domänen hinweg und stärkt die Abstraktion, sodass fahrzeugspezifische Unterschiede

lokal gekapselt bleiben. Dies ermöglicht die Wiederverwendung gemeinsamer Plattformelemente in einer größeren Vielfalt von Fahrzeugvarianten.

Phase 3: Ermöglichen modularer, fahrzeugweiter Architekturen

Phase 3 geht über die lokale Abstraktion hinaus und etabliert Modularität als fahrzeugweites Architekturprinzip. Skalierbare Rechenkonfigurationen und konsistente Schnittstellen ermöglichen eine unabhängige Entwicklung von Software und Hardware.

Jede Phase baut auf den zuvor etablierten Fähigkeiten auf, ohne dass alle Fahrzeugprogramme dieselbe Geschwindigkeit verfolgen müssen. Die resultierende Architektur bildet auch die Grundlage für verteilte Intelligenz und zukünftige Rechen- und Energietechnologien.

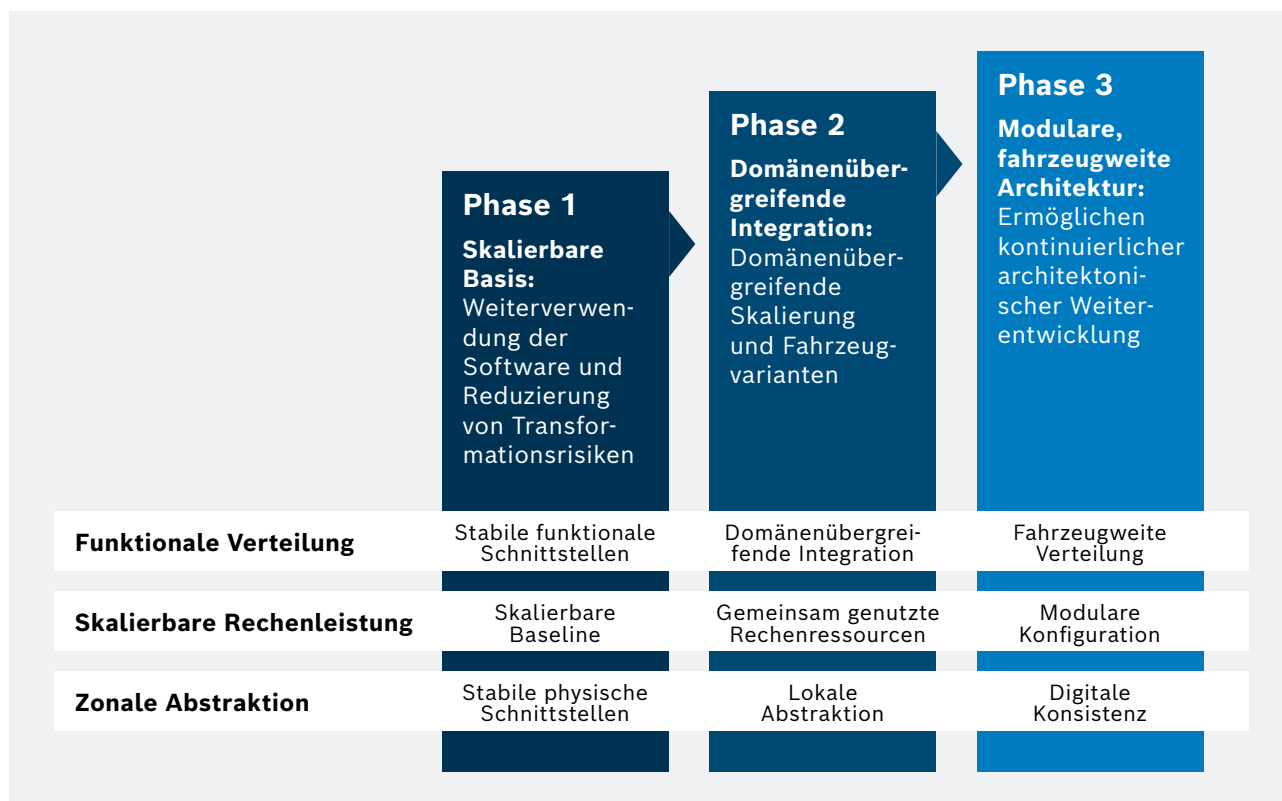


Abbildung 4: Ein phasenweiser Weg zu skalierbaren SDV-Architekturen. Quelle: Bosch-Phasenmodell.

