

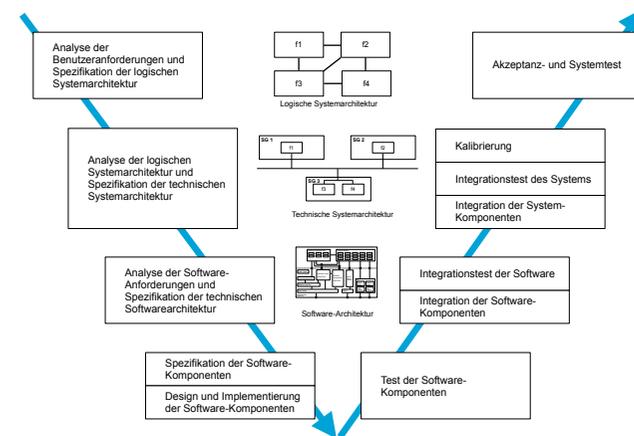
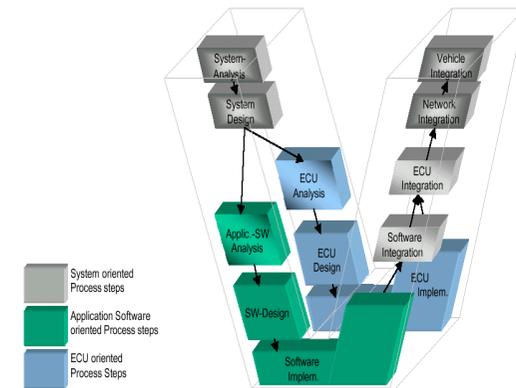


Vorlesung
Automotive Software Engineering
Teil 6 SW-Entwicklung (3)
Sommersemester 2015

Prof. Dr. rer. nat. Bernhard Hohlfeld
Bernhard.Hohlfeld@mailbox.tu-dresden.de
Technische Universität Dresden, Fakultät Informatik
Honorarprofessur Automotive Software Engineering

Vorlesung Automotive Software Engineering

Motivation und Überblick		
Beispiele aus der Praxis	SW-Entwicklung	Normen und Standards
	E/E-Entwicklung	
	Das Automobil	
	Die Automobilherstellung	
	Die Automobilbranche	



Lernziele SW-Entwicklung

- Vorgehensmodelle kennen
- Grundbegriffe des Systems Engineering kennen
- Den Kernprozess zur Entwicklung von elektronischen Systemen und Software im Fahrzeug verstehen
- Das V-Modell zur Software-Entwicklung in einer speziellen Ausprägung mit zahlreichen Beispielen kennen
- Die Unterstützungsprozesse zur Entwicklung von elektronischen Systemen und Software im Fahrzeug kennen

6. SW-Entwicklung

1. Vorgehensmodelle
2. Kernprozess
3. **Unterstützungsprozesse**

6. SW-Entwicklung / 3. Unterstützungsprozesse

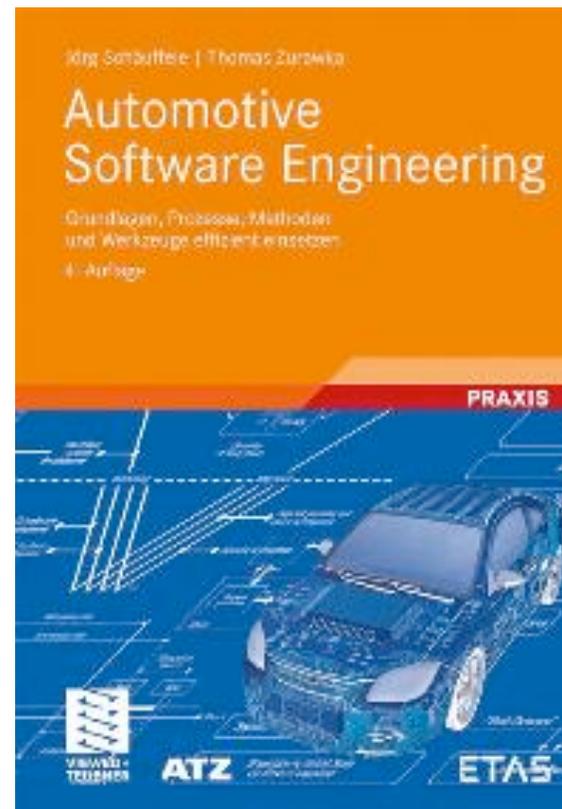
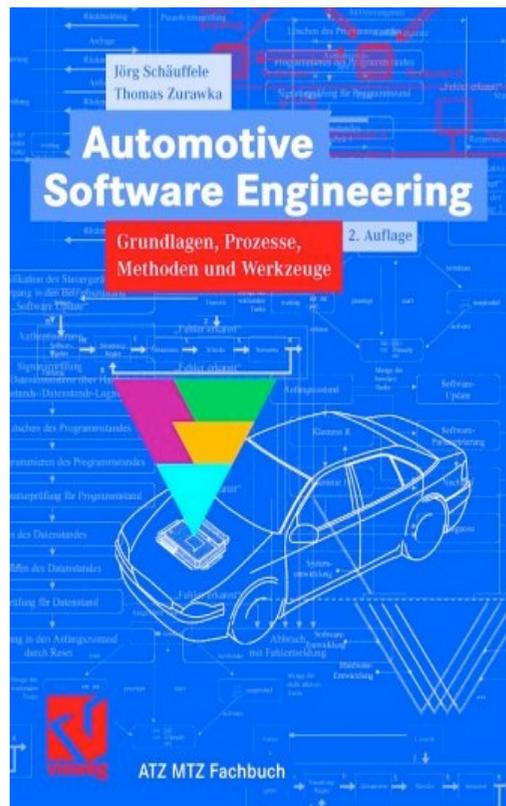
Unterstützungsprozesse für die Embedded Software Entwicklung

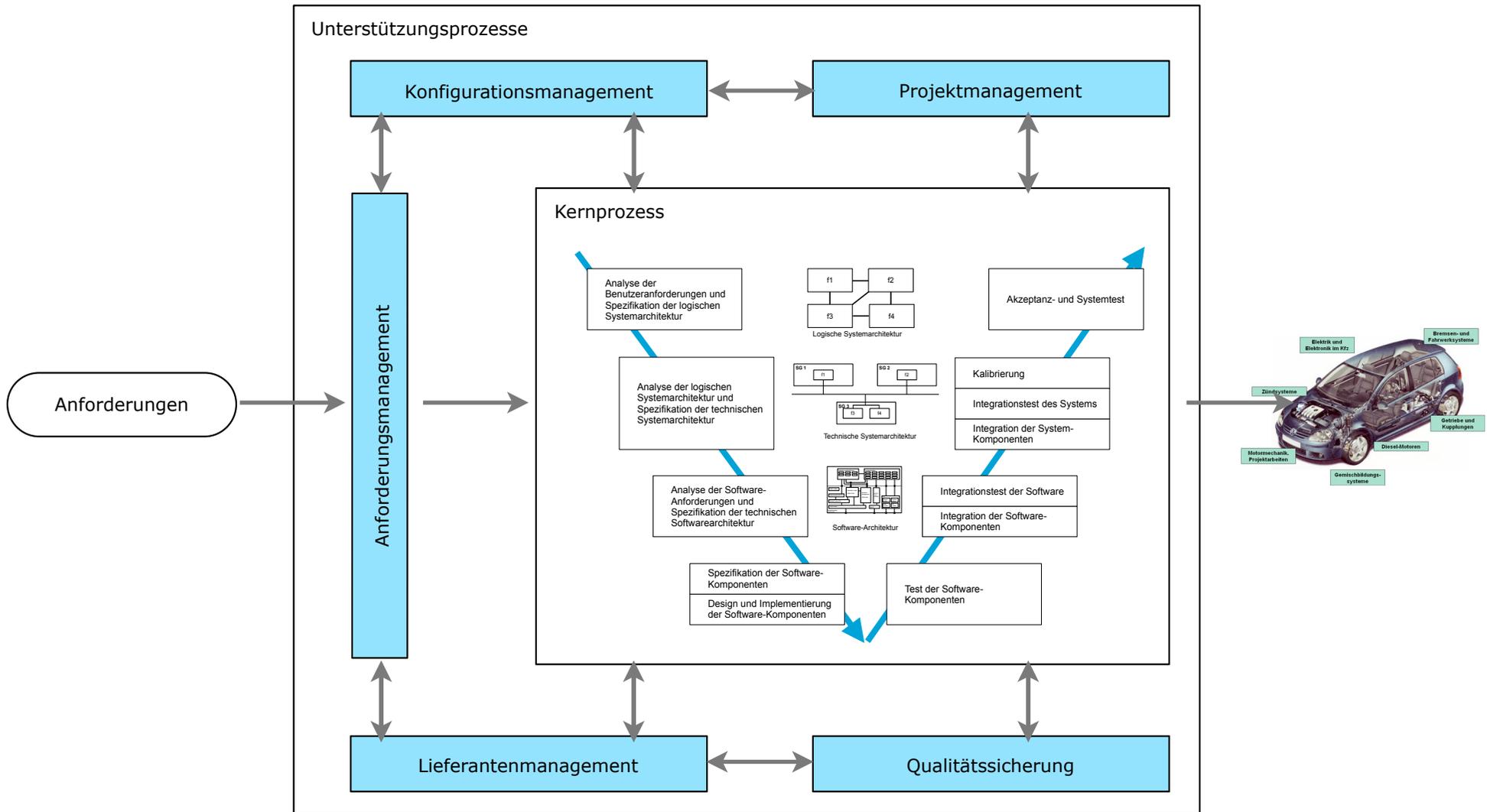
1. Vorgehensmodelle und Standards
2. Konfigurationsmanagement
3. Projektmanagement
4. Lieferantenmanagement
5. Anforderungsmanagement
6. Qualitätssicherung

Quelle

Kapitel 3

- Die Abbildungen werden mit freundlicher Genehmigung der Autoren verwendet.





6. SW-Entwicklung / 3. Unterstützungsprozesse

Unterstützungsprozesse für die Embedded Software Entwicklung

1. **Vorgehensmodelle und Standards**
2. Konfigurationsmanagement
3. Projektmanagement
4. Lieferantenmanagement
5. Anforderungsmanagement
6. Qualitätssicherung

Vorgehensmodelle und Standards

- CMMI Capability Maturity Model Integration
 - <http://www.sei.cmu.edu/cmmi>
 - <http://cmmiinstitute.com>
- SPICE Software Process Improvement and Capability Determination (ISO/IEC 15504-1)
 - <http://www.sqi.gu.edu.au/spice/>
- V-Modell
 - V-Model 1997
 - <http://www.v-modell.iabg.de>
 - V-Modell XT 2005
 - http://www.CIO.bund.de/DE/IT-Methoden/V-Modell_XT/v-modell_xt_node.html
 - flyXT
 - Unternehmensspezifische Anpassung des V-Modells XT bei EADS DE
 - Nicht öffentlich

Unterstützungsprozesse

Vorlesung Automotive SWE (Buch Schäuffele/ Zurawka)	V-Modell: Tätigkeitsbereiche	CMMI Reifegradstufe 2: Key Process Areas
Kernprozess	Systemerstellung	
Unterstützungsprozesse		
Konfigurationsmanagement	Konfigurationsmanagement	Konfigurationsmanagement
Projektmanagement	Projektmanagement	Projektplanung Projektverfolgung
Lieferantenmanagement	Teil des Projektmanagements	Lieferantenmanagement
Anforderungsmanagement	Teil des Projektmanagements	Anforderungsmanagement
Qualitätssicherung	Qualitätssicherung	Qualitätssicherung

Unterstützungsprozesse

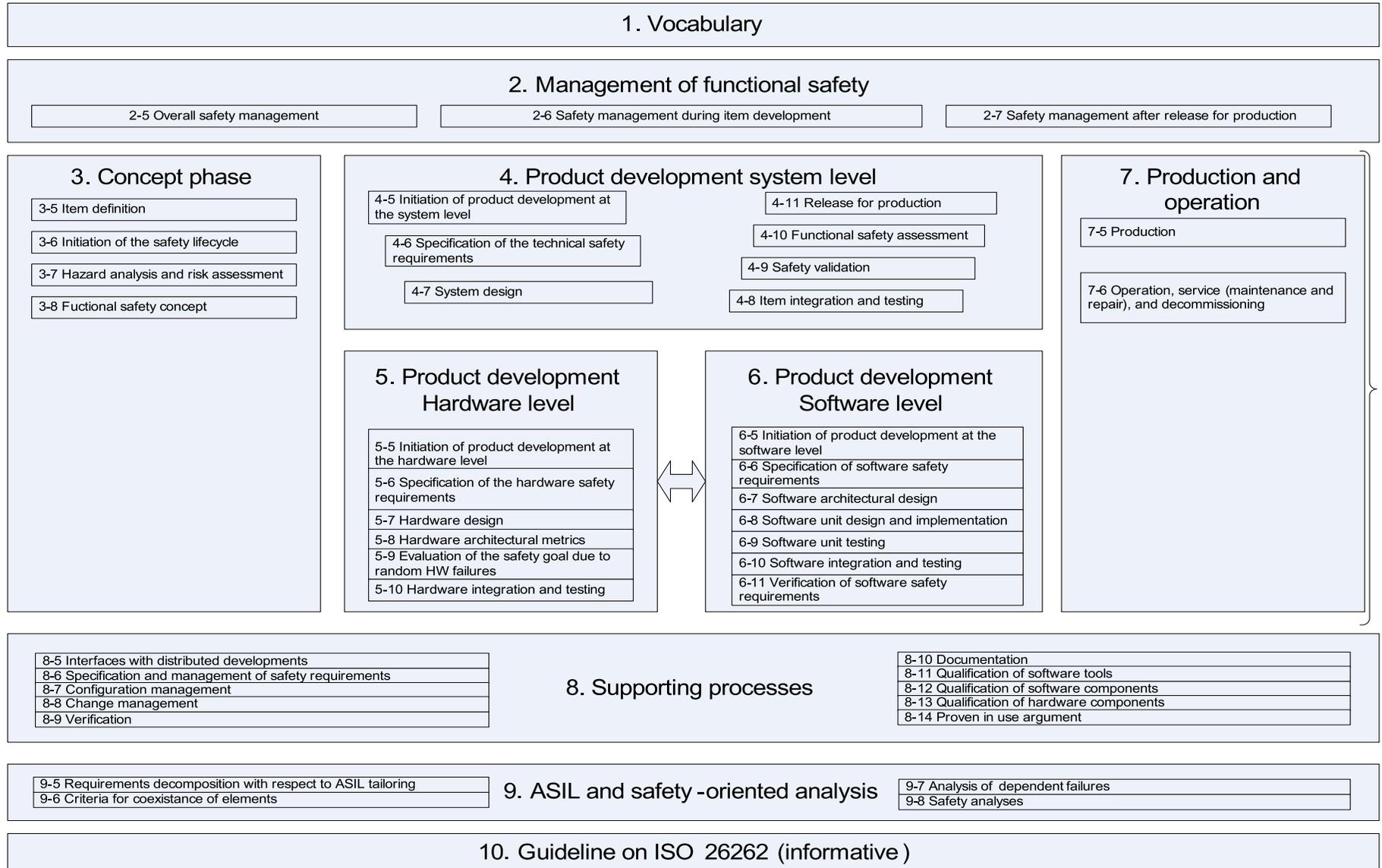
Vorlesung Automotive SWE (Buch Schäuffele/Zurawka)	BMW Group Standard embedded Software (GSeSW)
Kernprozess	Development processes
Teil des Kernprozesses	Software Development
Teil des Kernprozesses	Software Modification
Teil des Kernprozesses	Design Verification process
Teil des Kernprozesses	Testprocess

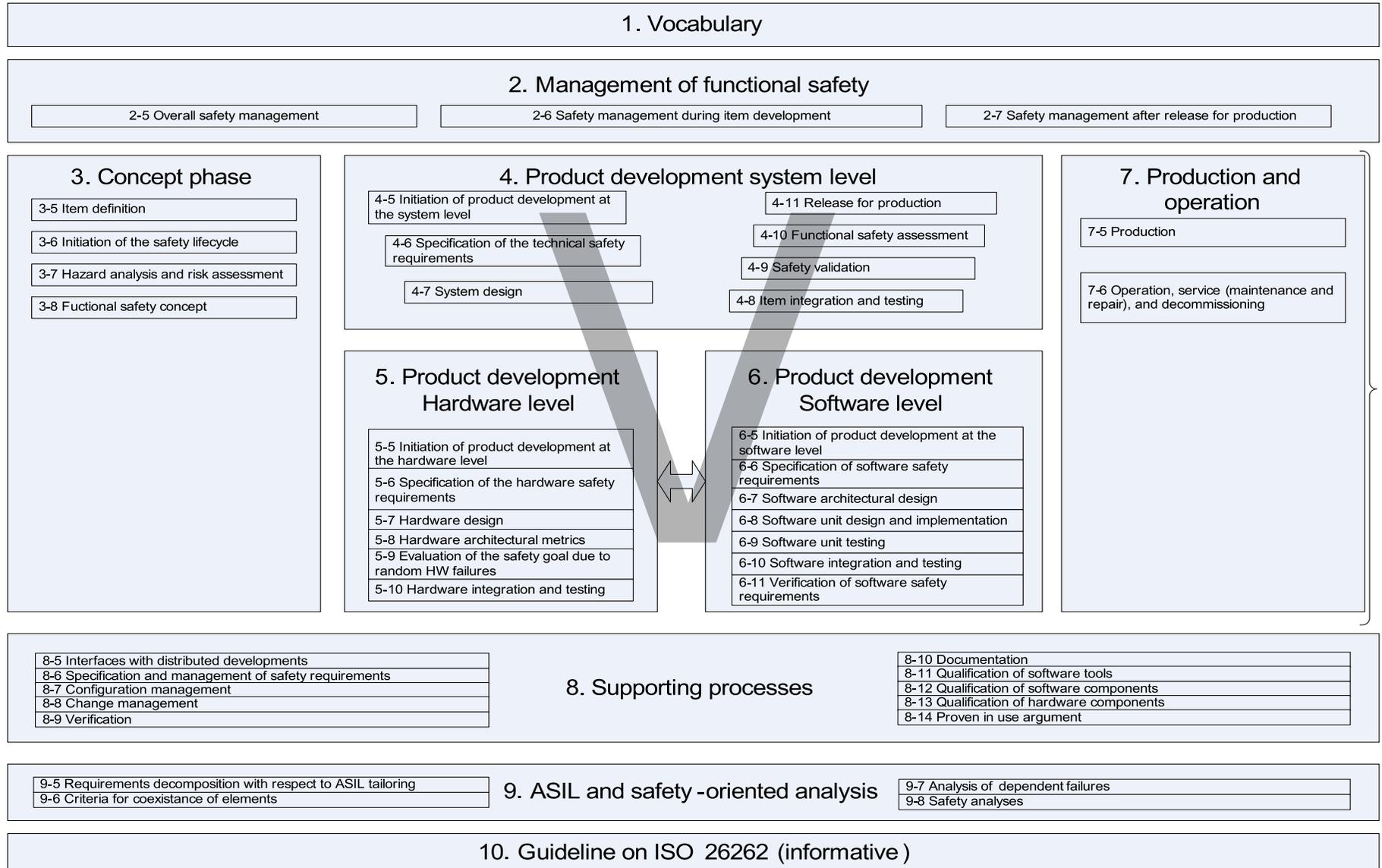
Unterstützungsprozesse

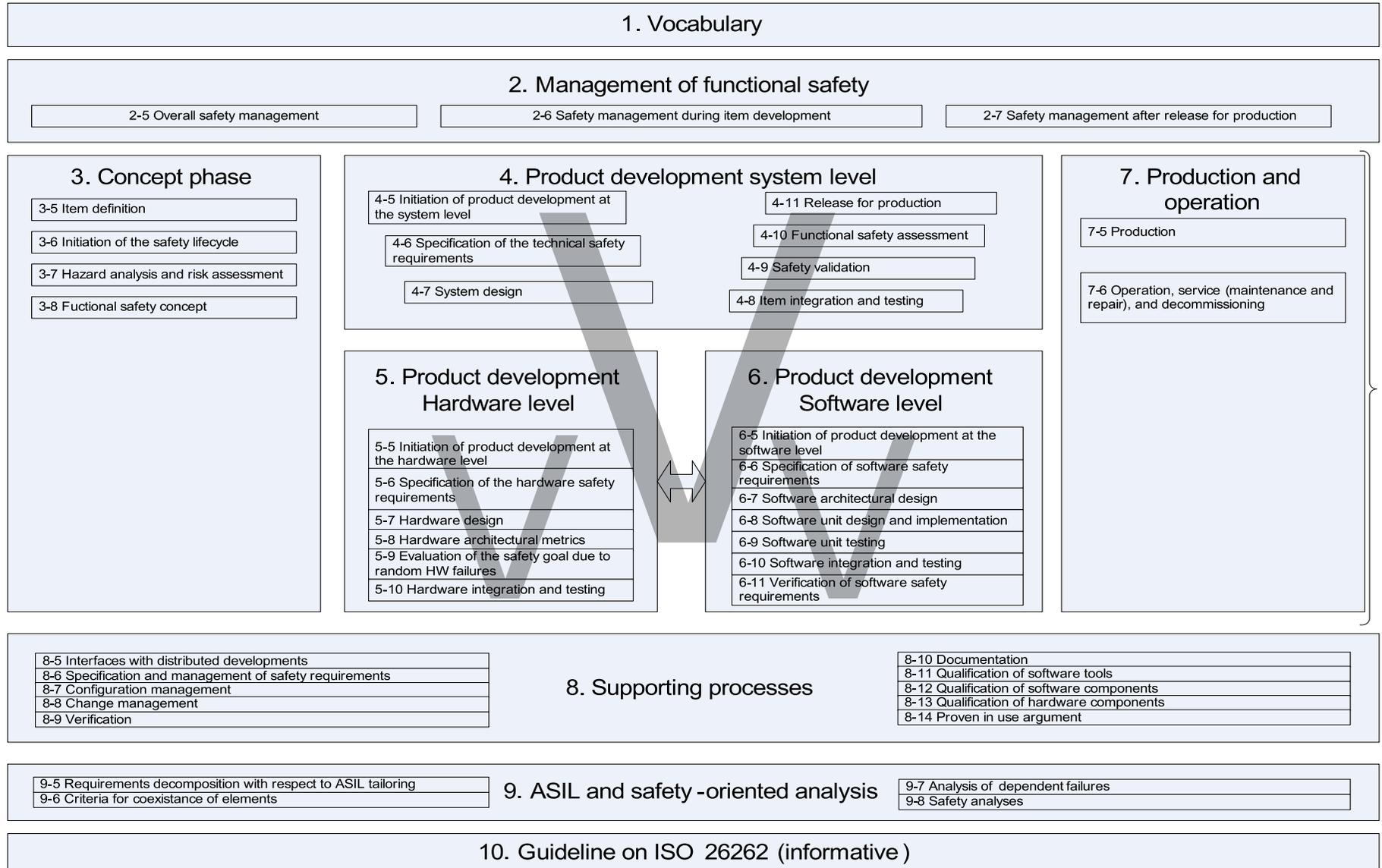
Vorlesung Automotive SWE (Buch Schäuffele/Zurawka)	BMW Group Standard embedded Software (GSeSW)
Unterstützungsprozesse	Organizational processes
Projektmanagement	Project Management
Teil des Projektmanagements	Risk Management
Teil des Projektmanagements	Role Assignment
Teilweise im Kernprozess (Erstellung Produktdokumentation)	Documentation
Teilweise im Kernprozess (Erstellung Produktdokumentation)	Training and Qualification
	Planning Training and Qualification

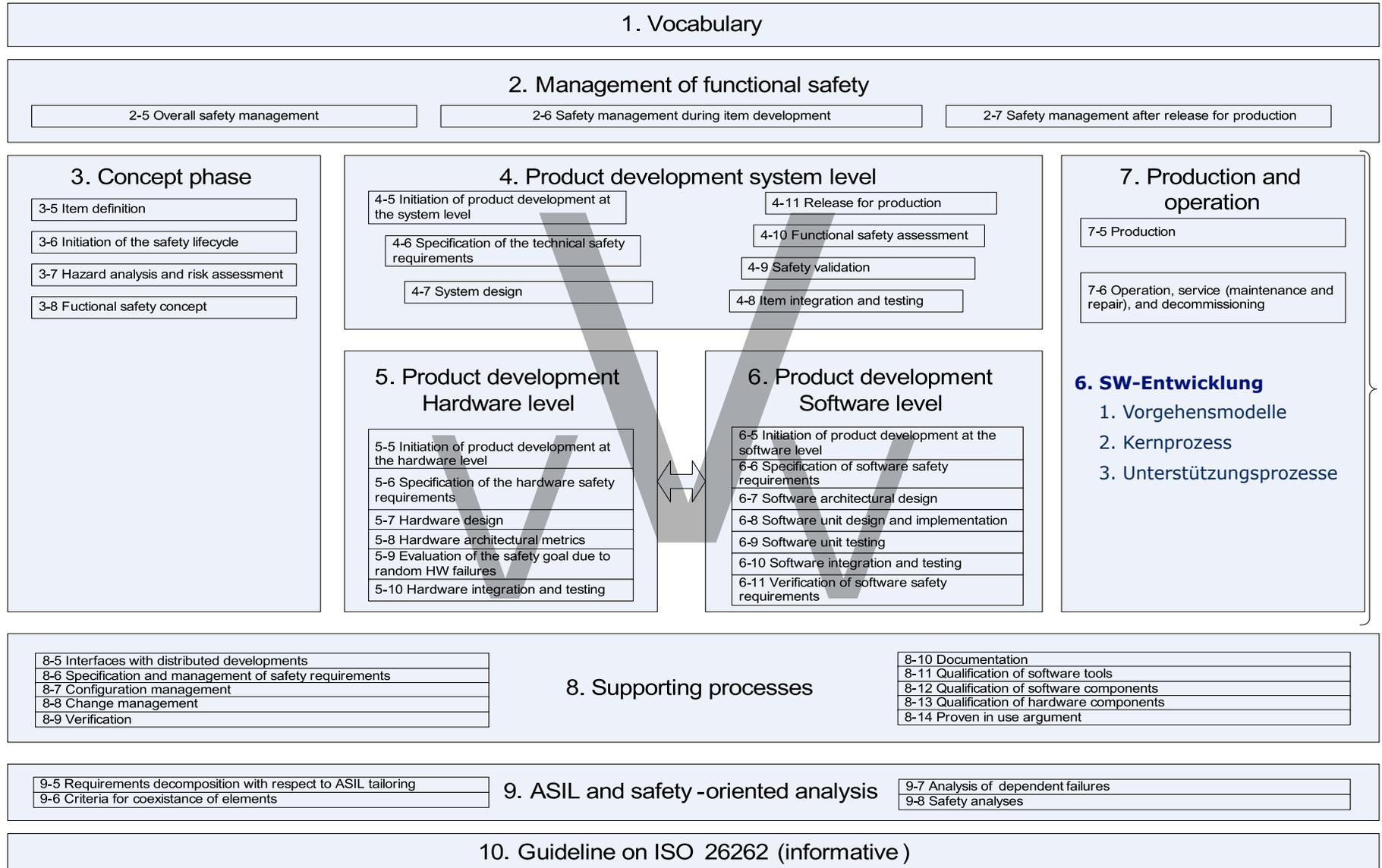
Unterstützungsprozesse

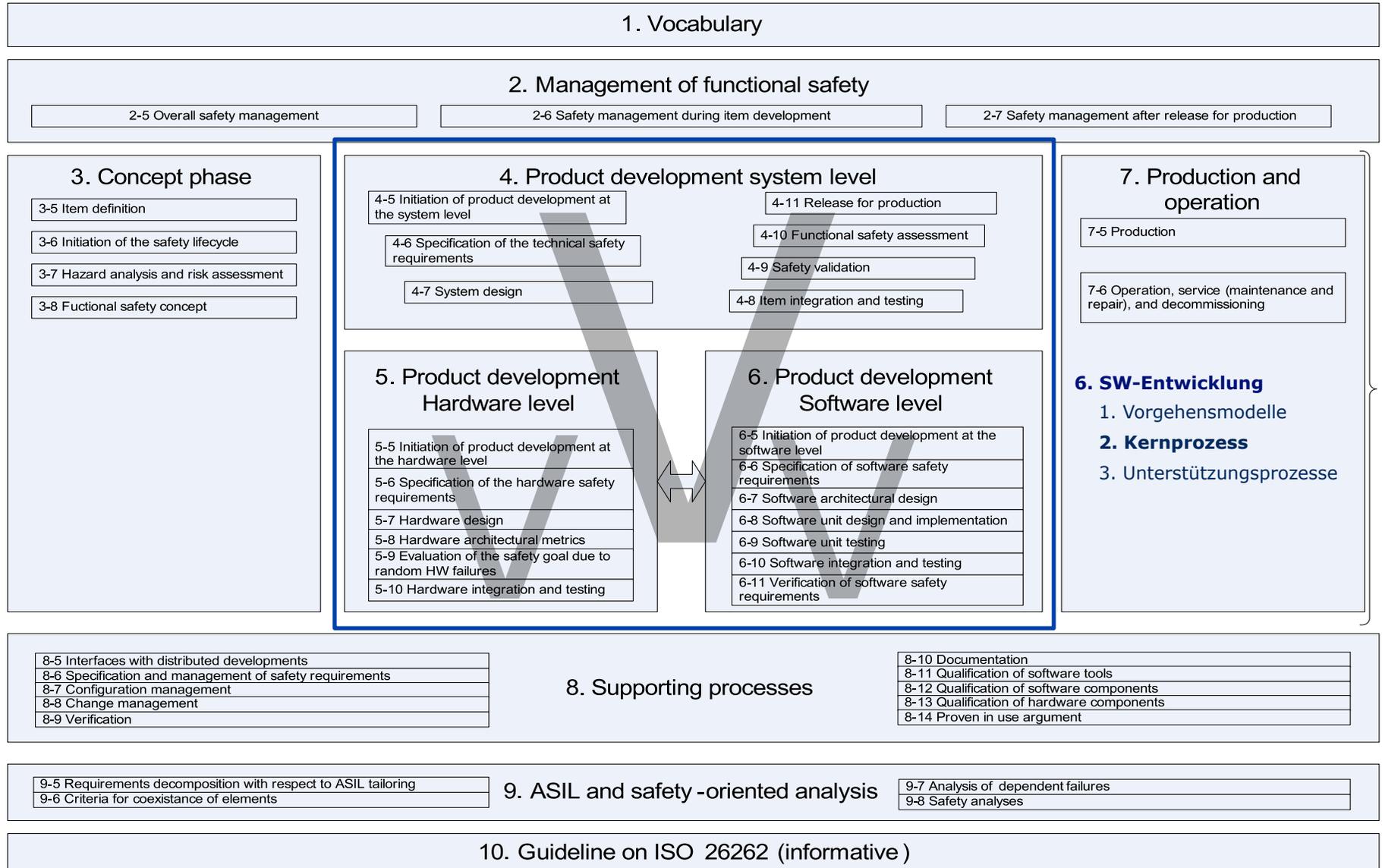
Vorlesung Automotive SWE (Buch Schäuffele/Zurawka)	BMW Group Standard embedded Software (GSeSW)
Unterstützungsprozesse	Supporting processes
Konfigurationsmanagement	Configuration Management
Qualitätssicherung	Quality Assurance
Teil des Projektmanagements	Problem-solving and Change Management

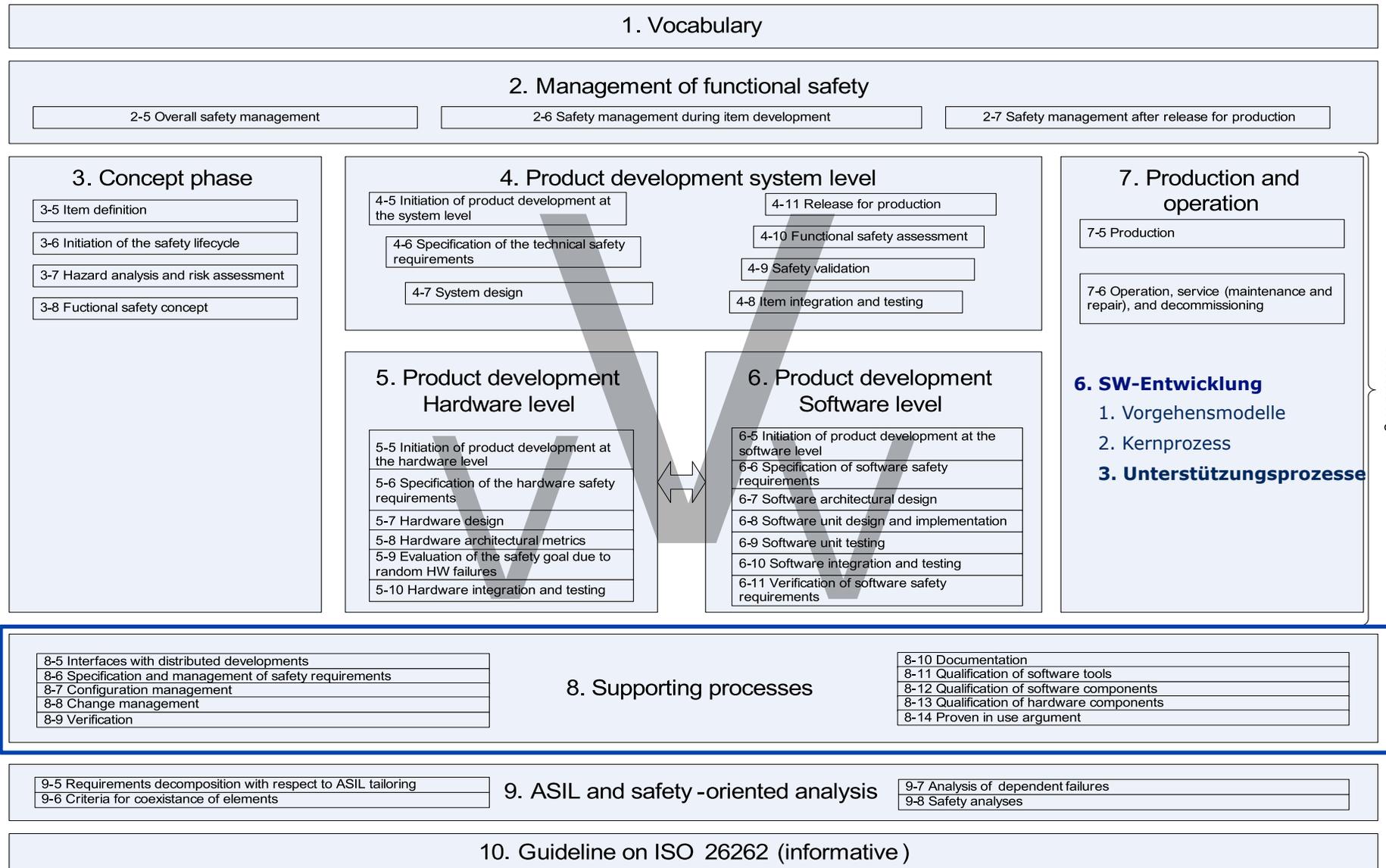












ISO 26262-8 Supporting processes

1. Interfaces within distributed developments
2. Specification and management of safety requirements
3. Configuration management
4. Change management
5. Verification clause
6. Documentation
7. Confidence in the use of software tools
8. Qualification of software components
9. Qualification of hardware components
10. Proven in use argument

6. SW-Entwicklung / 3. Unterstützungsprozesse

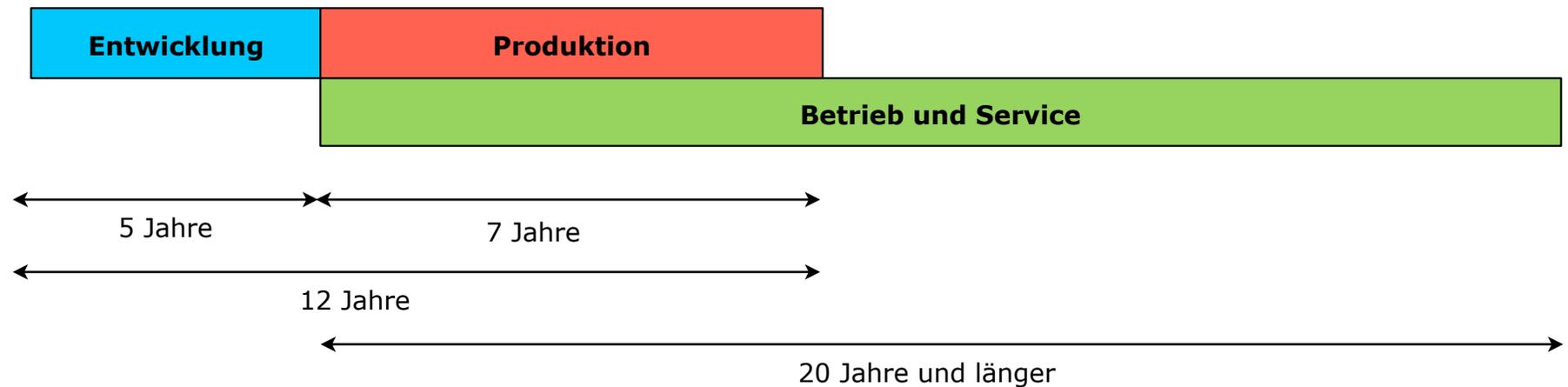
Unterstützungsprozesse für die Embedded Software Entwicklung

1. Vorgehensmodelle und Standards
2. **Konfigurationsmanagement**
 1. **Produkt und Lebenszyklus**
 2. Varianten und Skalierbarkeit
 3. Versionen und Konfigurationen
3. Projektmanagement
4. Lieferantenmanagement
5. Anforderungsmanagement
6. Qualitätssicherung

Typischer Produktlebenszyklus eines Fahrzeugs

Phasen

- Entwicklung
- Produktion
- Betrieb und Service



Unterschiedliche Anforderungen an Schnittstellen

- Unterschiedliche Lebenszyklen der Komponenten
 - Fahrzeug 10 - 20 Jahre
 - Steuergeräte 2 Jahre
 - Unterhaltungselektronik 6 Monate
 - Unterschiedliche Anforderungen an Schnittstellen des Steuergerätes in Entwicklung, Produktion, Betrieb und Service



- Messen
- Kalibrieren
- Rapid Prototyping
- Download und Debugging
- Software-Update durch Flashprogrammierung
- Diagnose

- Messen
- Software-Parametrierung
- Software-Konfiguration
- Software-Update durch Flashprogrammierung
- Diagnose

6. SW-Entwicklung / 3. Unterstützungsprozesse

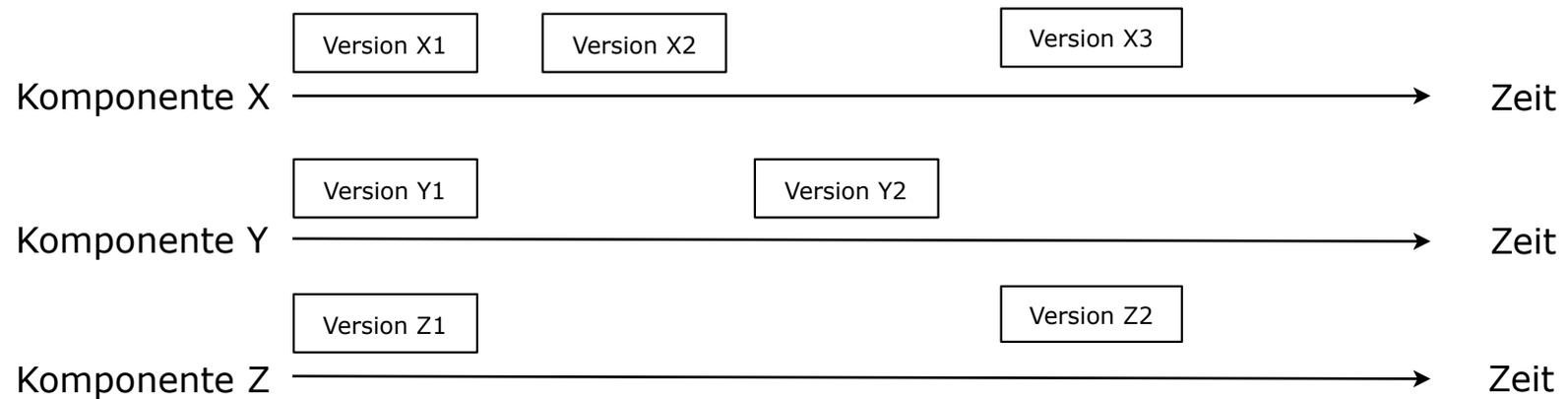
Unterstützungsprozesse für die Embedded Software Entwicklung

1. Vorgehensmodelle und Standards
2. **Konfigurationsmanagement**
 1. Produkt und Lebenszyklus
 2. **Varianten und Skalierbarkeit**
 3. Versionen und Konfigurationen
3. Projektmanagement
4. Lieferantenmanagement
5. Anforderungsmanagement
6. Qualitätssicherung

Versionen

- Bestehende elektronische Systeme werden weiterentwickelt
- Zusätzliche elektronische Systeme werden eingeführt
- Komponentenebene
 - Zu bestimmten Zeitpunkten verschiedene Versionen
- Systemebene
 - Zusätzlich Verwaltung der Beziehungen (Referenzen) zu den enthaltenen Komponenten
- Version:
 - Verwaltung der Komponente an sich
 - Änderung der Komponente (Zeit)

Versionen

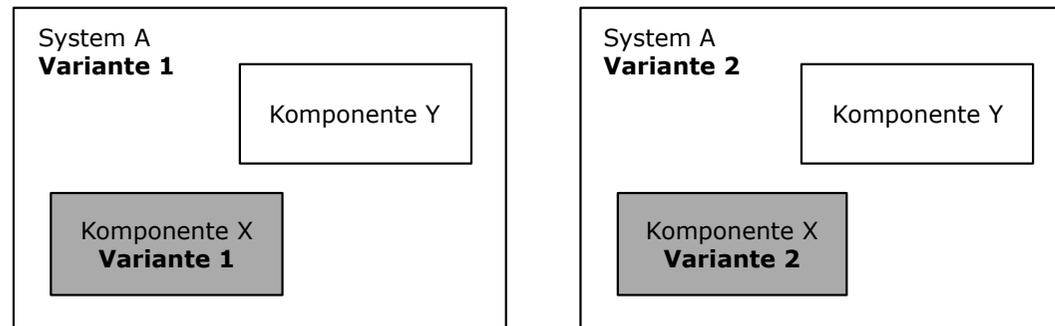


Verschiedene Versionen auf Komponentenebene

Varianten und Skalierbarkeit

Variantenbildung durch Komponenten

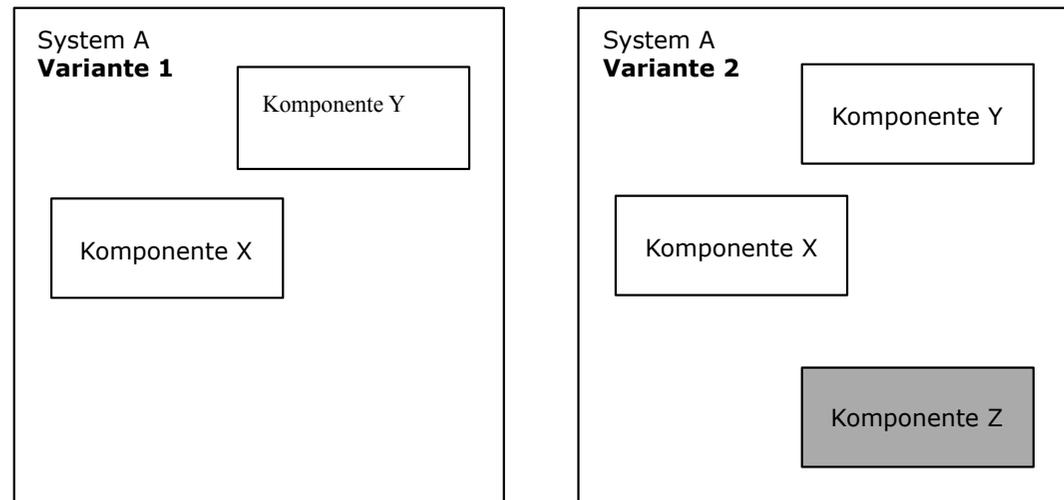
- Zunehmende Anzahl von Fahrzeugvarianten
- Gestiegene Kundenerwartungen
 - **Individualisierung**
 - Erweiterbarkeit
- Beispiel
 - Variante 1:
Automatikgetriebe
 - Variante 2:
Schaltgetriebe