03

SDV-Plattform: Standardisierung schafft Skalierbarkeit

→ Differenzieren, wo es darauf ankommt. Standardisieren und partnerschaftlich zusammenarbeiten, wo nicht.

Die Produktionsvolumina von Nutzfahrzeugen machen die Entwicklung und langfristige Wartung vollständig proprietärer SDV-Plattformen zunehmend zu einer

Herausforderung. OEMs benötigen daher einen Ansatz, der Differenzierung und Entwicklungseffizienz ausbalanciert.

3.1

Von fragmentierten Lösungen zu wiederverwendbaren Plattformen

In traditionellen E/E-Architekturen werden Fahrzeugfunktionen typischerweise auf dedizierten ECUs implementiert und unabhängig gewartet. Da der Softwareanteil und die Systemkomplexität weiter wachsen, führt dieser Ansatz zu hohem Integrationsaufwand, doppelter Entwicklung und steigenden Wartungskosten über den Lebenszyklus. SDV-Architekturen begegnen diesen Herausforderungen, indem sie Funktionalität auf gemeinsamen Rechenplattformen konsolidieren und eine gemeinsame Software-Plattform über Fahrzeugprogramme hinweg etablieren. Diese Konsolidierung verlagert den Fokus von individuell verwalteten ECUs auf eine gemeinsame Software-Grundlage, die mehrere Fahrzeugfunktionen unterstützen kann.

Abbildung 5 veranschaulicht eine vereinfachte, mehrschichtige Architektur einer leistungsstarken SDV-Plattform. Die Basis bildet die Hardware-Plattform, die die für

die Fahrzeugfunktionen erforderliche Leistung, Sicherheit und Ressourcen bereitstellt. Darauf aufbauend befindet sich die Software-Plattform, die die Laufzeitumgebung und gemeinsame Dienste für die Fahrzeuganwendungen zur Verfügung stellt. Diese Schicht schafft die Grundlage für die Entkopplung von Anwendungen von der zugrundeliegenden Hardware und kann die Einführung zukünftiger Rechner-Hardware vereinfachen.

Höhere Grade an Software-Portabilität und Wiederverwendung erfordern jedoch, dass standardisierte Schnittstellen und gemeinsame Datenmodelle konsistent über Fahrzeugprogramme hinweg definiert und angewendet werden. Diese Mechanismen reduzieren den fahrzeugspezifischen Integrationsaufwand und erleichtern die Integration von Anwendungen, Flottendiensten und Funktionen von Aufbauherstellern über den gesamten Fahrzeuglebenszyklus.



Abbildung 5: Eine vereinfachte, mehrschichtige Ansicht einer SDV-Plattform.

3.2

Nutzung der Expertise des Ökosystems

Eine Multi-Layer-SDV-Architektur hilft OEMs und ihren Technologiepartnern zu entscheiden, wo proprietäre Fähigkeiten erforderlich sind, wo gemeinsame Branchengrundlagen den Entwicklungsaufwand reduzieren können und wo produktionsreife Integrations- und Systemexpertise unerlässlich ist. Je nach Sourcing-Strategie können Partner auf verschiedenen Ebenen der SDV-Plattform beitragen. Hardware-Zulieferer stellen die Rechen-, Netzwerk- und Safety-Grundlage bereit. Software-Plattformanbieter steuern Laufzeitumgebungen, Abstraktionsschichten und gemeinsame Dienste bei. Funktionslieferanten können ausgewählte Fahrzeugfunktionen oder domänenspezifische Fähigkeiten liefern, während Aufbauhersteller und Anbieter digitaler Dienste kundenspezifische Anwendungen und Dienste auf Basis standardisierter Schnittstellen integrieren.