Bildung von Systemvarianten durch Komponentenvarianten

Varianten und Skalierbarkeit

- Variantenbildung durch Skalierung
- Zunehmende Anzahl von Fahrzeugvarianten
- Gestiegene Kundenerwartungen
 - Individualisierung
 - **Erweiterbarkeit**
- Beispiel
 - Komponente Z: Schiebedach



Bildung von Systemvarianten durch Skalierung

Baureihen, Fahrzeugvarianten, Karosserievarianten



1 Baureihe: 3er MBW

4 Fahrzeugvarianten: Limousine, Coupé, Kombi, Cabrio

7 Karosserievarianten: Mit und ohne Schiebedach

Karosserie: Varianten und Gleichteile

	Front	Fahrgastzelle mit Schiebedach	Fahrgastzelle ohne Schiebedach	Heck
Limousine mit Schiebedach	Variante L	Variante LmS	entfällt	Variante L
Limousine ohne Schiebedach	Variante L	entfällt	Variante LoS	Variante L
Coupé mit Schiebedach	Variante L	Variante CmS	entfällt	Variante L
Coupé ohne Schiebedach	Variante L	entfällt	Variante CoS	Variante L
Kombi mit Schiebedach	Variante L	Variante LmS	entfällt	Variante K
Kombi ohne Schiebedach	Variante L	entfällt	Variante LoS	Variante K
Cabrio	Variante L	entfällt	Variante Cabrio	Variante Cabrio
7 Karosserievarianten	1 Frontvariante	2 Fahrgastzellenvarianten	3 Fahrgastzellenvarianten	3 Heckvarianten

Karosserie: Varianten und Gleichteile

- Unterschiedliche Karosserievarianten benötigen unterschiedliche Karosseriefunktionen
- Realisierung der Karosseriefunktionen durch
 - Bedienelemente / Sollwertgeber
 - Aktuatoren
 - Sensoren
 - Steuergeräte
 - Software
- Folge: Varianten und Gleichteile auf den Ebenen
 - Bedienelemente / Sollwertgeber
 - Aktuatoren
 - Sensoren
 - Steuergeräte
 - Software

Karosseriefunktionen: Varianten und Gleichteile

Steuerung von 12 Funktionen	Limousine mit Schiebedach	Limousine ohne Schiebedach	Coupé mit Schiebedach	Coupé ohne Schiebedach	Kombi mit Schiebedach	Kombi ohne Schiebedach	Cabrio	
Türschliessen vorne F	Variante L	Variante L	Variante L	Variante L	Variante L	Variante L	Variante L	1
Türschliessen vorne BF	Variante L	Variante L	Variante L	Variante L	Variante L	Variante L	Variante L	1
Türschliessen hinten F	Variante L	Variante L	entfällt	entfällt	Variante L	Variante L	entfällt	1
Türschliessen hinten BF	Variante L	Variante L	entfällt	entfällt	Variante L	Variante L	entfällt	1
Fensterheber vorne F	Variante L	Variante L	Variante Coupé	Variante Coupé	Variante L	Variante L	Variante Cabrio	3
Fensterheber vorne BF	Variante L	Variante L	Variante Coupé	Variante Coupé	Variante L	Variante L	Variante Cabrio	3
Fensterheber hinten F	Variante L	Variante L	Variante Coupé	Variante Coupé	Variante L	Variante L	Variante Cabrio	3
Fensterheber hinten BF	Variante L	Variante L	Variante Coupé	Variante Coupé	Variante L	Variante L	Variante Cabrio	3
Schiebedach	Variante L	entfällt	Variante L	entfällt	Variante L	entfällt	entfällt	1
Beleuchtung Innenraum	Variante L	Variante L	Variante Coupé	Variante Coupé	Variante K	Variante K	Variante Cabrio	4
Heckklappe	Variante L	Variante L	Variante L	Variante L	Variante K	Variante K	Variante Cabrio	3
Verdeck	entfällt	entfällt	entfällt	entfällt	entfällt	entfällt	Variante Cabrio	1

Motorvarianten C-Klasse Limousine

Modell	Zylinder- anordnung/ -anzahl	Hubraum (cm ³)	Nenn- leistung (kW bei 1/min)[1]	Höchst- geschwind- igkeit (km/h)	Kraftstoff- verbrauch kombiniert (l/100 km)[2]	CO ₂ - Emissionen kombiniert (g/km)[2]
C 180 CDI BlueEFFICIENCY	R4	2.143	88/2.800–4.600 (88/3.000–4.600)	208 (206)	5,3–4,8 (5,3–4,9)	139–125 (140–129)
C 200 CDI BlueEFFICIENCY	R4	2.143	100/2.800–4.600 (100/2.800–4.600)	218 (215)	5,3–4,8 (5,3–4,9)	139–125 (140–129)
C 220 CDI BlueEFFICIENCY	R4	2.143	125/3.000–4.200 (125/3.000–4.200)	232 (231)	5,1–4,4 (5,2–4,8)	133–117 (136–125)
C 220 CDI BlueEFFICIENCY Edition	R4	2.143	125/3.000–4.200 (125/3.000–4.200)	232 (231)	4,1 (4,4)	109 (116)
C 250 CDI BlueEFFICIENCY	R4	2.143	150/4.200 (150/4.200)	240 (240)	5,3–4,8 (5,2–4,8)	140–125 (136–125)
C 250 CDI 4MATIC BlueEFFICIENCY	R4	2.143	– (150/4.200)	– (240)	– (5,7–5,4)	– (152–144)
C 300 CDI 4MATIC BlueEFFICIENCY	V6	2.987	– (170/3.800)	– (250)[3]	– (7,2–7,0)	– (189–185)
C 350 CDI BlueEFFICIENCY	V6	2.987	– (195/3.800)	– (250)[3]	– (6,0–5,9)	– (157–154)

- http://www.mercedes-benz.de/content/germany/mpc/mpc_germany_website/de/home_mpc/passengercars/home/new_cars/models/c-class/_w204/facts_/drivetrain/dieselenqines.html

Motorvarianten C-Klasse Limousine

Modell	Zylinder- anordnung/ -anzahl	Hubraum (cm ³)	Nenn- leistung (kW bei 1/min)[1]	Höchst- geschwind- igkeit (km/h)	Kraftstoff- verbrauch kombiniert (l/100 km)[2]	CO ₂ - Emissionen kombiniert (g/km)[2]
C 180 CDI BlueEFFICIENCY	R4	2.143	88/2.800–4.600 (88/3.000–4.600)	208 (206)	5,3–4,8 (5,3–4,9)	139–125 (140–129)
C 200 CDI BlueEFFICIENCY	R4	2.143	100/2.800–4.600 (100/2.800–4.600)	218 (215)	5,3–4,8 (5,3–4,9)	139–125 (140–129)
C 220 CDI BlueEFFICIENCY	R4	2.143	125/3.000–4.200 (125/3.000–4.200)	232 (231)	5,1–4,4 (5,2–4,8)	133–117 (136–125)
C 220 CDI BlueEFFICIENCY Edition	R4	2.143	125/3.000–4.200 (125/3.000–4.200)	232 (231)	4,1 (4,4)	109 (116)
C 250 CDI BlueEFFICIENCY	R4	2.143	150/4.200 (150/4.200)	240 (240)	5,3–4,8 (5,2–4,8)	140–125 (136–125)
C 250 CDI 4MATIC BlueEFFICIENCY	R4	2.143	– (150/4.200)	– (240)	– (5,7–5,4)	– (152–144)
C 300 CDI 4MATIC BlueEFFICIENCY	V6	2.987	– (170/3.800)	– (250)[3]	– (7,2–7,0)	– (189–185)
C 350 CDI BlueEFFICIENCY	V6	2.987	– (195/3.800)	– (250)[3]	– (6,0–5,9)	– (157–154)

- http://www.mercedes-benz.de/content/germany/mpc/mpc_germany_website/de/home_mpc/passengercars/home/new_cars/models/c-class/_w204/facts_/drivetrain/dieselenqines.html

Softwarevarianten und Fahrzeugvarianten

- Karosserieformen und Karosseriefunktionen
 - Softwarevarianten folgen Fahrzeugvarianten

- Motorvarianten C-Klasse
 - Softwarevarianten ermöglichen Fahrzeugvarianten



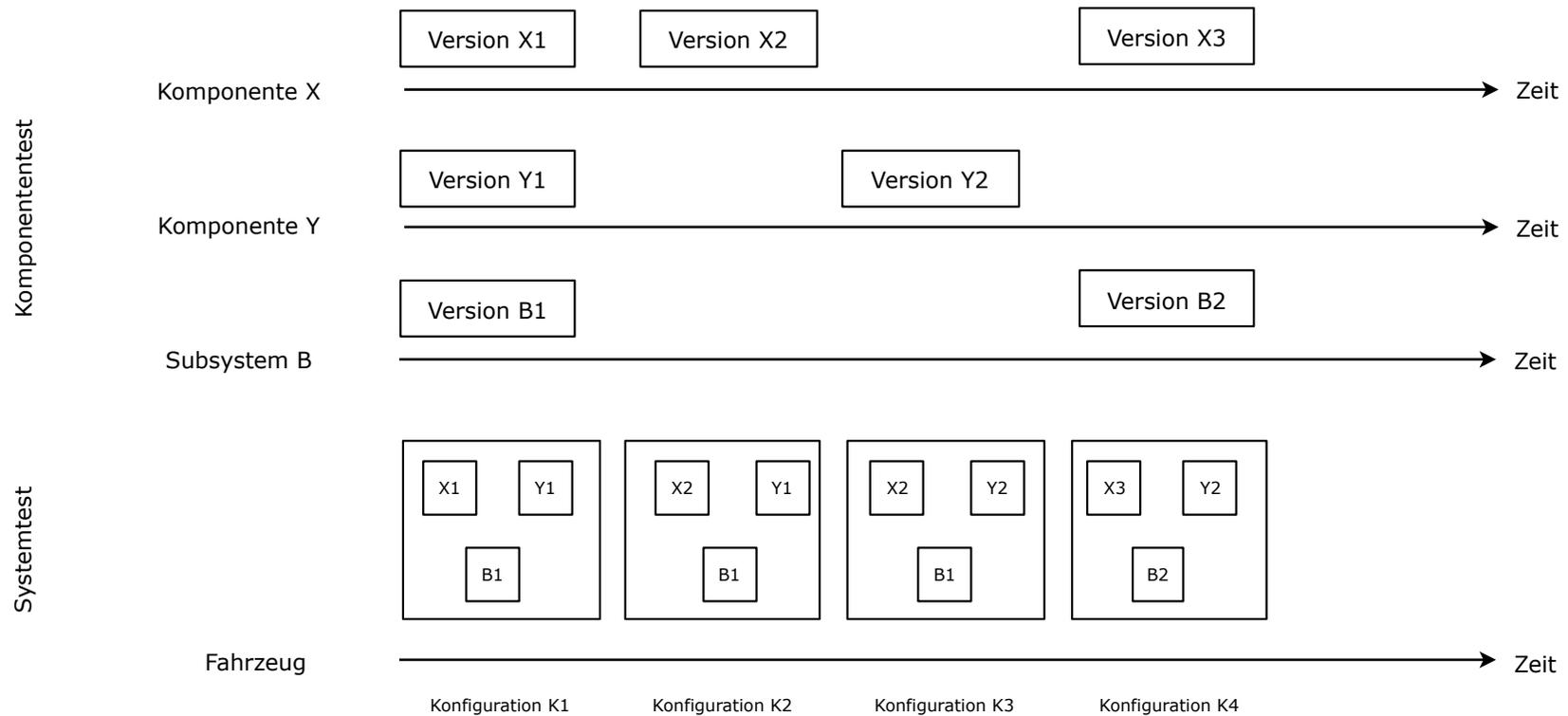
6. SW-Entwicklung / 3. Unterstützungsprozesse

Unterstützungsprozesse für die Embedded Software Entwicklung

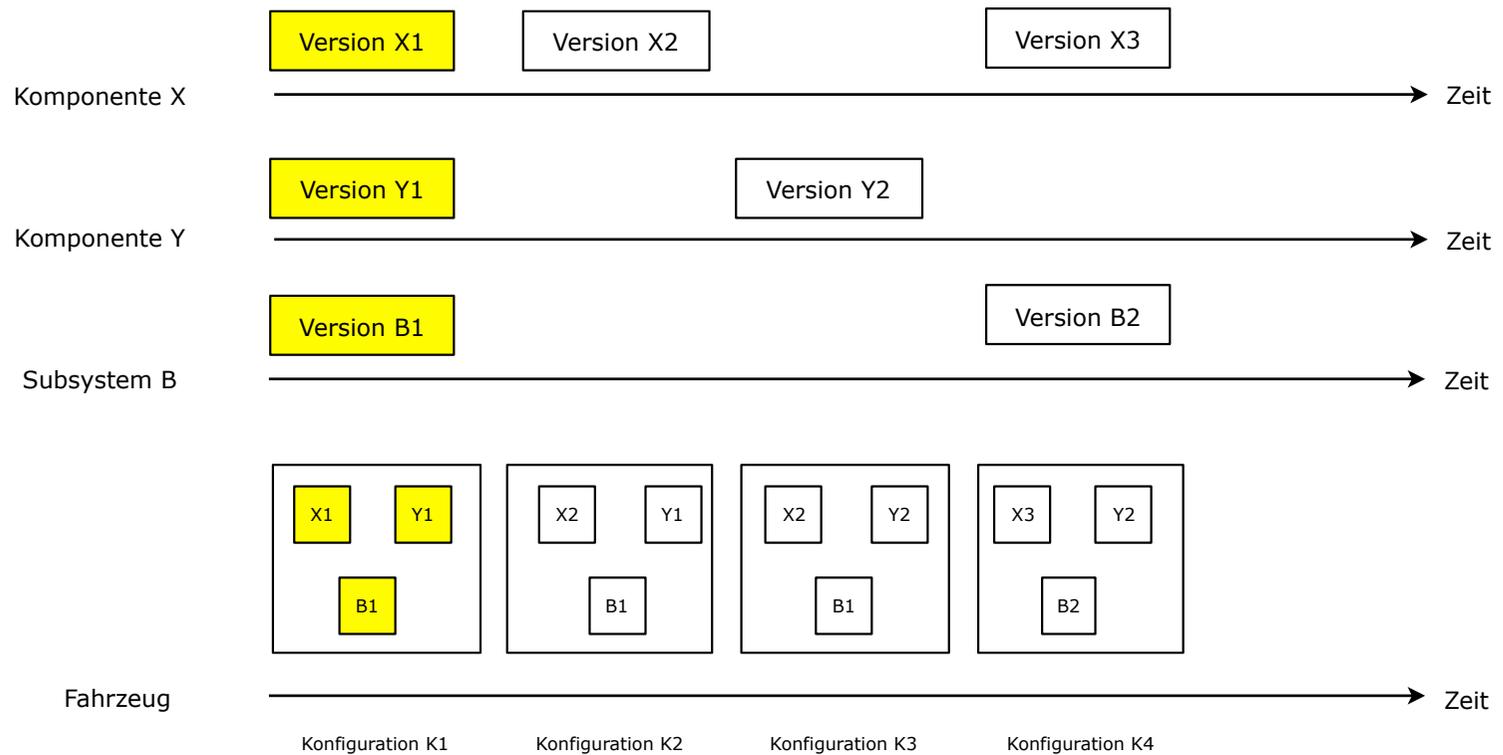
1. Vorgehensmodelle und Standards
2. **Konfigurationsmanagement**
 1. Produkt und Lebenszyklus
 2. Varianten und Skalierbarkeit
 3. **Versionen und Konfigurationen**
3. Projektmanagement
4. Lieferantenmanagement
5. Anforderungsmanagement
6. Qualitätssicherung

Integration der System-Komponenten

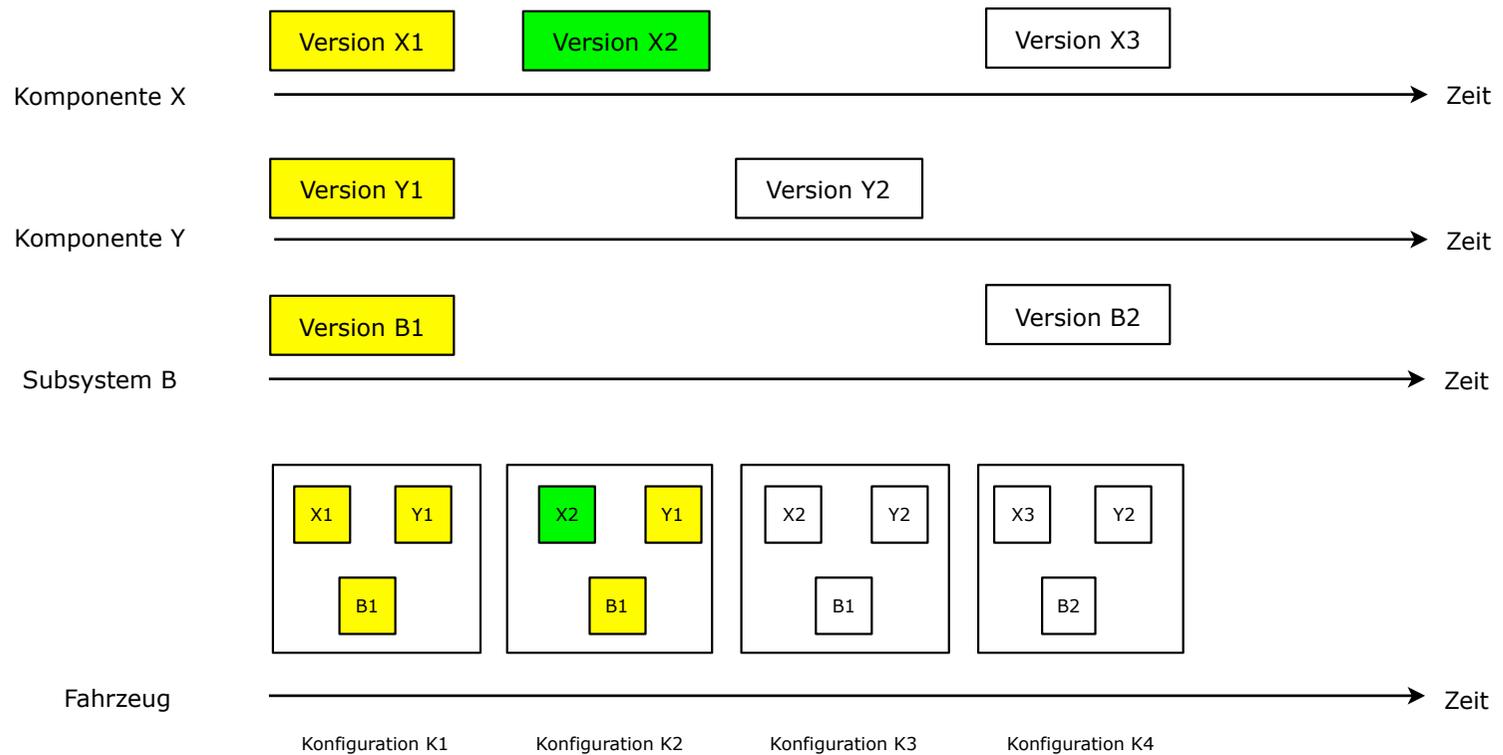
Abhängigkeiten zwischen Komponenten- und Systemtest