Für die Software-Plattform stehen Industrie-Standard- und Open-Source-Grundlagen sowohl für sicherheitsrelevante als auch für nicht sicherheitsrelevante Fahrzeugfunktionen zur Verfügung. Diese gemeinsamen Grundlagen können doppelte Entwicklungsaufwände reduzieren und eine branchenweite Kostenteilung für nicht differenzierende Softwareschichten ermöglichen, während die für OEM-spezifische Implementierungen erforderliche Flexibilität erhalten bleibt. OEMs und Technologiepartner können auf diesen Plattformen aufbauen und Entwicklungsressourcen auf fahrzeugspezifische Fähigkeiten und kundenorientierte Differenzierung konzentrieren. Wiederverwendbare Software-Grundlagen sind jedoch nicht automatisch produktionsreife Fahrzeugplattformen. Sie erfordern Integration, Validierung, Safety / Security und Wartung

über den Lebenszyklus. Technologiepartner können helfen, diese Grundlagen zu industrialisieren und sie mit ausgewählten Fähigkeiten auf Systemebene zu ergänzen. Dies ermöglicht es OEMs, die Plattformentwicklung zu beschleunigen und gleichzeitig die Kontrolle über die kundenspezifische Differenzierung zu behalten.

Software kann jedoch nicht ohne Einschränkungen über Fahrzeugprogramme hinweg

wiederverwendet werden. Der erreichbare Grad der Wiederverwendung hängt von der Art der Funktion, ihrer Sicherheitsrelevanz und ihren Abhängigkeiten von Sensoren, Rechenressourcen und Echtzeitleistung ab. Multi-Layer-Architekturen beseitigen den Anpassungsaufwand nicht vollständig, reduzieren jedoch dessen Auswirkungen, indem sie wiederverwendbare Plattformdienste von funktions- und hardwarespezifischen Implementierungen trennen.

3.3

Skalierung der Softwareintegration und -validierung

Mit zunehmendem Softwareanteil werden virtuelle Entwicklung und kontinuierliches Testen unerlässlich, um Komplexität und Integrationsrisiken zu bewältigen. Entwicklungsumgebungen auf Basis virtualisierter Steuergeräte (vECUs) ermöglichen Softwareentwicklung, Integration und Validierung bereits, bevor die vollständige Zielhardware verfügbar ist.

Für die Skalierung dieses Ansatzes sind Continuous Integration (CI) und automatisierte Tests notwendig, damit Softwareänderungen kontinuierlich validiert werden

können. Integrationsprobleme werden deutlich früher erkannt und können behoben werden, noch bevor sie auf physischen Prototypen sichtbar sind.

Obwohl die endgültige Validierung an die Zielhardware und das physische Fahrzeug gebunden bleibt, ermöglicht die virtuelle Entwicklung einen früheren Aufbau der Software-Reife. Dadurch lassen sich Integrationsrisiken in späten Projektphasen reduzieren und ein skalierbarer Entwicklungsansatz über mehrere Fahrzeugprogramme hinweg unterstützen.

Wesentliche wirtschaftliche Vorteile einer standardisierten SDV-Plattform

Wiederverwendbare Plattform

SDV-Plattformen reduzieren die Fragmentierung und ermöglichen die Wiederverwendung über Fahrzeugprogramme und Varianten hinweg.

Gezielter Beitrag des Ökosystems

OEMs und Technologiepartner können ihre Expertise dort konzentrieren, wo sie den größten Mehrwert schafft.

Frühe Softwarereife

Virtuelle Entwicklung und kontinuierliches Testen verlagern die Integration und Validierung in eine frühere Phase des Entwicklungsprozesses.

04

Organisatorische Transformation: Struktur, Geschwindigkeit und Mindset in Einklang bringen

→ Produkte können sich nicht weiterentwickeln, wenn Organisationen stillstehen.

Software-definierte Fahrzeuge erfordern mehr als nur neue Technologien. Sie verändern grundlegend die Art und Weise, wie Nutzfahrzeuge entwickelt werden, und treiben den Bedarf an neuen Entwicklungsprozessen, Governance-Modellen und Organisationsstrukturen voran. Da sich Software und Hardware in unterschiedlichen Geschwindigkeiten entwickeln, müssen Organisationen kontinuierliche Software-Innovationen mit den Zuverlässigkeits-, Validierungs- und Sicherheitsanforderungen von Nutzfahrzeugen in Einklang bringen.