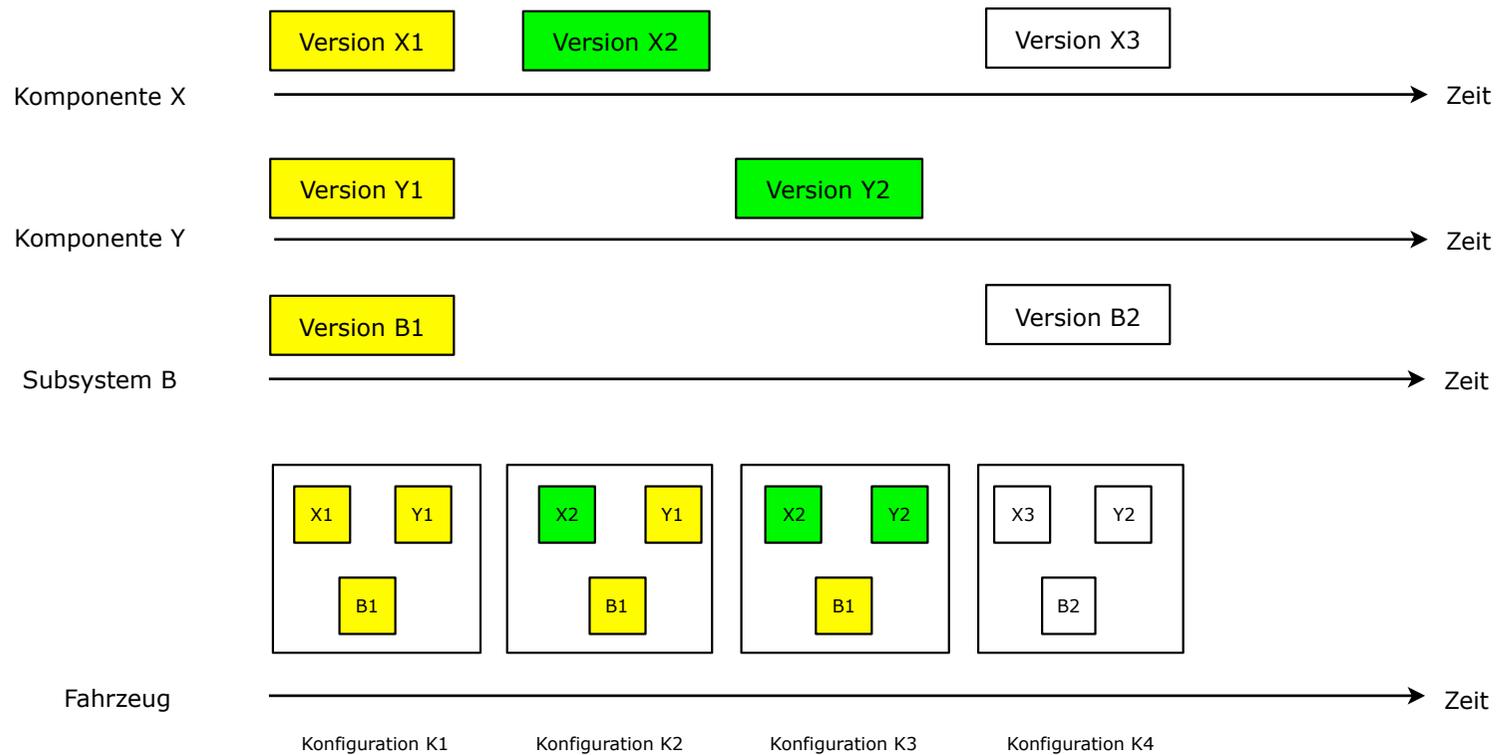
Integration der System-Komponenten



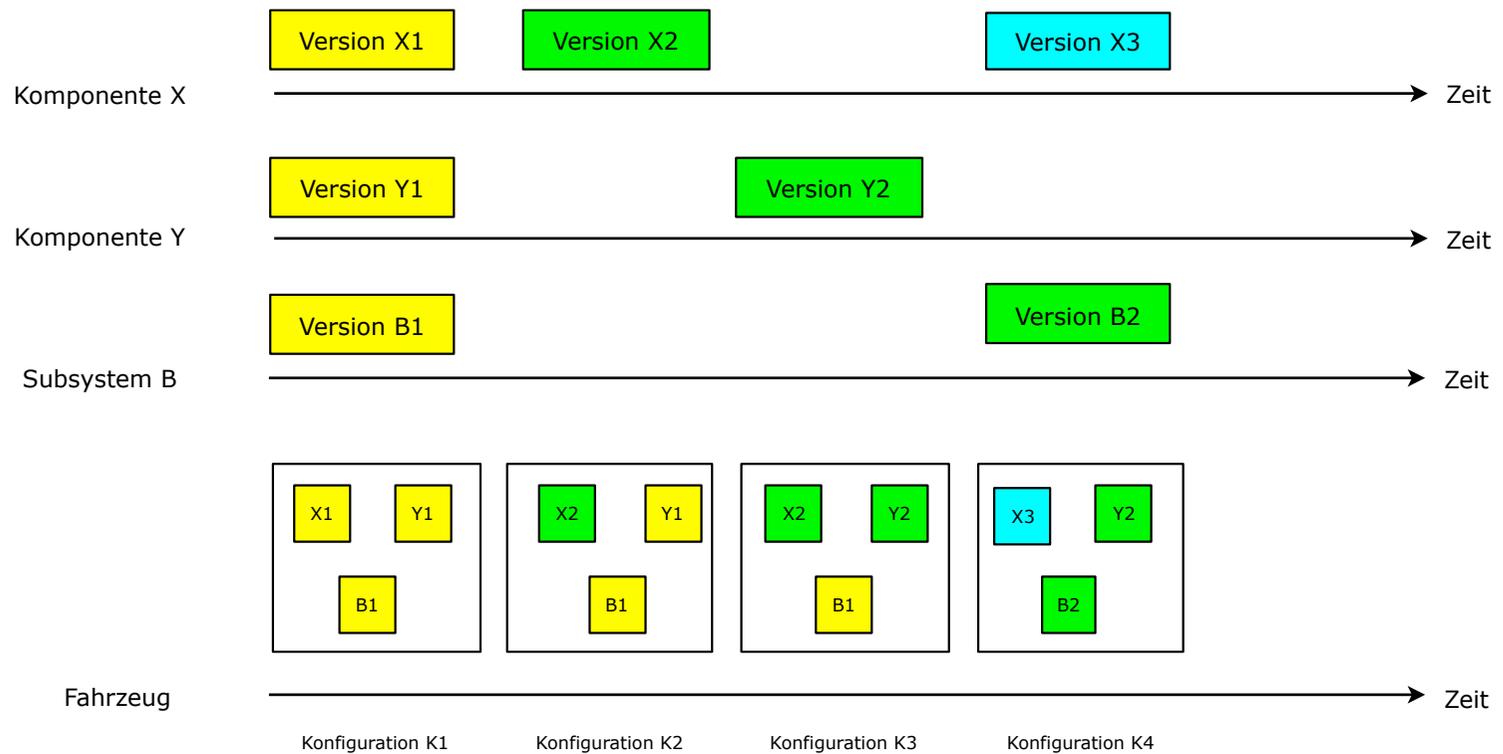
Integration der System-Komponenten



Integration der System-Komponenten

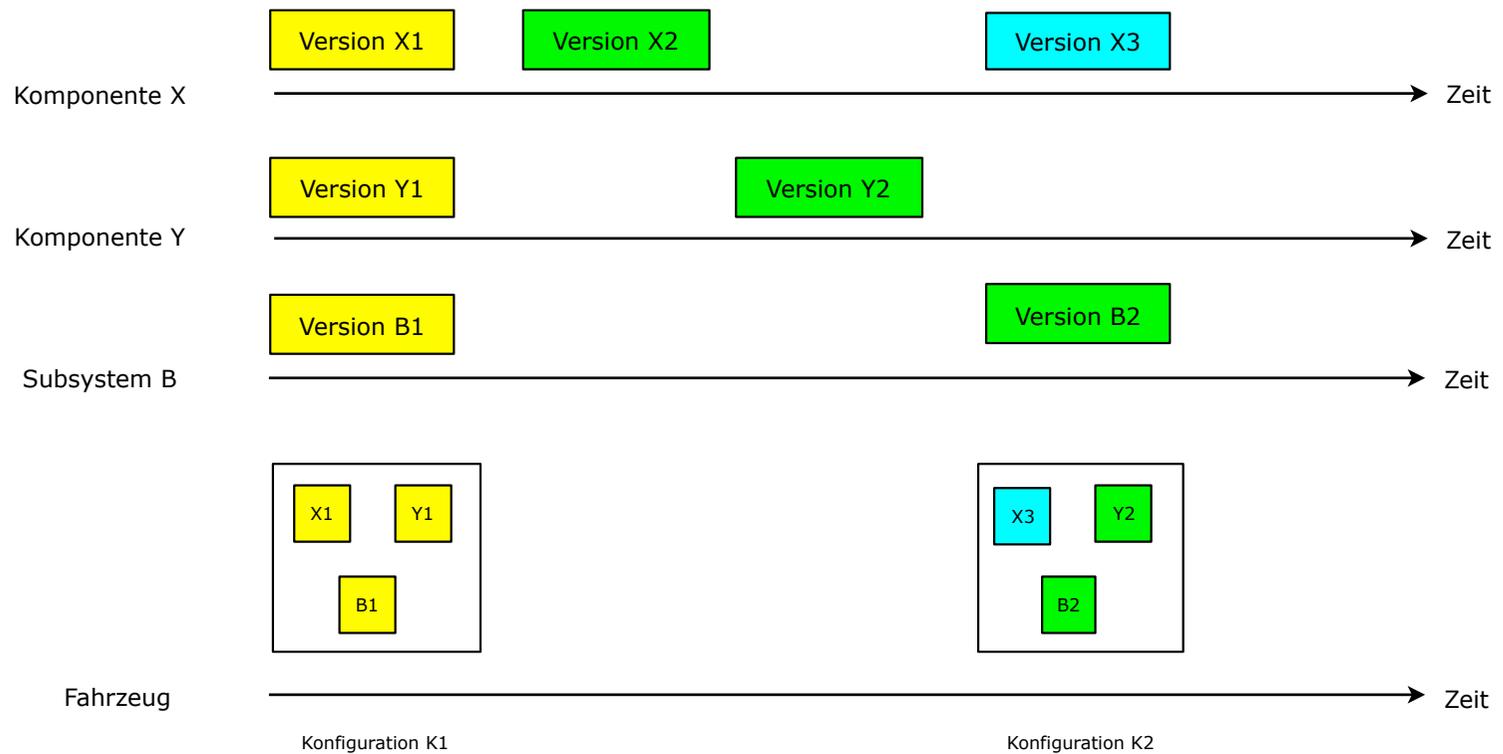


Integration der System-Komponenten



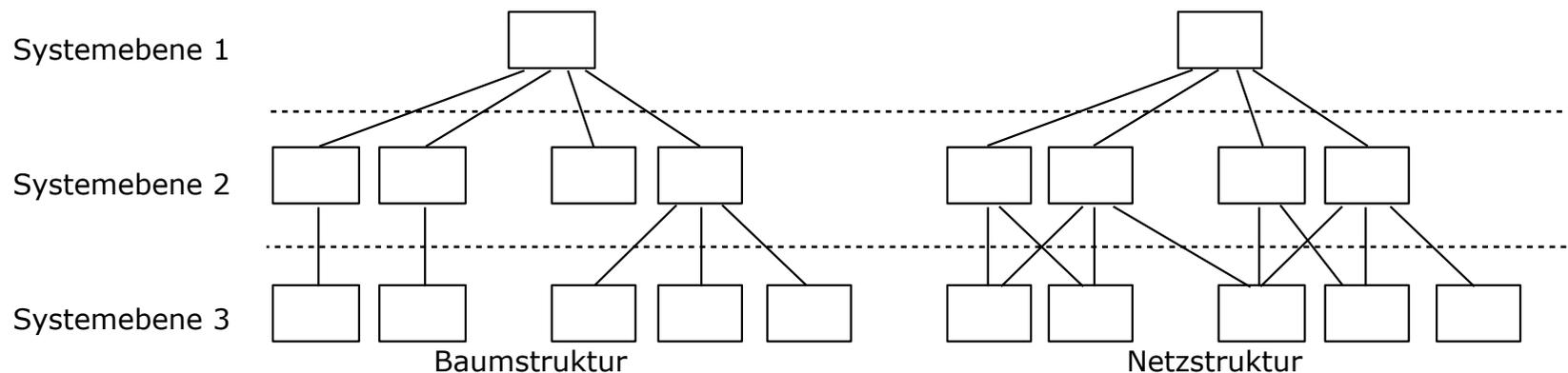
Integration der System-Komponenten

In der Praxis meist Bündelung: Weniger Konfigurationen



Versionen und Konfigurationen

- Systemvarianten können auf allen Systemebenen auftreten
- Hierarchiebeziehungen
 - Baumstrukturen
 - Jede Komponente gehört zu genau einem System
 - Netzstrukturen
 - Eine Komponente kann zu mehreren Systemen gehören
- Das Versions- und Konfigurationsmanagement (Verwaltung von Versionen und Konfigurationen) geht von Netzstrukturen aus
 - Warum?

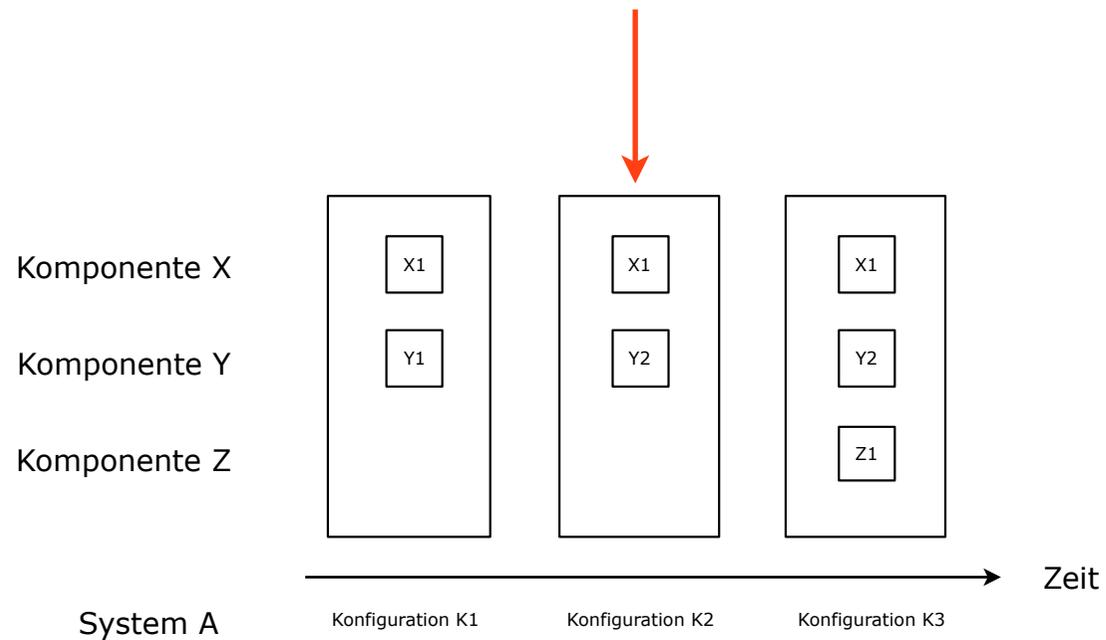


Konfiguration

- Eine Konfiguration ist eine versionierbare Komponente, die eine Menge anderer versionierbarer Komponenten referenziert.
- Verwaltung der Beziehungen zu den enthaltenen Komponentenversionen
- Referenzierte Versionen dürfen nicht mehr geändert werden :-)
 - ... müssen über die Lebenszeit der Konfiguration verfügbar sein

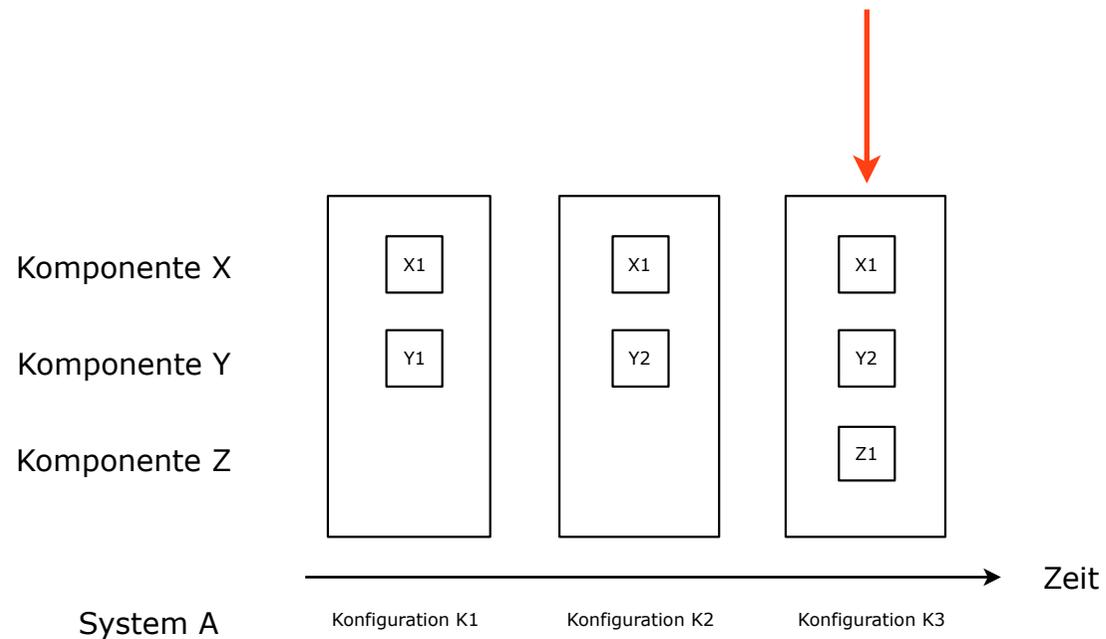
Konfiguration

- Versionen einer Konfiguration
 - **Versionen der Komponenten**
 - Änderung der Hierarchiebeziehungen



Konfiguration

- Versionen einer Konfiguration
 - Versionen der Komponenten
 - **Änderung der Hierarchiebeziehungen**



Konfigurationsmanagement

- Eigentlich: Versions- und Konfigurationsmanagement
- Verwaltet die Beziehungen zwischen Systemen und Komponenten und deren Änderungen
 - Weiterentwicklungen von Komponenten
 - Änderung der Hierarchien von Systemen
- Wichtiger Bestandteil von
 - Entwicklungsprozessen
 - Produktionsprozessen
 - Serviceprozessen
- Das Konfigurationsmanagement verwaltet alle Arbeitsergebnisse der Entwicklung
 - Spezifikationen
 - Implementierungen (Programmstände, Datenstände)
 - Beschreibungsdateien für Diagnose, Software-Update, Software-Parametrierung
 - Dokumentation
 - ...

6. SW-Entwicklung / 3. Unterstützungsprozesse

Unterstützungsprozesse für die Embedded Software Entwicklung

1. Vorgehensmodelle und Standards
2. Konfigurationsmanagement
3. **Projektmanagement**
 1. **Projektplanung**
 2. Projektverfolgung und Risikomanagement
4. Lieferantenmanagement
5. Anforderungsmanagement
6. Qualitätssicherung

Projekte

- Als Projekte werden Aufgabenstellungen bezeichnet, die durch folgende Merkmale gekennzeichnet sind
 - Aufgabenstellung mit Risiko und einer gewissen Einmaligkeit - keine Routineaufgaben
 - Eindeutige Aufgabenstellung
 - Verantwortung und Zielsetzung für ein zu lieferndes Gesamtergebnis
 - Zeitliche Befristung mit klarem Anfangs- und Endtermin
 - Begrenzter Ressourceneinsatz
 - Besondere auf das Projekt abgestimmte Organisation
 - Häufig: Teilaufgaben und Organisationseinheiten
 - Verschiedenartig
 - Untereinander verbunden
 - Wechselseitig voneinander abhängig

Projektziele

Qualitätsziele

- Welche Anforderungen sollen vom Gesamtergebnis erfüllt werden?

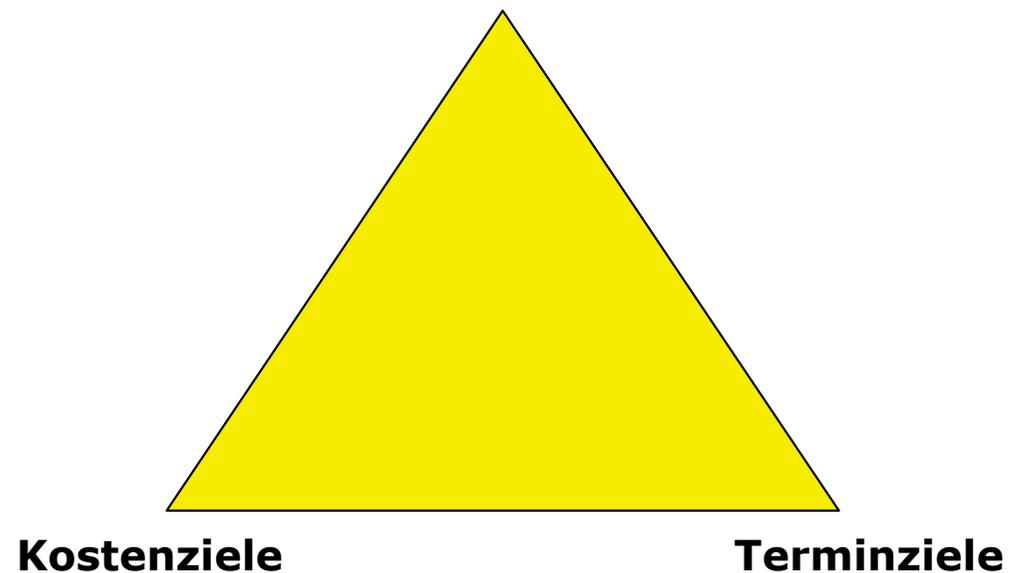
Kostenziele

- Wie viel darf die Erarbeitung des Gesamtergebnisses kosten?

Terminziele

- Wann soll das Gesamtergebnis vorliegen?

Qualitätsziele



Projektmanagement - Projektplanung

- Planung der Umsetzung der Projektziele
 - Qualitätsplanung
 - Kostenplanung
 - Terminplanung
- Organisationsplanung
- Einsatzplanung
- Risikoanalyse

6. SW-Entwicklung / 3. Unterstützungsprozesse

Unterstützungsprozesse für die Embedded Software Entwicklung

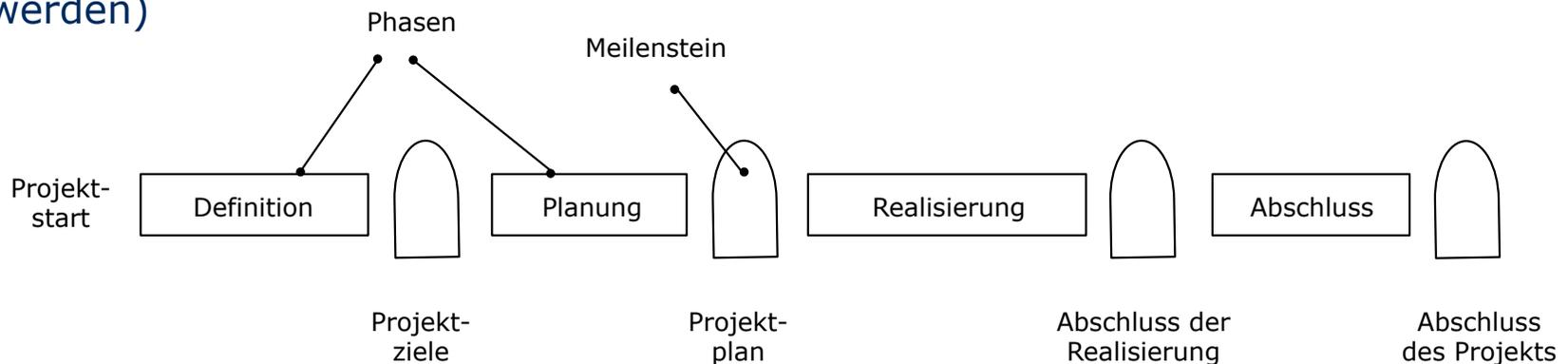
1. Vorgehensmodelle und Standards
2. Konfigurationsmanagement
3. **Projektmanagement**
 1. Projektplanung
 2. **Projektverfolgung und Risikomanagement**
4. Lieferantenmanagement
5. Anforderungsmanagement
6. Qualitätssicherung

Projektmanagement

- Projektverfolgung / Projektsteuerung
- Verfolgung und Überwachung von
 - Qualität
 - Kosten
 - Terminen
- Risikomanagement
 - Beobachtung von auftretenden Risiken und Gegensteuerung

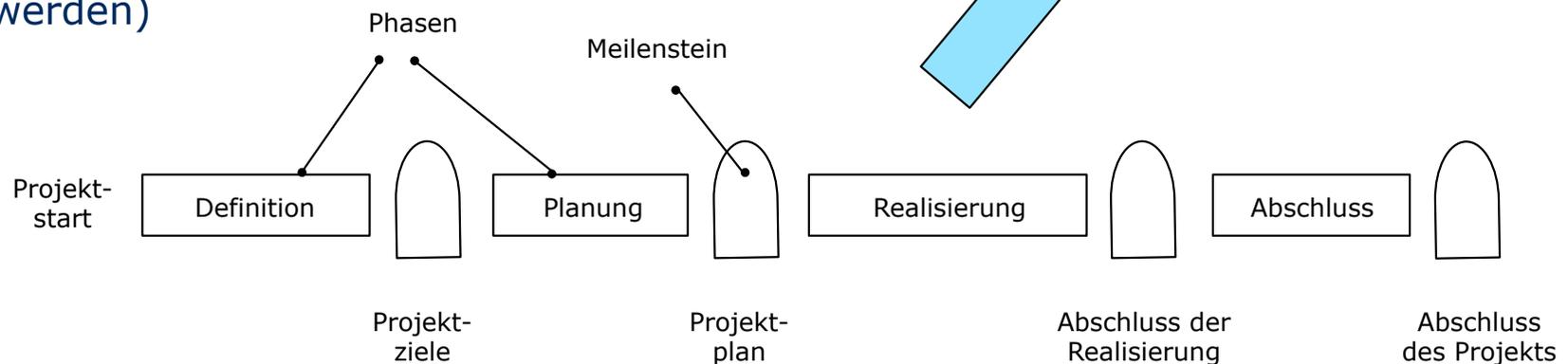
Projektplanung

- Definition der Teilaufgaben eines Projektes
- Meilenstein
 - Abschluss einer Teilaufgabe
 - Termin für
 - Teillieferungen
 - Tests
 - Teilzahlungen
- Projektphase
 - Zeitraum, in dem eine Teilaufgabe bearbeitet wird
 - Im allgemeinen vier Projektphasen (die in der Realität weiter untergliedert werden)



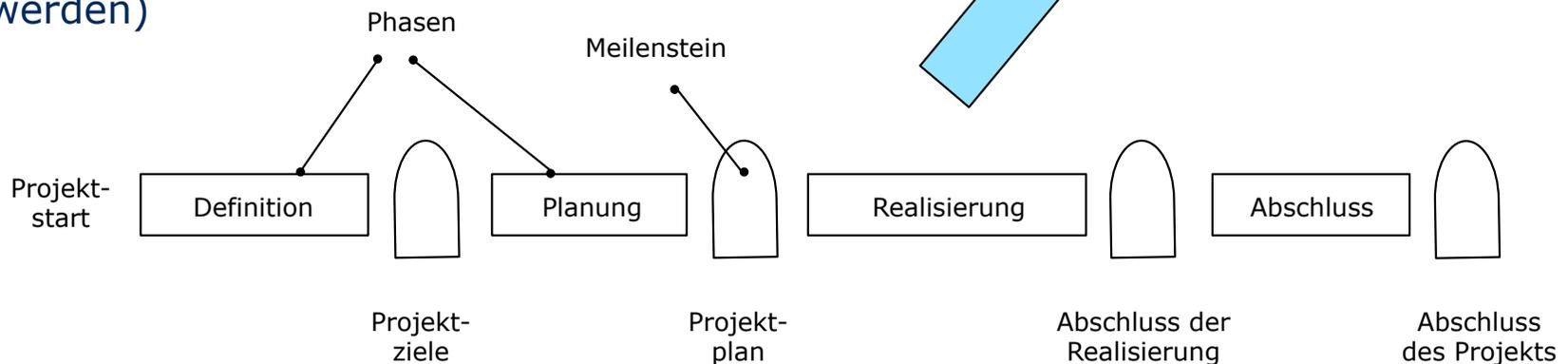
Projektplanung

- Definition der Teilaufgaben eines Projektes
- Meilenstein
 - Abschluss einer Teilaufgabe
 - Termin für
 - Teillieferungen
 - Tests
 - Teilzahlungen
- Projektphase
 - Zeitraum, in dem eine Teilaufgabe bearbeitet wird
 - Im allgemeinen vier Projektphasen (die in der Realität weiter untergliedert werden)



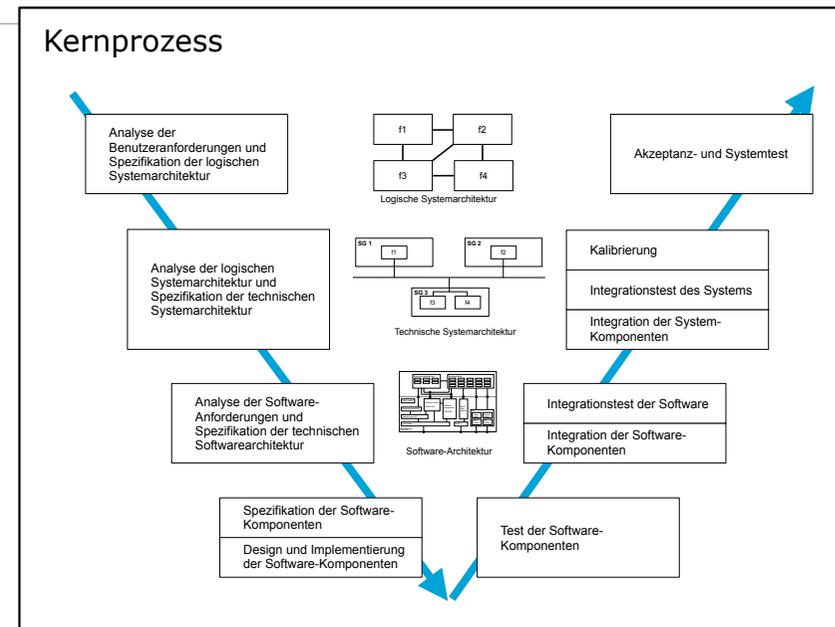
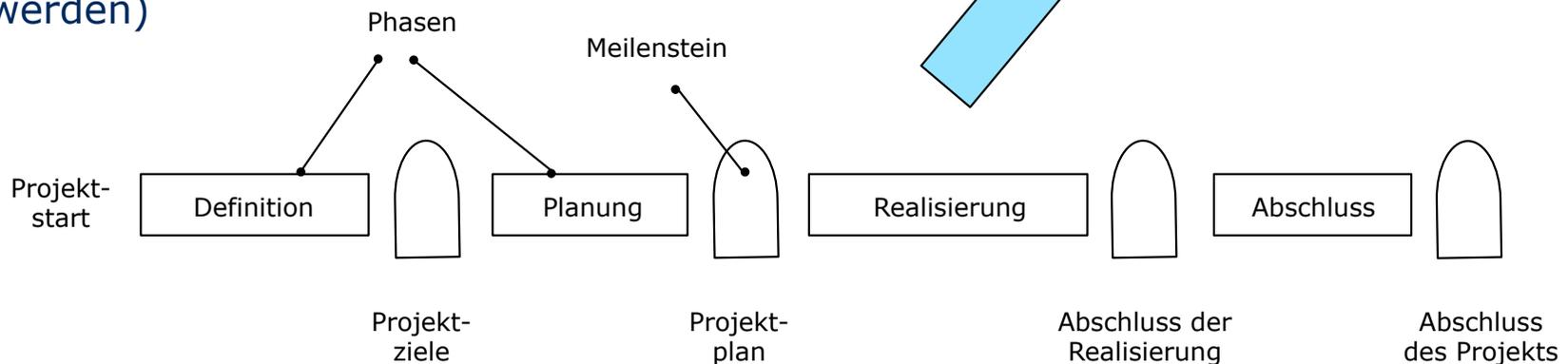
Projektplanung

- Definition der Teilaufgaben eines Projektes
- Meilenstein
 - Abschluss einer Teilaufgabe
 - Termin für
 - Teillieferungen
 - Tests
 - Teilzahlungen
- Projektphase
 - Zeitraum, in dem eine Teilaufgabe bearbeitet wird
 - Im allgemeinen vier Projektphasen (die in der Realität weiter untergliedert werden)



Projektplanung

- Definition der Teilaufgaben eines Projektes
- Meilenstein
 - Abschluss einer Teilaufgabe
 - Termin für
 - Teillieferungen
 - Tests
 - Teilzahlungen
- Projektphase
 - Zeitraum, in dem eine Teilaufgabe bearbeitet wird
 - Im allgemeinen vier Projektphasen (die in der Realität weiter untergliedert werden)



Projektplanung: Qualität und Kosten

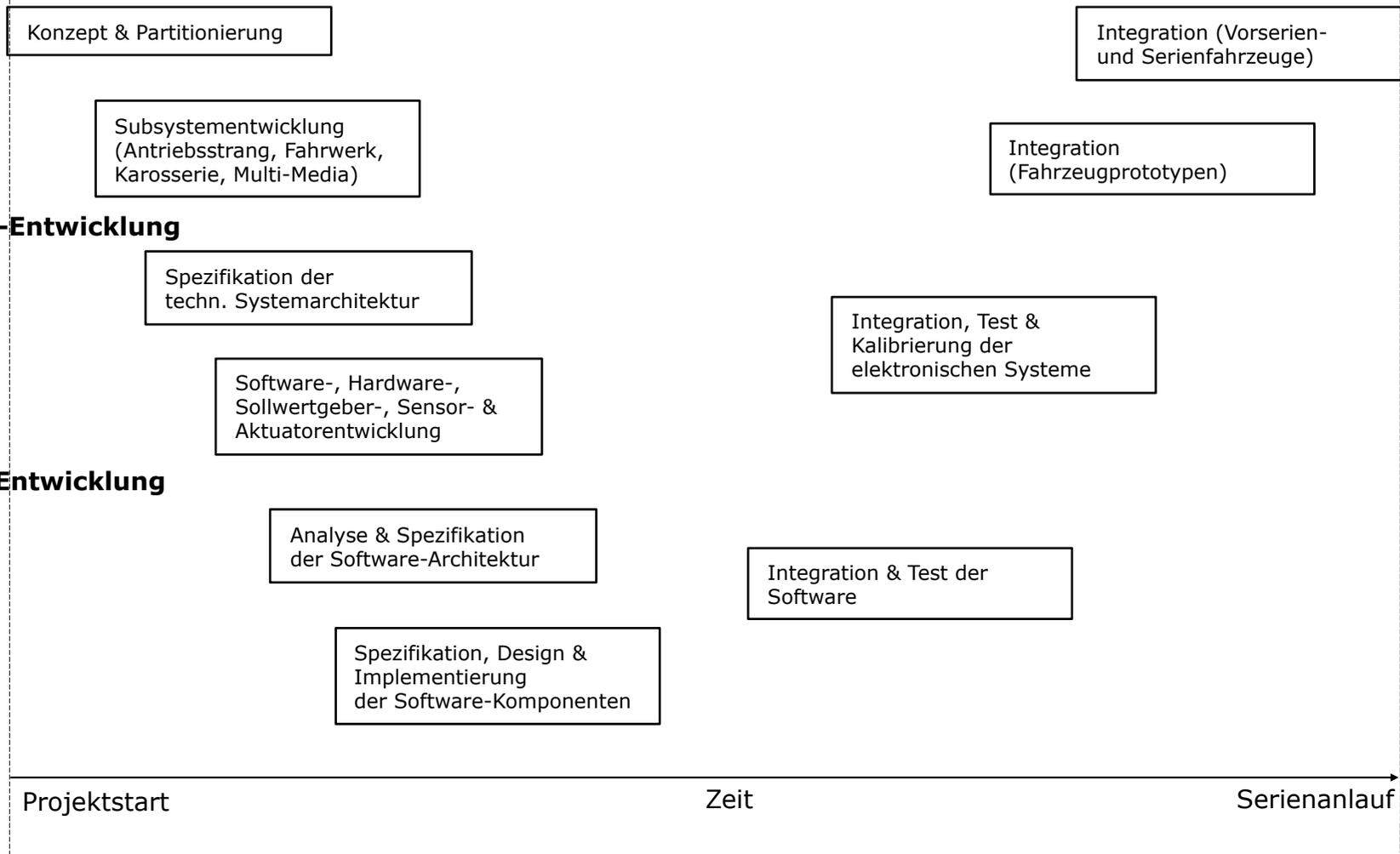
- Qualitätsplanung
 - Festlegung der Massnahmen zur Erreichung des Gesamtergebnisses
 - Für alle Projektphasen
 - Richtlinien zur Qualitätssicherung
 - Massnahmen zur Qualitätsprüfung
 - Phasenübergreifende Zusammenfassung in einem Qualitätsplan
- Kostenplanung
 - Personalkosten
 - Einsatzpläne für Mitarbeiter
 - Sachkosten
 - Materialkosten
 - Raumkosten
 - Reisekosten
 - Massnahmen
 - Z.B. Wiederverwendung von Ergebnissen aus anderen Projekten

Projektplanung: Termine

- Terminplanung
 - Festlegung des Zeitraums für die Durchführung der Projektphasen
 - Anfangstermin
 - Endtermin / Meilenstein
 - Herausforderungen
 - Einsatz von Mitarbeitern in verschiedenen Projekten
 - Gleichzeitige Durchführung mehrerer Projekte
 - Abhängigkeiten zwischen den Projektphasen
 - Abarbeitung
 - Sequentiell bei abhängigen Aufgaben
 - Parallel bei unabhängigen Aufgaben
 - Verkürzung der Entwicklungszeit
 - Simultaneous Engineering
 - Planung und Synchronisation zeitlich paralleler Entwicklungsschritte

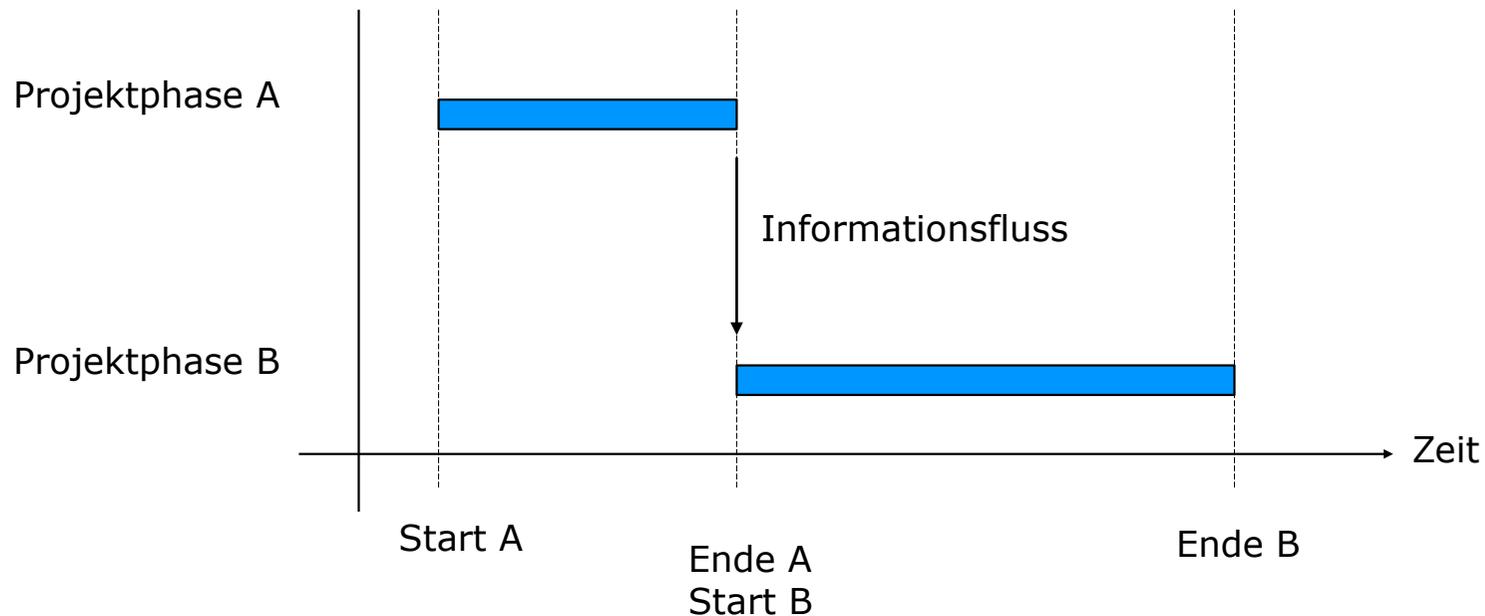
Beispiel: Terminplan für Fahrzeugentwicklung

Fahrzeugentwicklung



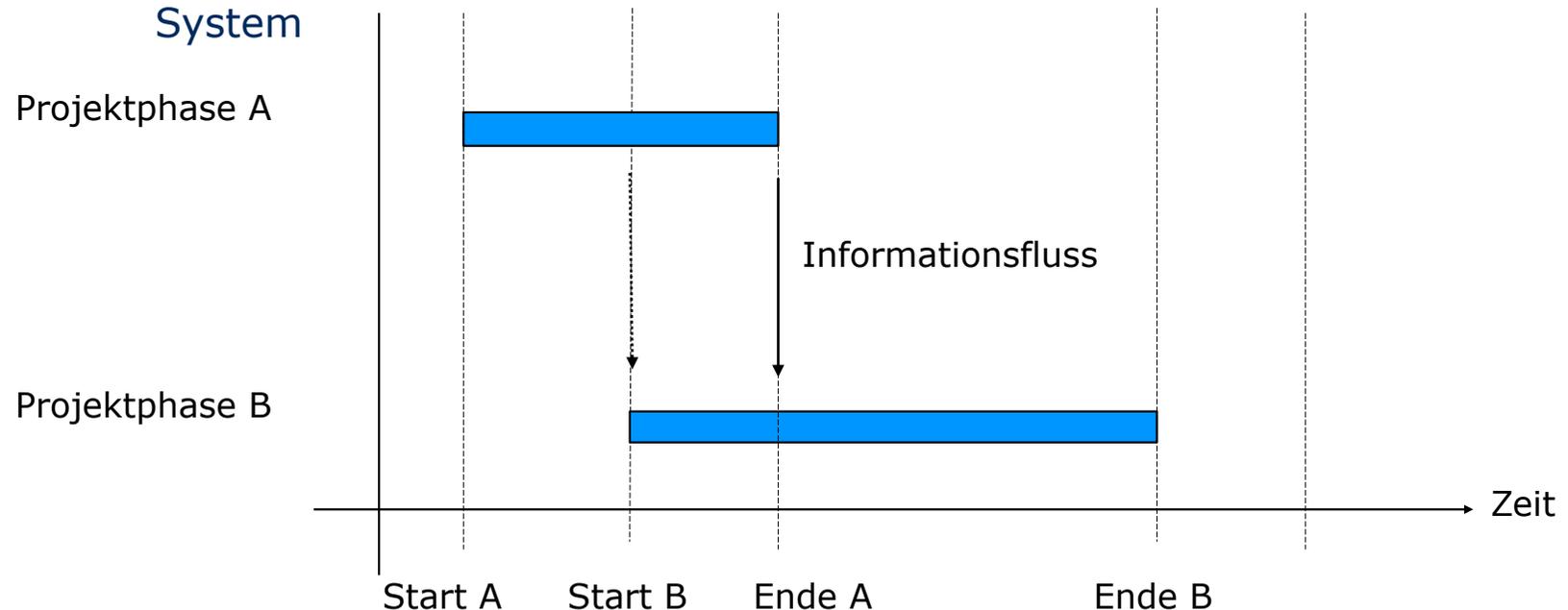
Sequentielle Planung von Projektphasen

- Start von Phase B nach Ende von Phase A
- Vorteile: Sequentieller Informationsfluss ohne Risiko
- Nachteile: Lange Bearbeitungszeit
- Beispiel: Integrationstest (Phase B) nach Integration (Phase A)



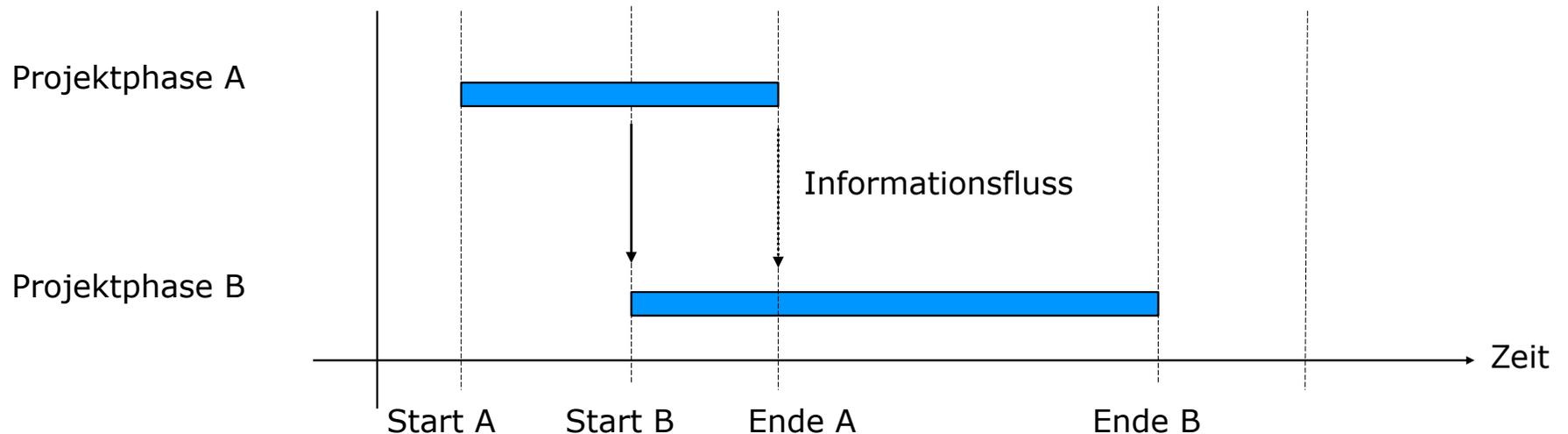
Parallele Planung ohne Einfrieren

- Start von Phase B mit Teilinformationen von Phase A ohne frühes Einfrieren von Entscheidungen in Phase A
- Vorteile: Kürzere Bearbeitungsdauer
- Nachteile: Risiko von Verzögerungen durch Iterationen in der Phase B
- Beispiel: Anwendungssoftware (Phase B) wird auf Rapid Prototyping System vor Fertigstellung der Plattformsoftware (Phase A) entwickelt
Risiko: Plattformsoftware verhält sich anders wie Rapid Prototyping System