Die Bewältigung dieser Herausforderung erfordert ein Two-Speed-Operating-Model, in dem beide Welten in einem unterschiedlichen Takt arbeiten und durch End-to-End-Wertströme und eine starke funktionsübergreifende Zusammenarbeit eng miteinander verbunden bleiben. Anstatt bestehende Strukturen über Nacht zu ersetzen, müssen OEMs das Nebeneinander von traditioneller Fahrzeugentwicklung und software-zentrierten Organisationen ermöglichen. Gleichzeitig sind eine strategische Ausrichtung und das Engagement der Führungsebene unerlässlich,

um das Zielbetriebsmodell zu etablieren, während die Implementierung einem schrittweisen Übergang folgen sollte. Organisationen können dadurch neue Fähigkeiten entwickeln, ohne das Kerngeschäft zu gefährden.

Über Prozesse und Strukturen hinaus erfordert die SDV-Transformation einen Wandel im Mindset. Organisationen müssen sich von einer komponentenorientierten Projektabwicklung hin zu einem produkt- und plattformzentrierten Denken entwickeln. Dies erfordert starke Softwarekompetenzen in den Bereichen Architektur, Integration und Lifecycle-Management sowie eine klare Verantwortung für gemeinsame Plattformen und Software-Assets. Mit wachsendem Softwareanteil wird die Plattformverantwortung (Platform Ownership) entscheidend, um gemeinsame Schnittstellen aufrechtzuerhalten, die architektonische Konsistenz durchzusetzen und die Wiederverwendung über Fahrzeuggenerationen hinweg zu maximieren. Dies hilft, eine Fragmentierung zu verhindern und stellt sicher, dass die Software-Evolution mit der langfristigen Fahrzeugstrategie im Einklang bleibt.

05

Fazit und Ausblick

→ Software definiert das Fahrzeug. Intelligenz wird seine Zukunft gestalten.

Software-definierte Nutzfahrzeuge sind eine architektonische Herausforderung. Wachsender Softwareanteil, zunehmende funktionale Komplexität und hohe Fahrzeugvielfalt stellen neue Anforderungen an die Fahrzeugarchitekturen. Funktionale Verteilung, skalierbare Rechenleistung und zonale Abstraktion bilden die Grundlage, um diesen Anforderungen zu begegnen. Zusammen ermöglichen sie es OEMs, Software von fahrzeugspezifischen Implementierungen zu trennen, die Wiederverwendung über Fahrzeugprogramme hinweg zu erhöhen und eine kontinuierliche Softwareentwicklung zu unterstützen.

Wiederverwendbare Software-Plattformen, standardisierte Schnittstellen und virtuelle Integration sind wichtige Wegbereiter, doch die Transformation wird nicht allein durch Technologie erreicht. Der Erfolg hängt auch von starker Software-Expertise, angepassten Arbeitsweisen und einer konsequenten architektonischen Governance ab. Zusammen bilden diese Elemente die organisatorische und technische Grundlage, die zur Realisierung Software-definierter Nutzfahrzeuge erforderlich ist.

Auf diesen Grundlagen können OEMs vom individuell entwickelten Fahrzeug zu software-getriebenen Fahrzeugplattformen übergehen. Funktionen, Daten und Rechenressourcen können zunehmend über eingebettete Systeme, zentrale Fahrzeugcomputer und vernetzte Plattformen hinweg interagieren. Intelligenz kann dann dort platziert werden, wo sie den größten Wert für Sicherheit, Uptime und operative Effizienz schafft.

Diese Entwicklung ebnet den Weg zum KI-definierten Fahrzeug (AI-defined vehicle). KI kann etablierte Systeme ergänzen, indem sie den Flottenbetrieb weiter verbessert, vorausschauende Wartung unterstützt, den Energieverbrauch optimiert und die Fahrerassistenz verbessert. Gleichzeitig haben Fortschritte bei Rechen- und Fahrzeug-Energiearchitekturen das Potenzial, neue Möglichkeiten zu erschließen. Zukünftig ermöglichen modulare Hardwarekonzepte wie Chiplets eine flexiblere Skalierung der Rechenleistung, während ausgewählte 48-V-Anwendungen den steigenden Energiebedarf decken. Der entscheidende Schritt liegt in der Fähigkeit, Funktionen, Daten, Rechenleistung und Intelligenz im gesamten Fahrzeug miteinander zu koordinieren – weit über die Einführung einzelner Technologien hinaus.