Parallele Planung mit Einfrieren

- Start von Phase B mit Teilinformationen von Phase A mit frühem Einfrieren einiger Entscheidungen in Phase A
- Vorteile: Kürzere Bearbeitungsdauer
- Nachteile: Risiko von Qualitätseinbußen durch frühe Einschränkungen in Phase A
- Beispiel: Kalibrierung von implementierten Funktionen der Anwendungssoftware (Phase B) vor Fertigstellung aller Funktionen (Phase A)

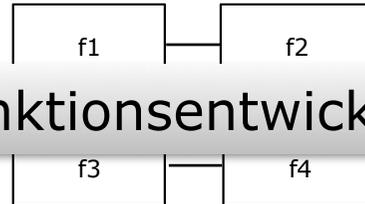


Rollen und Aufgabengebiete in der Entwicklung

Rolle	Aufgabengebiet
Funktionsentwicklung	Analyse der Benutzeranforderungen und Spezifikation der logischen Systemarchitektur
Systementwicklung	Analyse der logischen Systemarchitektur und Spezifikation der technischen Systemarchitektur
Software-Entwicklung	Analyse der Software-Anforderungen, Spezifikation, Design, Implementierung, Integration und Test der Software
Hardware-Entwicklung	Analyse der Hardware-Anforderungen, Spezifikation, Design, Realisierung, Integration und Test der Hardware
Sensor-/Sollwertgeber-/ Aktuator-Entwicklung	Analyse der Anforderungen, Spezifikation, Design, Realisierung, Integration und Test von Sensoren, Sollwertgebern und Aktuatoren
Integration, Erprobung und Kalibrierung	Integration, Test und Kalibrierung von Systemen des Fahrzeugs und deren Funktionen

Analyse der Benutzeranforderungen und Spezifikation der logischen Systemarchitektur

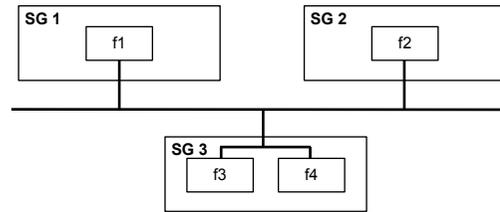
Funktionsentwicklung



Logische Systemarchitektur

Akzeptanz- und Systemtest

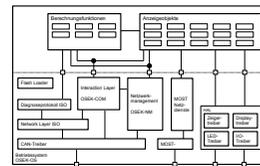
Analyse der logischen Systemarchitektur und Spezifikation der technischen Systemarchitektur



Technische Systemarchitektur

Kalibrierung
 Integrationstest des Systems
 Integration der System-Komponenten

Analyse der Software-Anforderungen und Spezifikation der technischen Softwarearchitektur



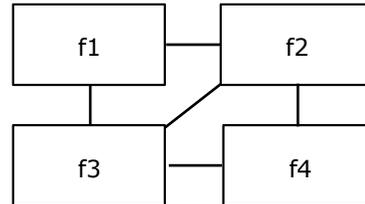
Software-Architektur

Integrationstest der Software
 Integration der Software-Komponenten

Spezifikation der Software-Komponenten
 Design und Implementierung der Software-Komponenten

Test der Software-Komponenten

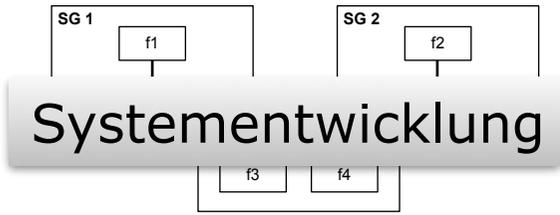
Analyse der Benutzeranforderungen und Spezifikation der logischen Systemarchitektur



Logische Systemarchitektur

Akzeptanz- und Systemtest

Analyse der logischen Systemarchitektur und Spezifikation der technischen Systemarchitektur

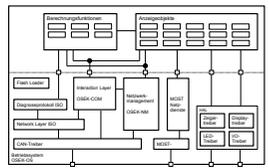


Systementwicklung

Technische Systemarchitektur

Kalibrierung
 Integrationstest des Systems
 Integration der System-Komponenten

Analyse der Software-Anforderungen und Spezifikation der technischen Softwarearchitektur



Software-Architektur

Integrationstest der Software
 Integration der Software-Komponenten

Spezifikation der Software-Komponenten
 Design und Implementierung der Software-Komponenten

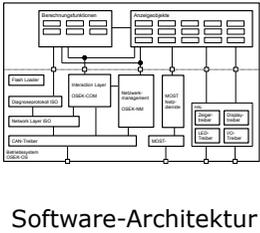
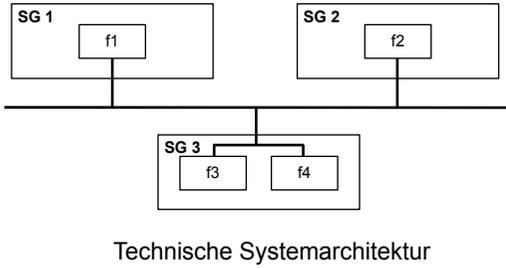
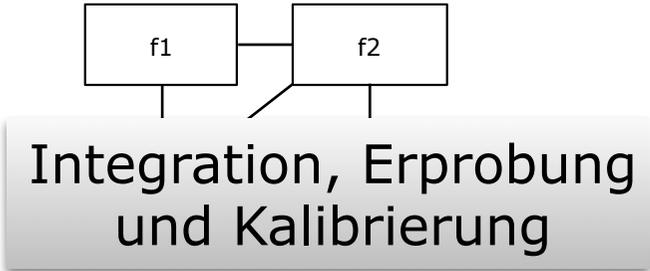
Test der Software-Komponenten

Analyse der Benutzeranforderungen und Spezifikation der logischen Systemarchitektur

Analyse der logischen Systemarchitektur und Spezifikation der technischen Systemarchitektur

Analyse der Software-Anforderungen und Spezifikation der technischen Softwarearchitektur

Spezifikation der Software-Komponenten
Design und Implementierung der Software-Komponenten



Kalibrierung
Integrationstest des Systems
Integration der System-Komponenten

Integrationstest der Software
Integration der Software-Komponenten

Test der Software-Komponenten

Akzeptanz- und Systemtest

6. SW-Entwicklung / 3. Unterstützungsprozesse

Unterstützungsprozesse für die Embedded Software Entwicklung

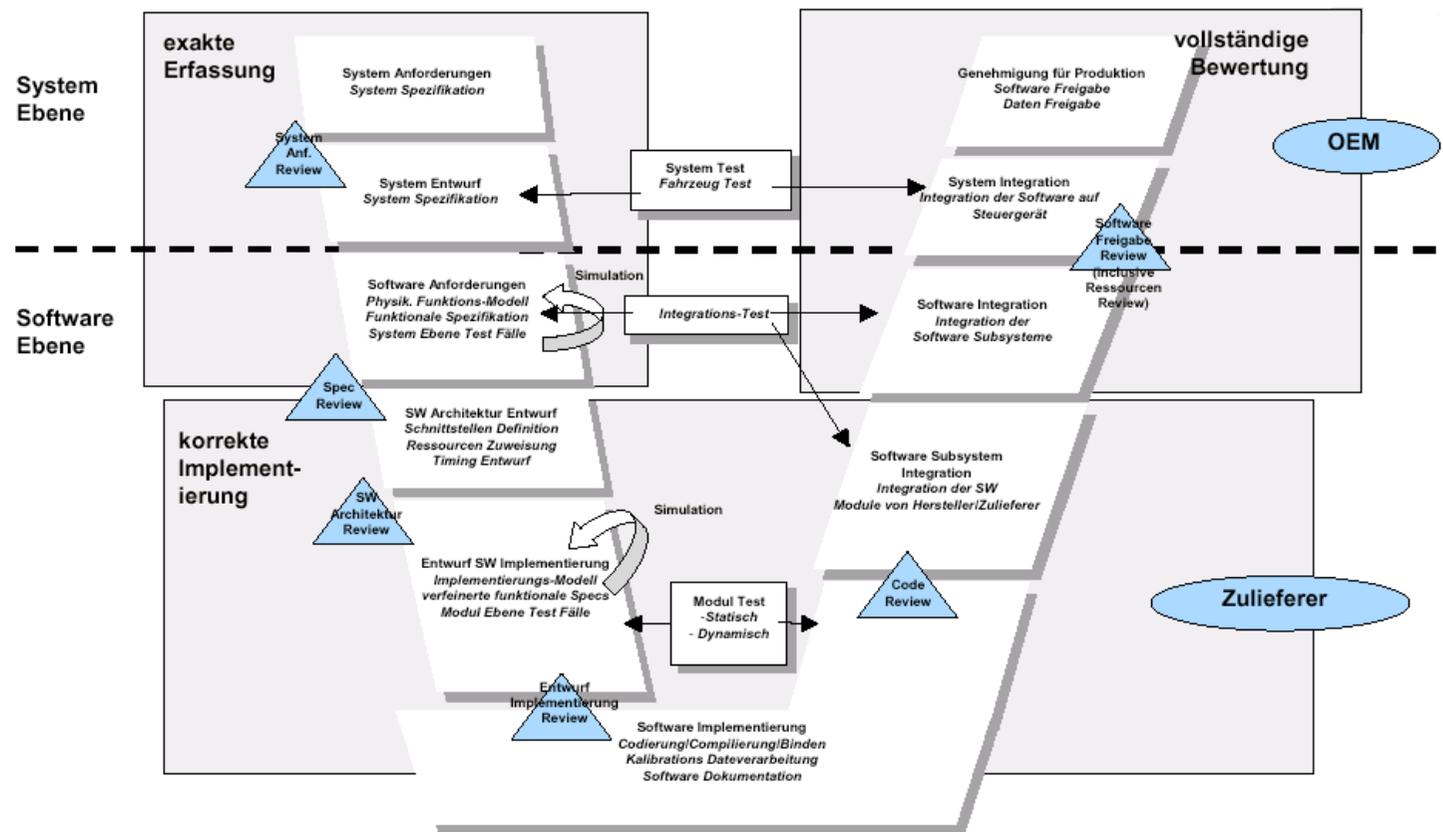
1. Vorgehensmodelle und Standards
2. Konfigurationsmanagement
3. Projektmanagement
4. **Lieferantenmanagement**
 1. **System- und Komponentenverantwortung**
 2. Schnittstellen für Spezifikation und Integration
 3. Festlegung des firmenübergreifenden Entwicklungsprozesses
5. Anforderungsmanagement
6. Qualitätssicherung

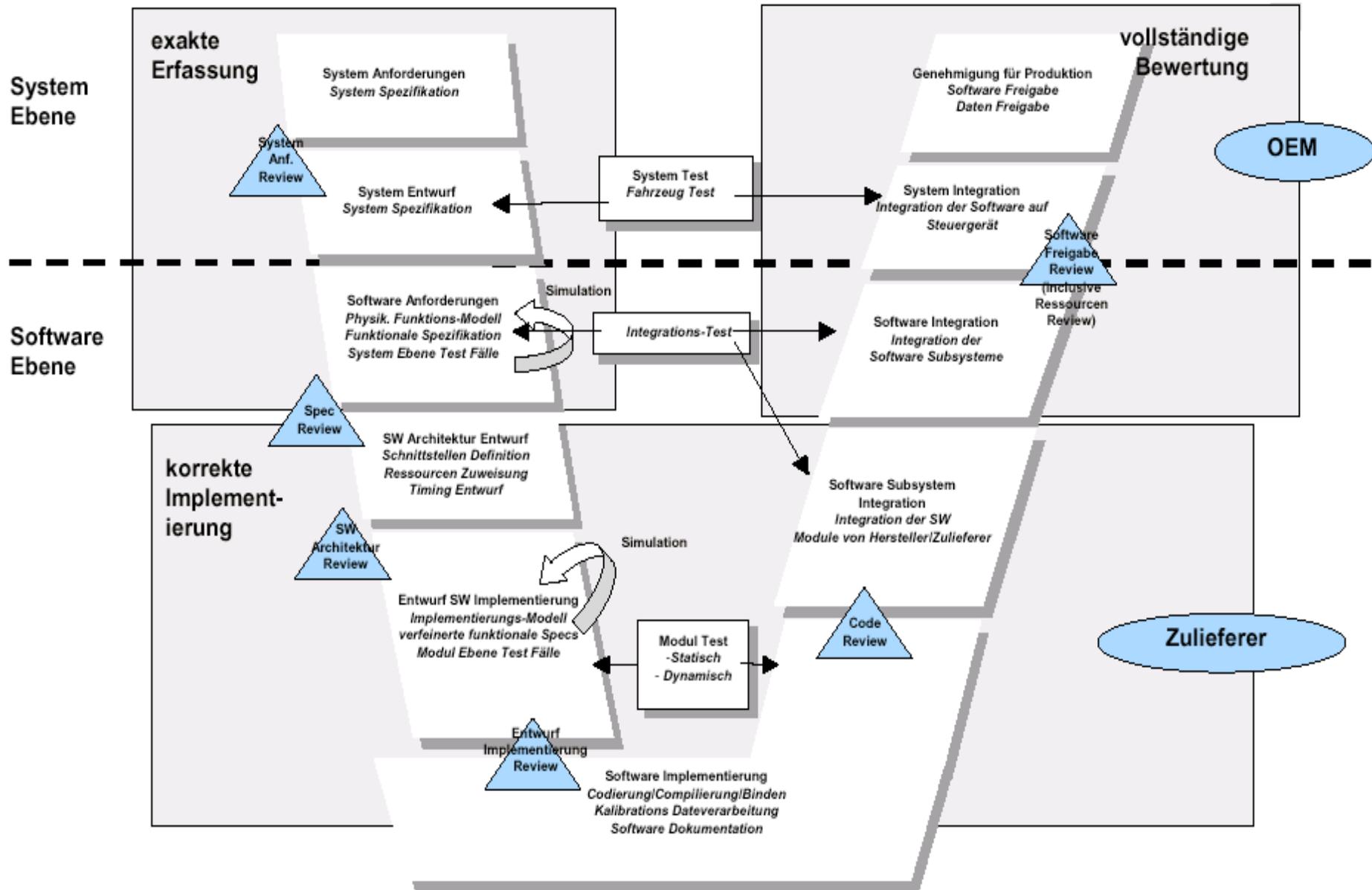
Entwicklung elektronischer Systeme im Automobil

- Starke Arbeitsteilung zwischen Fahrzeugherstellern und Zulieferern
- Festlegung der Anforderungen an eine zu entwickelnde Funktion
 - Fahrzeughersteller
- Realisierung der Funktion durch elektronische Systeme
 - Zulieferer
- Abstimmung und Abnahme der Funktion im Fahrzeug
 - Fahrzeughersteller
- Firmenübergreifende Zusammenarbeit
 - Technische Aspekte
 - Organisatorische Aspekte
 - Rechtliche Aspekte
- Lieferantenmanagement
 - Alle Aufgaben, die im Rahmen der Systementwicklung an den Schnittstellen zwischen Fahrzeughersteller und Zulieferern beachtet werden müssen

System- und Komponentenverantwortung

- Präzise Definition der Schnittstellen zwischen Fahrzeughersteller und Zulieferern
- Orientierung am V-Modell
- Fahrzeughersteller
 - Gesamtfahrzeug
- Zulieferer
 - Komponenten





BMW Group Standard embedded Software

- BMW Group Standard embedded Software (GSeSW)
- Beispiel für einen firmeninternen Standard zur Beschreibung des (Software-) Entwicklungsprozesses
- Quelle - NICHT MEHR AKTUELL, Datei bei dSpace angefordert
- http://www.dspace.de/shared/data/pdf/ankon2004/05_bmw_riedesser-rissling_e.pdf

BMW Group Standard

GS 95014

2007-04

Deskriptoren: Embedded Software, Entwicklung, Entwicklungsprozess
Descriptors: Embedded Software, Development, Development Process

Embedded Software Entwicklung (ESW)

Embedded Software Development (ESW)

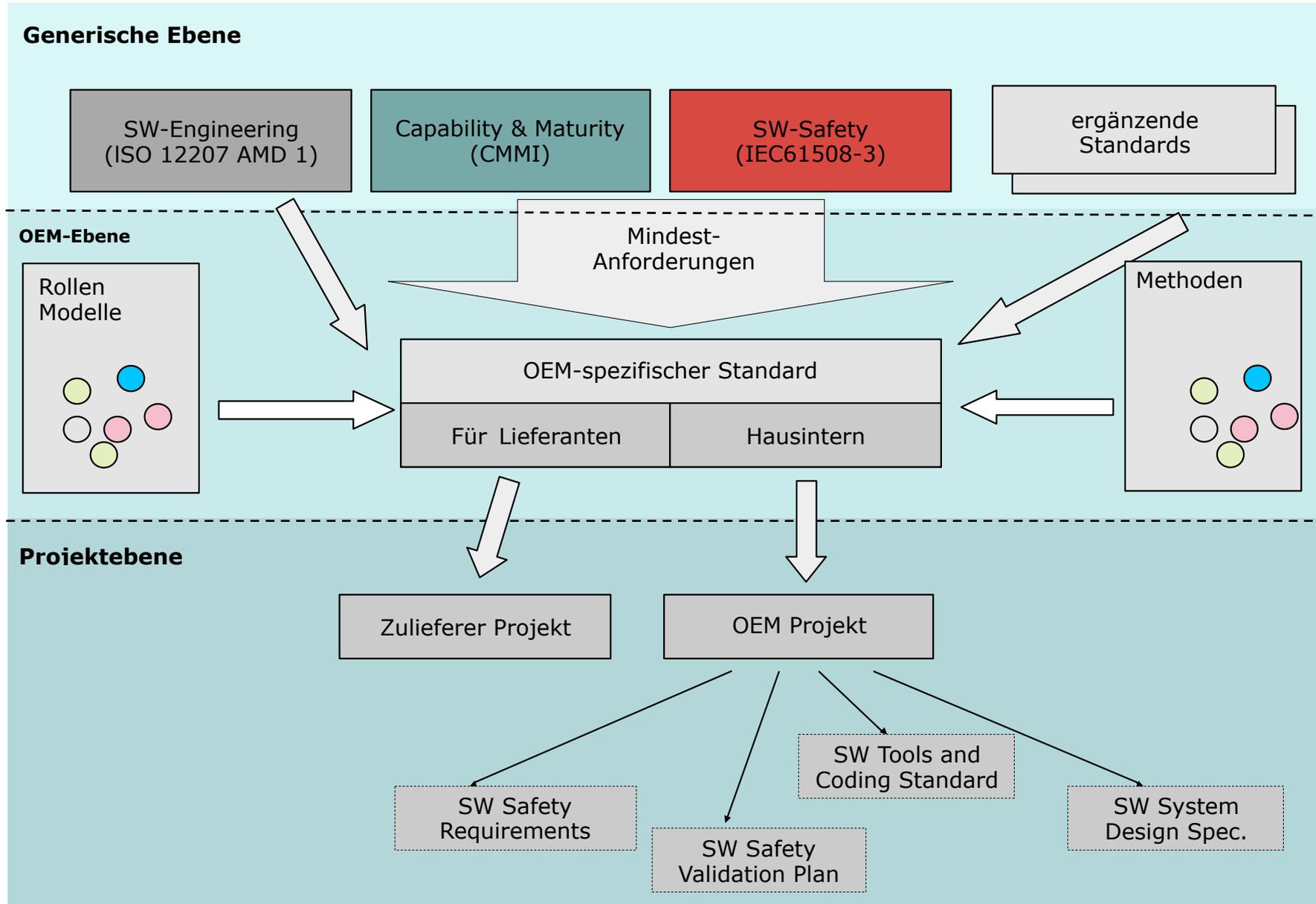
Ausdrucke unterliegen nicht dem Änderungsdienst.
Print-outs are not subject to the change service.

Fortsetzung Seite 2 bis 177
Continued on pages 2 to 177

BMW AG Normung: 80788 München

© BMW AG Alle Rechte vorbehalten / All rights reserved
Interleaf.doc Bearbeiter / Editor: Robert Maier





6. SW-Entwicklung / 3. Unterstützungsprozesse

Unterstützungsprozesse für die Embedded Software Entwicklung

1. Vorgehensmodelle und Standards
2. Konfigurationsmanagement
3. Projektmanagement
4. **Lieferantenmanagement**
 1. System- und Komponentenverantwortung
 2. **Schnittstellen für Spezifikation und Integration**
 3. Festlegung des firmenübergreifenden Entwicklungsprozesses
5. Anforderungsmanagement
6. Qualitätssicherung

Schnittstellen für Spezifikation und Integration

- Zwei Arten von Schnittstellen
 - Spezifikationsschnittstelle im linken Ast des V-Modells
 - Integrationsschnittstelle im rechten Ast des V-Modells
- Grosse Komplexität
 - Hersteller
 - n Komponenten von n verschiedenen Zulieferern
 - 1:n-Beziehung an der Spezifikationsschnittstelle
 - 1:n-Beziehung an der Integrationsschnittstelle
 - Zulieferer
 - 1 Komponente an m verschiedene Hersteller
 - 1:m-Beziehung an der Spezifikationsschnittstelle
 - 1:m-Beziehung an der Integrationsschnittstelle

Herstellersicht Baugruppenverantwortlicher Türe

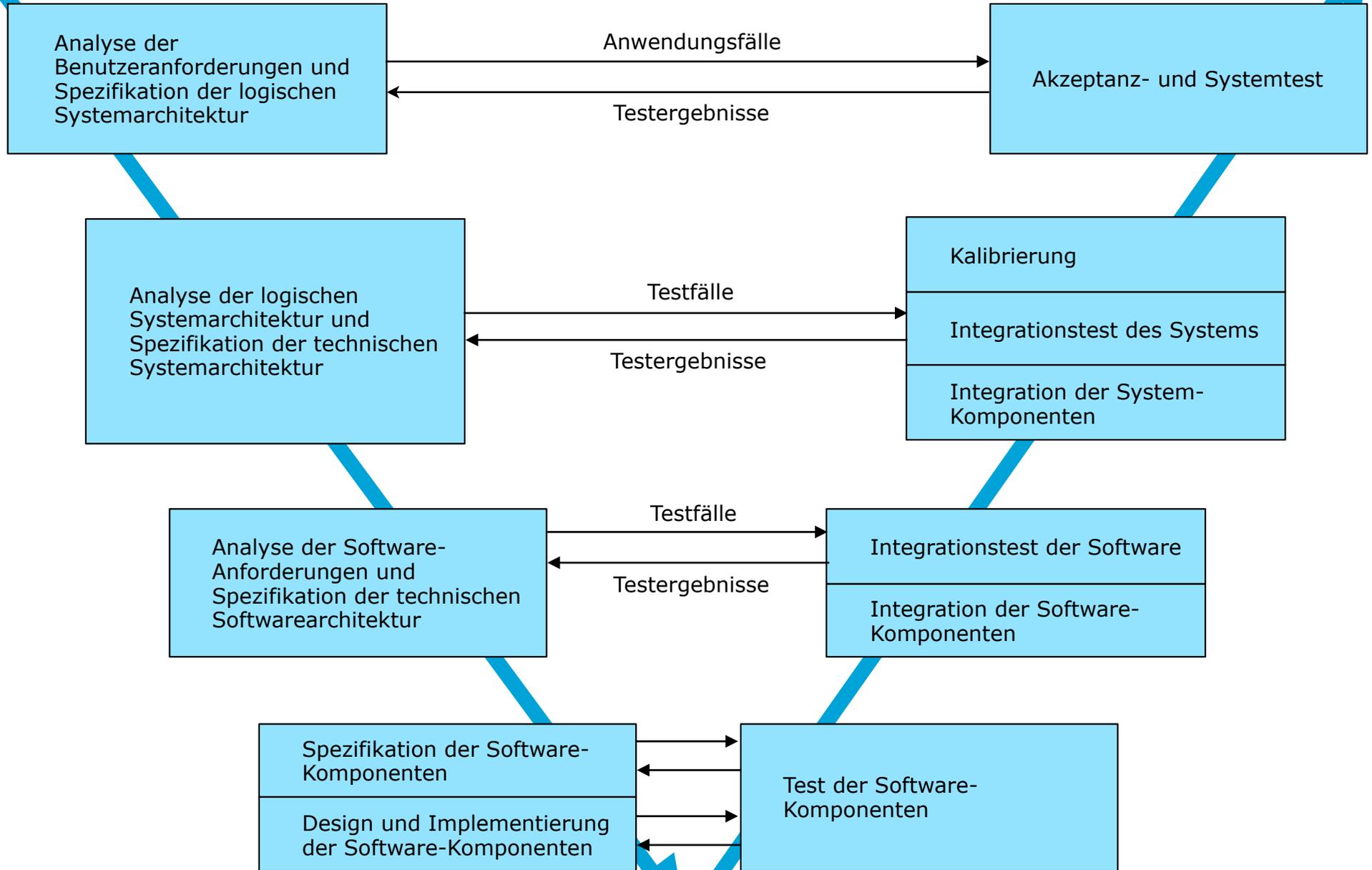
- Ansprechpartner
 - Baugruppenverantwortlicher Karosserie
 - Baugruppenverantwortlicher Sitze
 - Baugruppenverantwortlicher Kombi-Instrument
 - Baugruppenverantwortlicher Blinker
 - Baugruppenverantwortlicher Mittelkonsole
 - Baugruppenverantwortlicher Soundsystem
 - Baugruppenverantwortlicher Seitenairbag
 - Verantwortlicher Passive Sicherheit
 - Verantwortlicher EMV
 - Verantwortlicher Verkabelung
 - Verantwortlicher Vernetzung
- Verantwortlicher Telematik
- **Zulieferer**
 - **Schliesssystem**
 - **Scheiben**
 - **Fensterheber**
 - **Aussenspiegel**
 - **Türsteuergerät**
 - **Schalter**
- Schnittstellen
 - Mechanik
 - Energie
 - Information

Herstellersicht (Prinzipdarstellung)

	Bosch	Conti	Kostal	Dräxl- meier	Magna	Delphi	TRW
Schliesssystem		X	X				
Scheiben			X				
Fensterheber			X	X	X		
Aussenspiegel					X		X
Türsteuergerät	X	X					
Schalter						X	X

Zulieferersicht (Prinzipdarstellung)

	BMW	Mer- cedes	Audi	Por- sche	VW	Ford	Opel
Schliesssystem	X	X	X				
Scheiben			X				
Fensterheber			X	X	X		
Aussenspiegel	X	X	X	X	X	X	X
Türsteuergerät		X					
Schalter						X	X



Analyse der Benutzeranforderungen und Spezifikation der logischen Systemarchitektur

Akzeptanz- und Systemtest

Anwendungsfälle

Testergebnisse

Analyse der logischen Systemarchitektur und Spezifikation der technischen Systemarchitektur

Kalibrierung
Integrationstest des Systems
Integration der System-Komponenten

Testfälle

Testergebnisse

Analyse der Software-Anforderungen und Spezifikation der technischen Softwarearchitektur

Integrationstest der Software
Integration der Software-Komponenten

Testfälle

Testergebnisse

Spezifikation der Software-Komponenten
Design und Implementierung der Software-Komponenten

Test der Software-Komponenten

Analyse der Benutzeranforderungen und Spezifikation der logischen Systemarchitektur

Akzeptanz- und Systemtest

Anwendungsfälle

Testergebnisse

Analyse der logischen Systemarchitektur und Spezifikation der technischen Systemarchitektur

Kalibrierung
 Integrationstest des Systems
 Integration der System-Komponenten

Testfälle

Testergebnisse

Analyse der Software-Anforderungen und Spezifikation der technischen Softwarearchitektur

Integrationstest der Software
 Integration der Software-Komponenten

Testfälle

Testergebnisse

Spezifikation der Software-Komponenten
 Design und Implementierung der Software-Komponenten

Test der Software-Komponenten

Analyse der Benutzeranforderungen und Spezifikation der logischen Systemarchitektur

Akzeptanz- und Systemtest

Anwendungsfälle

Testergebnisse

Analyse der logischen Systemarchitektur und Spezifikation der technischen Systemarchitektur

Kalibrierung
 Integrationstest des Systems
 Integration der System-Komponenten

Testfälle

Testergebnisse

Analyse der Software-Anforderungen und Spezifikation der technischen Softwarearchitektur

Integrationstest der Software
 Integration der Software-Komponenten

Testfälle

Testergebnisse

Spezifikation der Software-Komponenten
 Design und Implementierung der Software-Komponenten

Test der Software-Komponenten

Analyse der Benutzeranforderungen und Spezifikation der logischen Systemarchitektur

Akzeptanz- und Systemtest

Anwendungsfälle

Testergebnisse

Analyse der logischen Systemarchitektur und Spezifikation der technischen Systemarchitektur

Kalibrierung
 Integrationstest des Systems
 Integration der System-Komponenten

Testfälle

Testergebnisse

Analyse der Software-Anforderungen und Spezifikation der technischen Softwarearchitektur

Integrationstest der Software
 Integration der Software-Komponenten

Testfälle

Testergebnisse

Spezifikation der Software-Komponenten
 Design und Implementierung der Software-Komponenten

Test der Software-Komponenten

Analyse der
Benutzeranforderungen und
Spezifikation der logischen
Systemarchitektur

Akzeptanz- und Systemtest

Anwendungsfälle

Testergebnisse

Analyse der logischen
Systemarchitektur und
Spezifikation der technischen
Systemarchitektur

Kalibrierung
 Integrationstest des Systems
 Integration der System-
Komponenten

Testfälle

Testergebnisse

Analyse der Software-
Anforderungen und
Spezifikation der technischen
Softwarearchitektur

Integrationstest der Software
 Integration der Software-
Komponenten

Testfälle

Testergebnisse

Spezifikation der Software-
Komponenten
 Design und Implementierung
der Software-Komponenten

Test der Software-
Komponenten

6. SW-Entwicklung / 3. Unterstützungsprozesse

Unterstützungsprozesse für die Embedded Software Entwicklung

1. Vorgehensmodelle und Standards
2. Konfigurationsmanagement
3. Projektmanagement
4. **Lieferantenmanagement**
 1. System- und Komponentenverantwortung
 2. Schnittstellen für Spezifikation und Integration
 3. **Festlegung des firmenübergreifenden Entwicklungsprozesses**
5. Anforderungsmanagement
6. Qualitätssicherung

Firmenübergreifender Entwicklungsprozess

- Elektronische Steuergeräte sind eingebettete Systeme, die für den Benutzer nicht unmittelbar in Erscheinung treten
 - Beispiel: Türsteuergerät
- Für den Benutzer sind die Funktionen sichtbar
 - Beispiele:
 - Fensterheber
 - Spiegelverstellung
 - Sitzverstellung
- Grundfunktionen können herstellerübergreifend entwickelt werden
 - Beispiel:
 - Steuerung von Fensterheber, Spiegelverstellung, Sitzverstellung
- Wettbewerbsdifferenzierende Zusatzfunktionen werden herstellerspezifisch entwickelt
 - Beispiel:
 - Automatische Spiegelverstellung und Sitzverstellung (Memory, Ergonomie)

Firmenübergreifender Entwicklungsprozess

- Elektronische Steuergeräte sind eingebettete Systeme, die für den Benutzer nicht unmittelbar in Erscheinung treten
 - Beispiel: Türsteuergerät
- Für den Benutzer sind die Funktionen sichtbar
 - Beispiele:
 - Fensterheber
 - Spiegelverstellung
 - Sitzverstellung
- Grundfunktionen können herstellerübergreifend entwickelt werden
 - Beispiel:
 - Steuerung von Fensterheber, Spiegelverstellung, Sitzverstellung
- Wettbewerbsdifferenzierende Zusatzfunktionen werden herstellerspezifisch entwickelt
 - Beispiel:
 - Automatische Spiegelverstellung und Sitzverstellung (Memory, Ergonomie)

Grundidee von

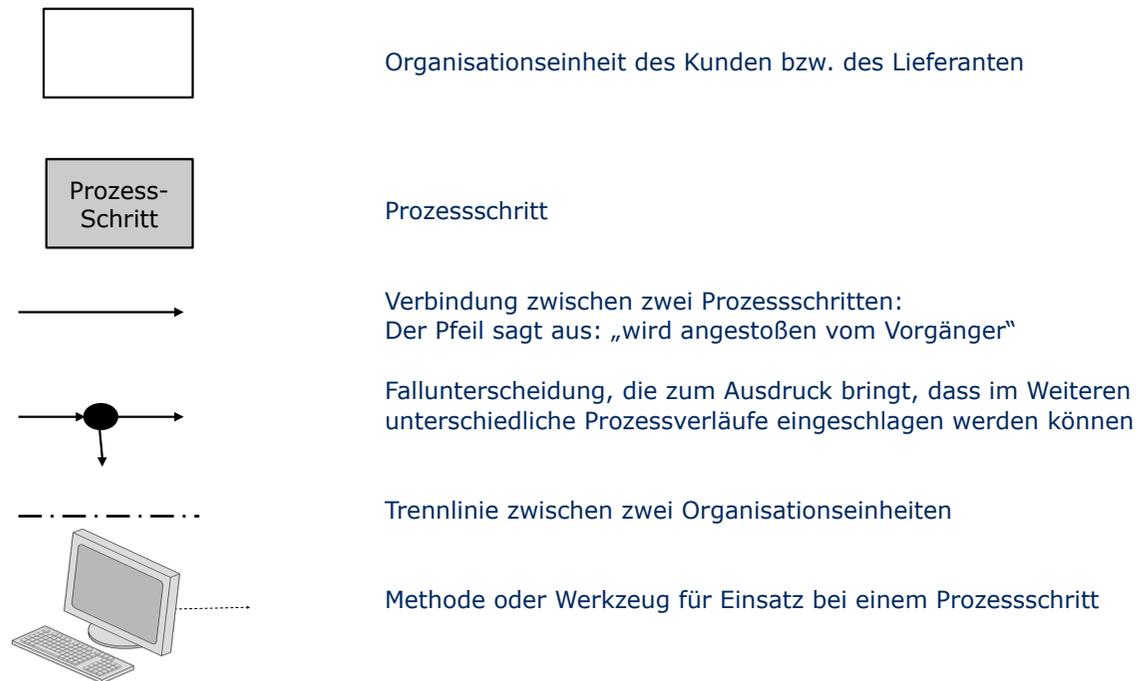
Enabling Innovation

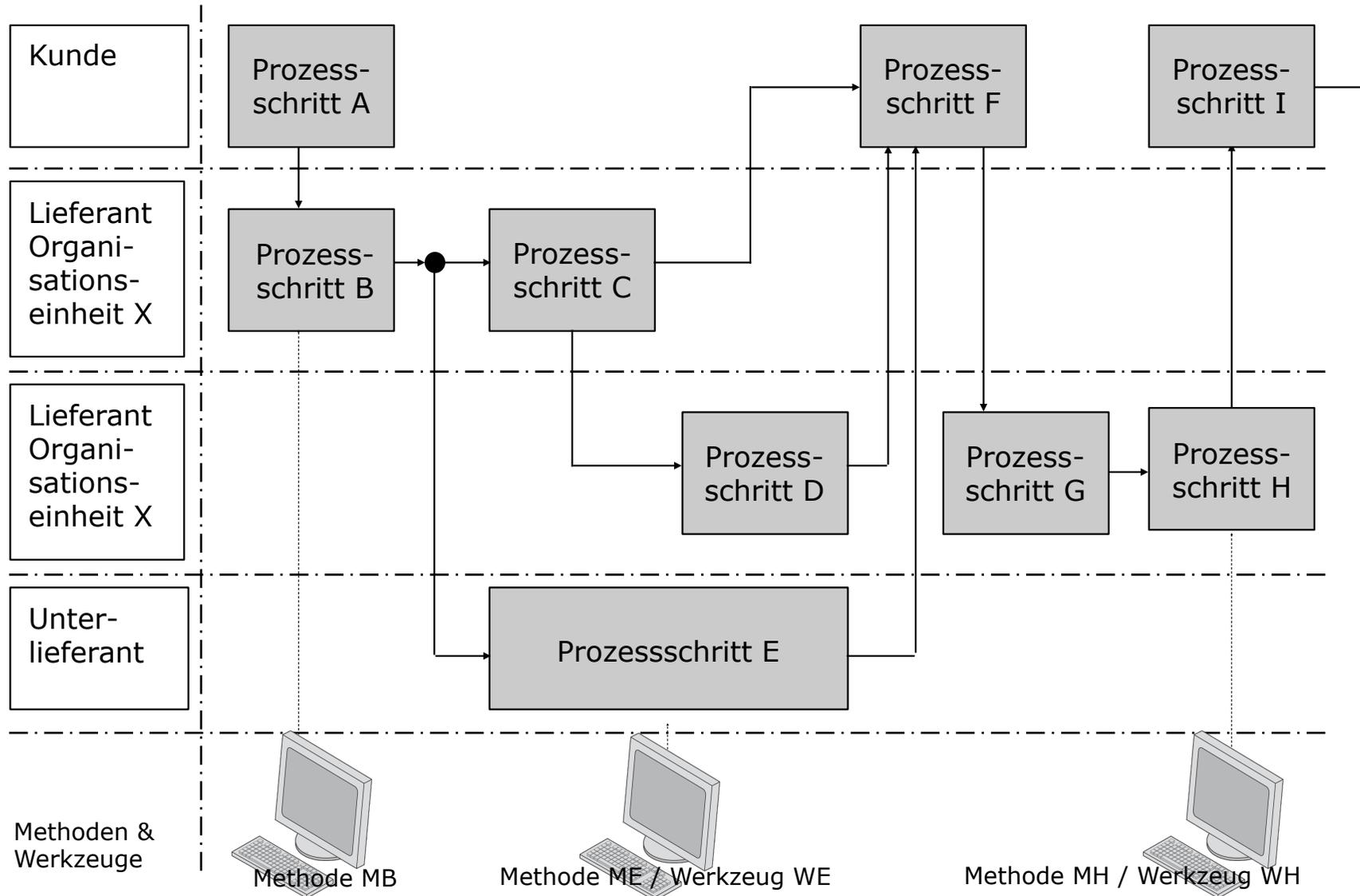


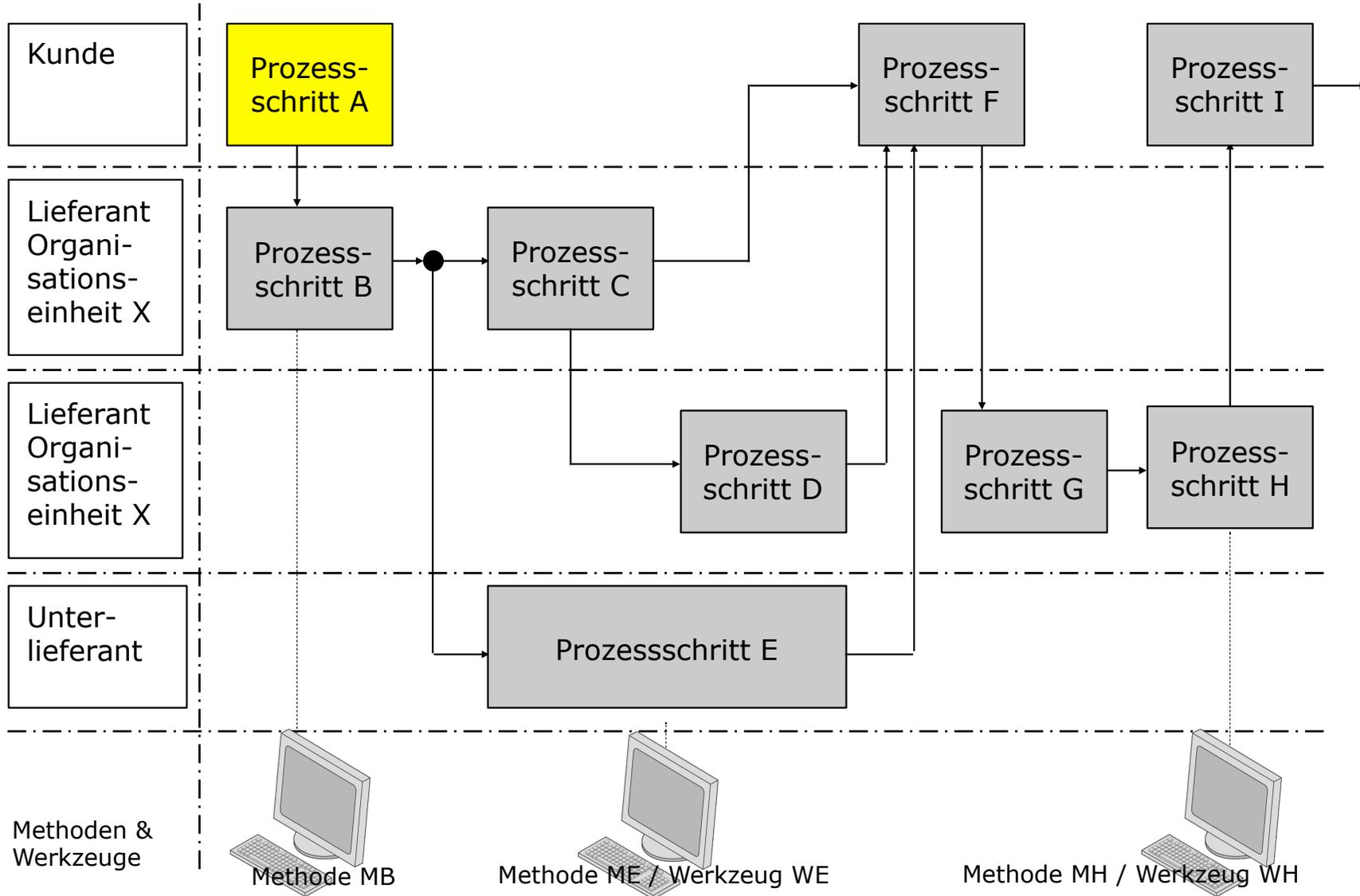
LOV-Diagramme

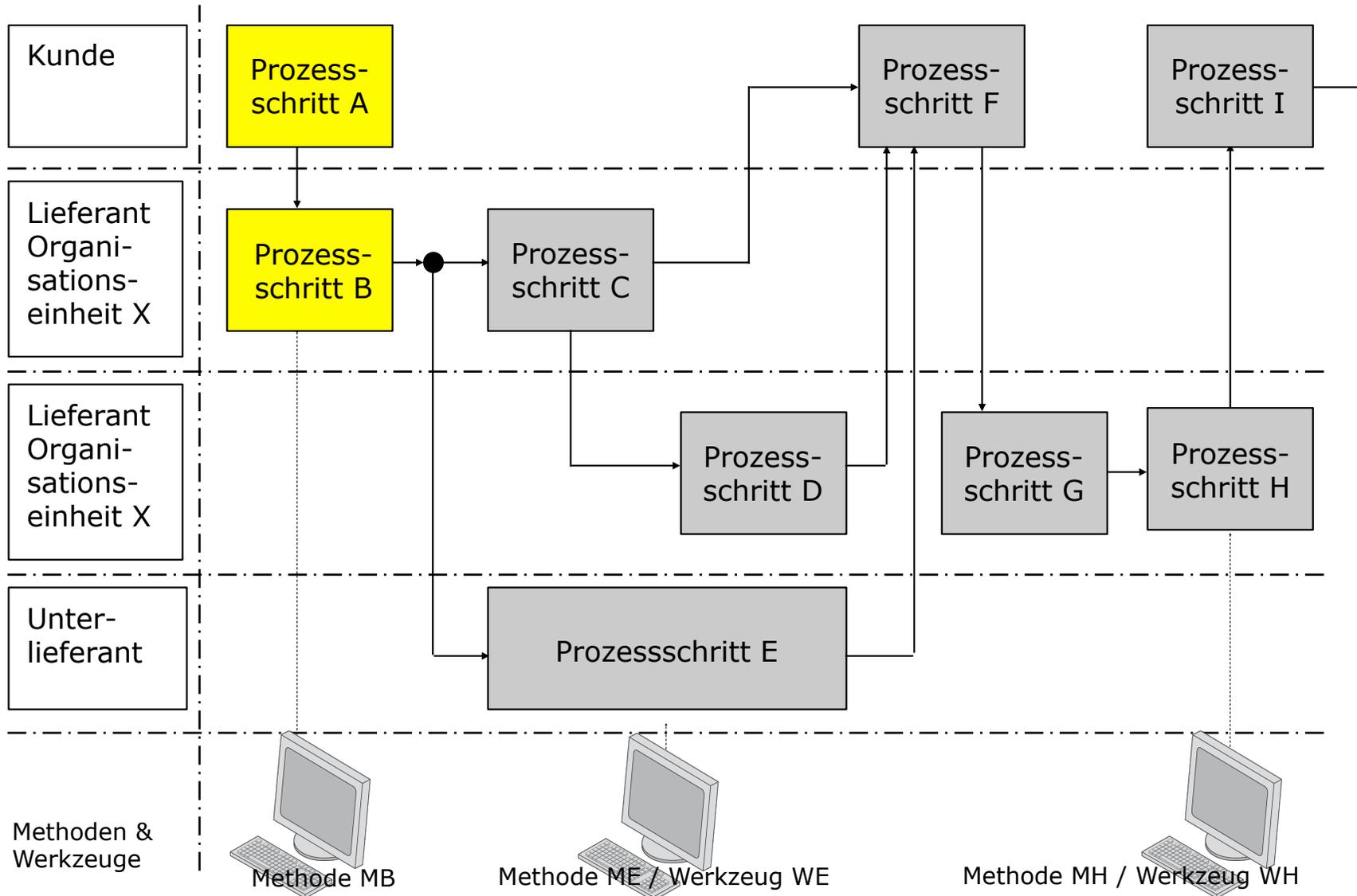
- LOV: Line of Visibility
- Grafische Beschreibung der komplexen Beziehungen zwischen Herstellern und Zulieferern