Schlusswort



Dieses Whitepaper baut auf einer klaren Überzeugung auf: Transformation gelingt nur gemeinsam. Die Komplexität zu beherrschen und das Potenzial Software-definierter Fahrzeuge auszuschöpfen, ist nur durch eine enge Zusammenarbeit innerhalb der Branche möglich. Die in diesem Whitepaper geteilten Erkenntnisse unterstreichen diese Notwendigkeit.

Die Branche steht nun an einem Punkt, an dem die Richtung klar ist: Software wird zu einem zentralen Werttreiber, E/E-Architekturen konsolidieren sich, und Fahrzeuge entwickeln sich kontinuierlich über ihren Lebenszyklus weiter. Die Herausforderung besteht darin, diese Transformation umzusetzen und dabei die Kontrolle über Komplexität und Kosten zu behalten.

Diese Herausforderung ist besonders relevant für Nutzfahrzeuge, bei denen lange Produktlebenszyklen, hohe Integrationsanforderungen und die Vielfalt der Fahrzeuganwendungen die Komplexität des Wandels zu Software-definierten Fahrzeugen weiter erhöhen. Die erfolgreiche Bewältigung dieser Herausforderungen erfordert verschiedene Arten von Partnerschaften auf unterschiedlichen Ebenen und während der verschiedenen Implementierungsphasen. Kein Unternehmen kann diesen Übergang allein meistern.

Auch wenn die organisatorischen Ausgangspunkte unterschiedlich sind, bleiben die zugrunde liegenden Herausforderungen dieselben. Ob Sie bestehende Legacy-Architekturen modernisieren oder Ihre Softwareorganisation komplett neu definieren, die heute getroffenen Entscheidungen werden die zukünftige Wettbewerbsfähigkeit bestimmen.

Ganz gleich, an welchem Punkt der Transformation zum Software-definierten Fahrzeug Sie stehen: Treten Sie mit uns in Kontakt, um den Weg zu gestalten, der zu Ihrer Architektur und Ihren Anforderungen passt.

Mouna Romani

Vice President

Cross-Domain Computing Solutions

Commercial Vehicle Group

Bosch Mobility

Impressum

Referenzen:

- [1] Burkacky, O., Deichmann, J., Frank, S., Hepp, D., Rocha, A. When Code Is King: Mastering Automotive Software Excellence. McKinsey Center for Future Mobility, McKinsey & Company, 2021.
- [2] A. Herlt, J. Deichmann, M. Kellner, A. Khan. The big shift: Moving commercial vehicle OEMs to centralized E/E and Software. McKinsey & Company, 2024.
- [3] PwC Japan Group. What is an SDV (Software Defined Vehicle)? Defining SDVs beyond just vehicles. PwC Japan Group, 2024. <https://www.pwc.com/jp/en/knowledge/column/definition-of-sdv.html> Aufgerufen am: Aug. 5, 2026.
- [4] A. Friedel, N. Schuppensteiner, M. Willand. The Software Race: Are Chinese Automakers Taking the Lead? MHP GmbH, 2023.
- [5] Capgemini. Software-Defined Vehicles for the Commercial Vehicle Market. Capgemini, 2025.
- [6] ETAS GmbH. The Synergy of Eclipse S-CORE and Google's AAOS SDV: A Complementary Approach to the Software-Defined Vehicle. ETAS GmbH, 2026.
- [7] D. Maher. The Software-Defined Vehicle: A Structural Industry Reset. Seraph, 2026.

Kontaktieren Sie uns

Sie haben Fragen, Anmerkungen oder eine geschäftliche Anfrage? Zögern Sie nicht, sich mit uns in Verbindung zu setzen.

✉ contact.SDV-CV@bosch.com



SDV-Portfolio und weitere Whitepaper
<https://www.bosch-mobility.com/de/mobility-themen/sdv-portfolio/>



Erfahren Sie mehr auf unserer Webseite
www.bosch-mobility.com



Bosch Mobility on LinkedIn
www.linkedin.com/show-case/bosch-mobility/

Herausgeber:

Robert Bosch GmbH
 Bosch Mobility
 Commercial Vehicle Group
 Wernerstraße 51
 70469 Stuttgart, Germany