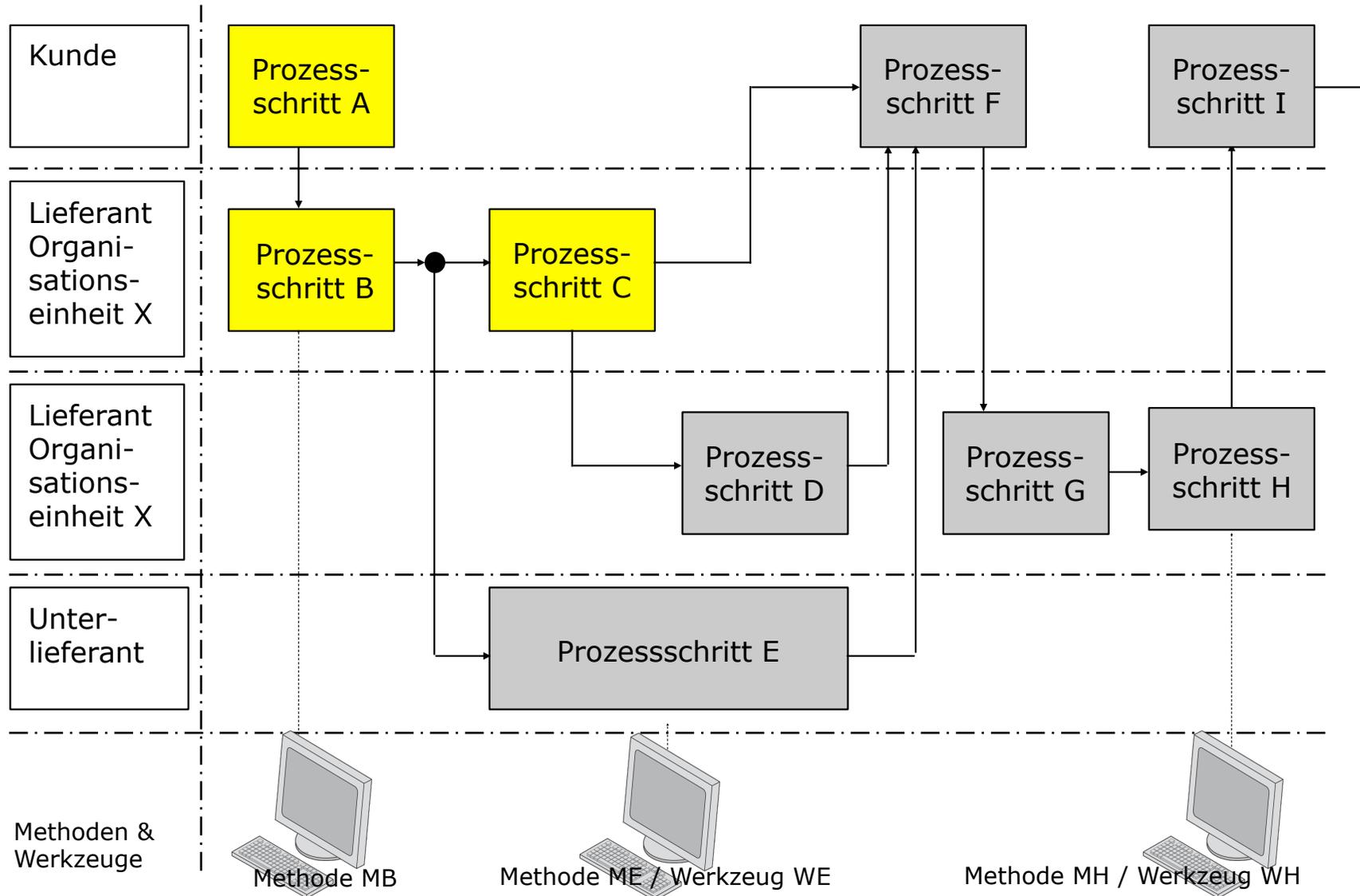
In der linken Spalte des LOV-Diagramms:
Für Prozessschritte verantwortliche Organisationseinheit
In der oberen Zeile des LOV-Diagramms:
Prozessschritte des Kunden

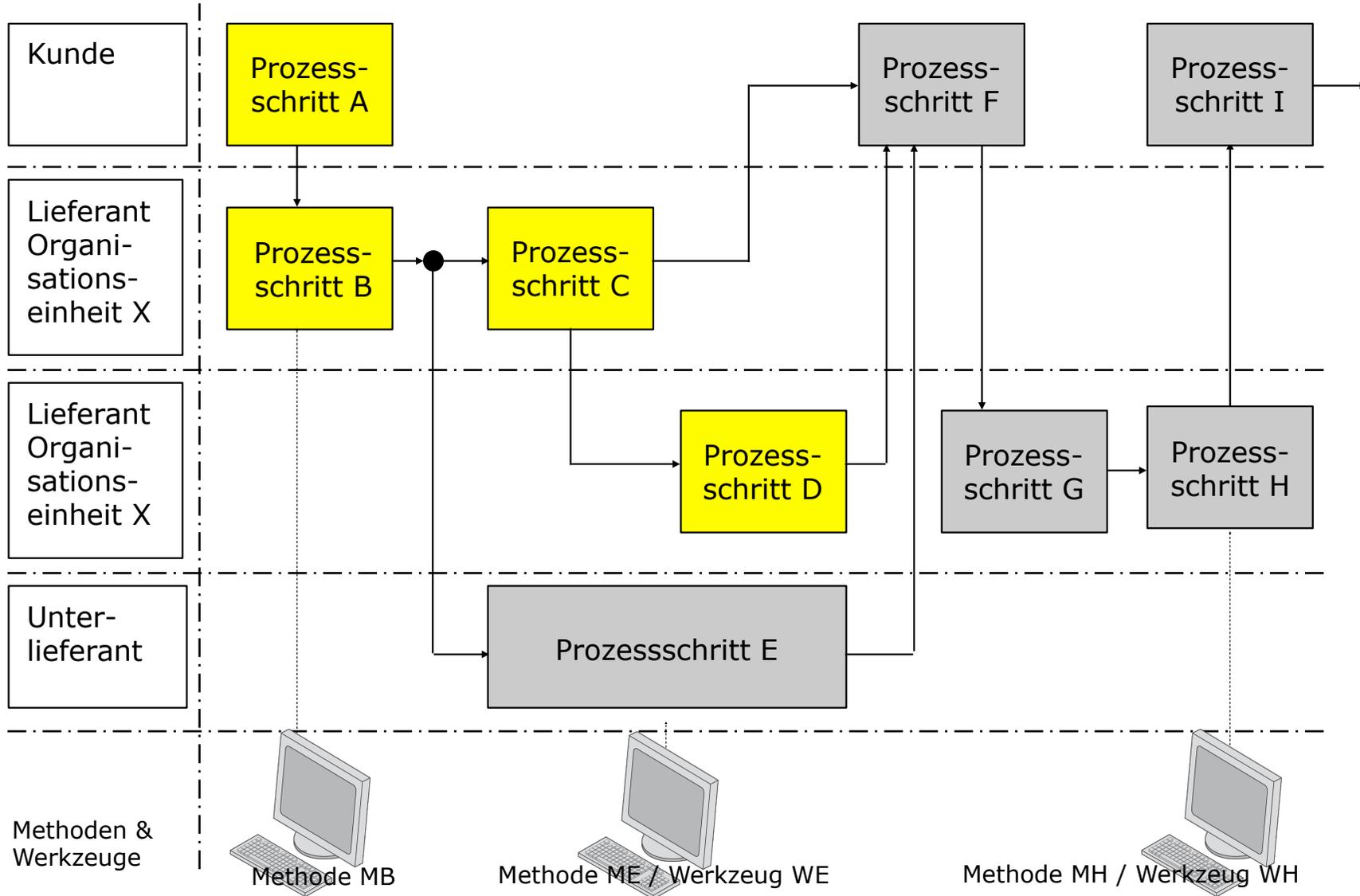


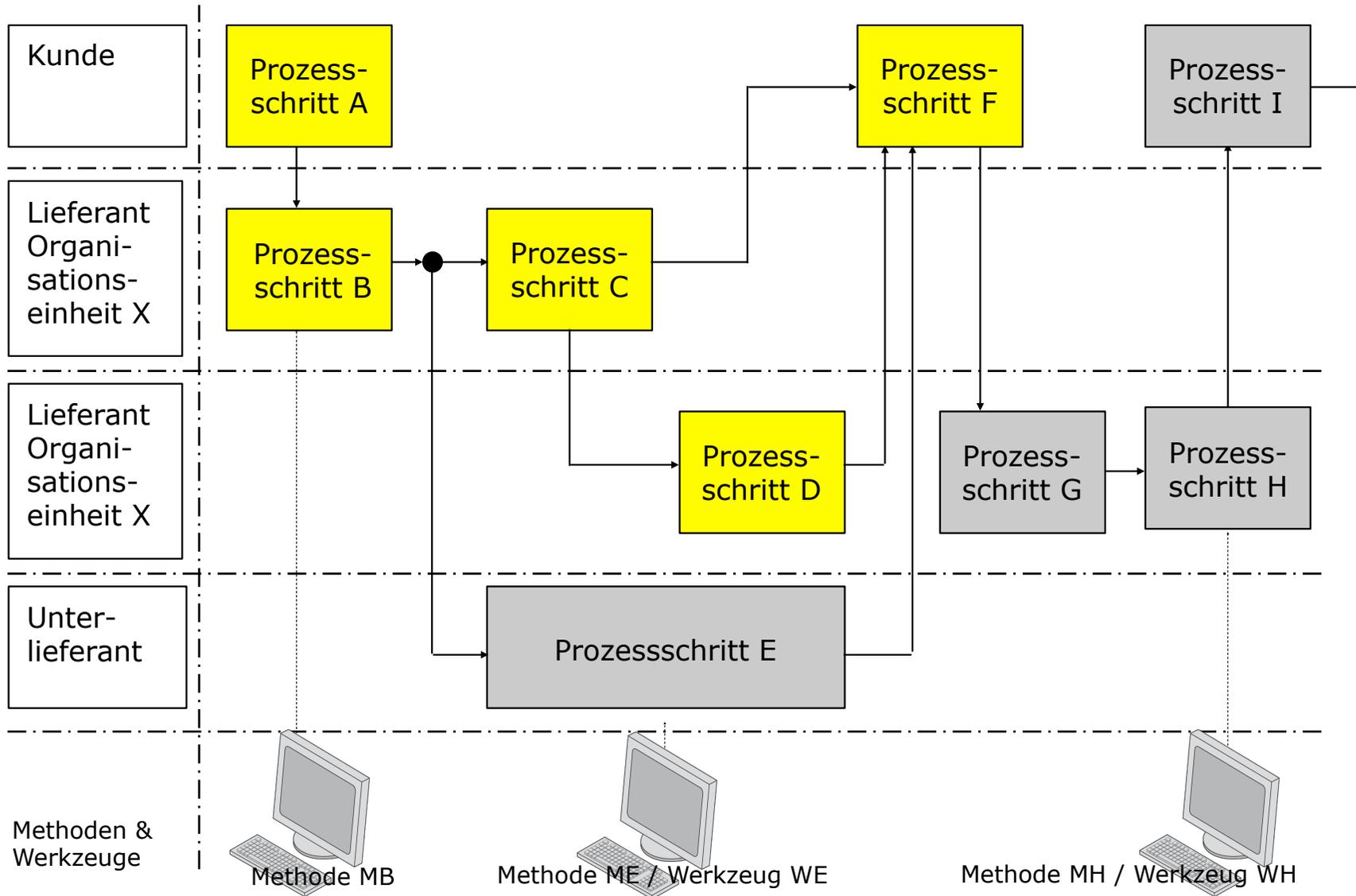


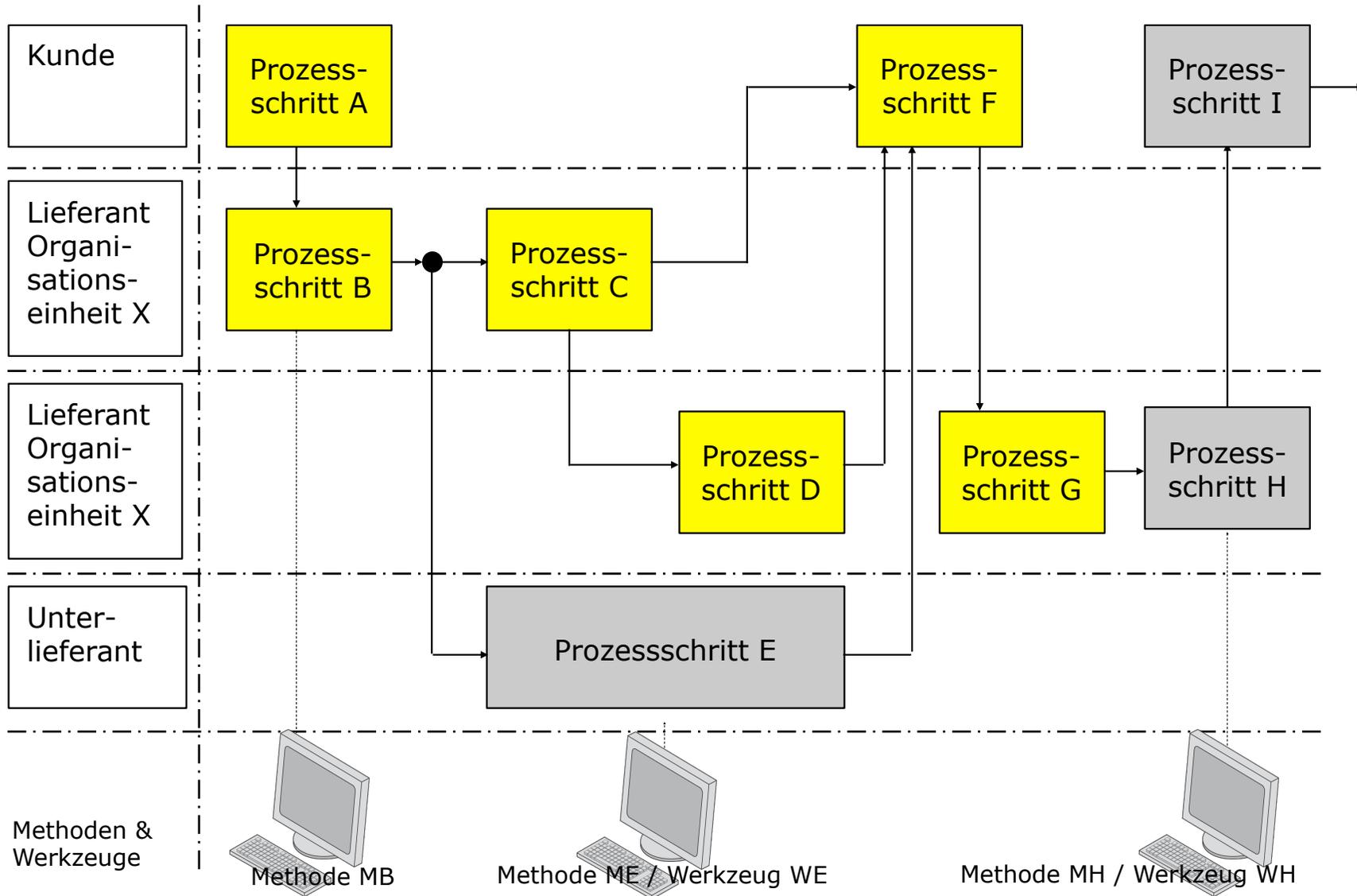


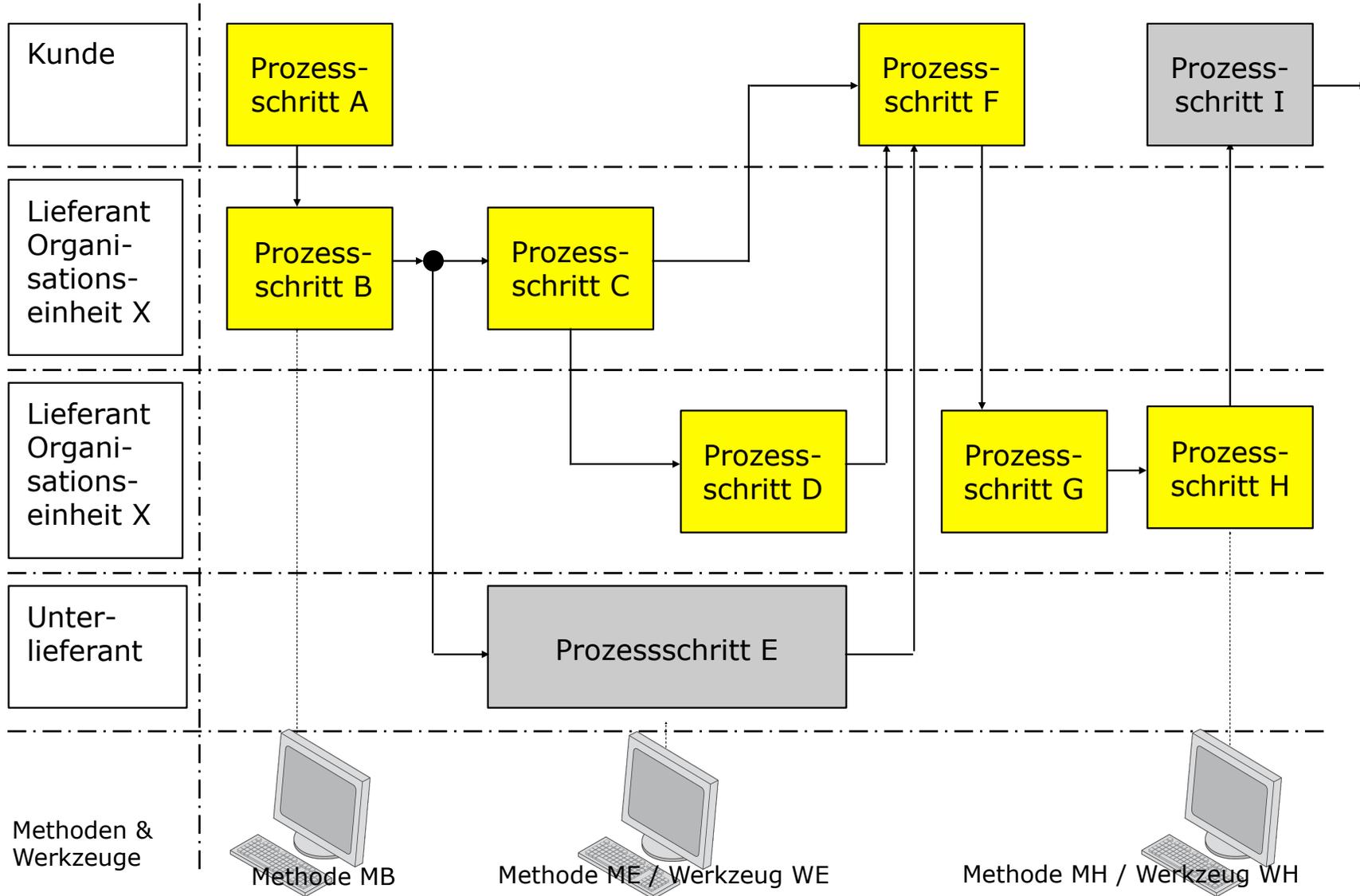


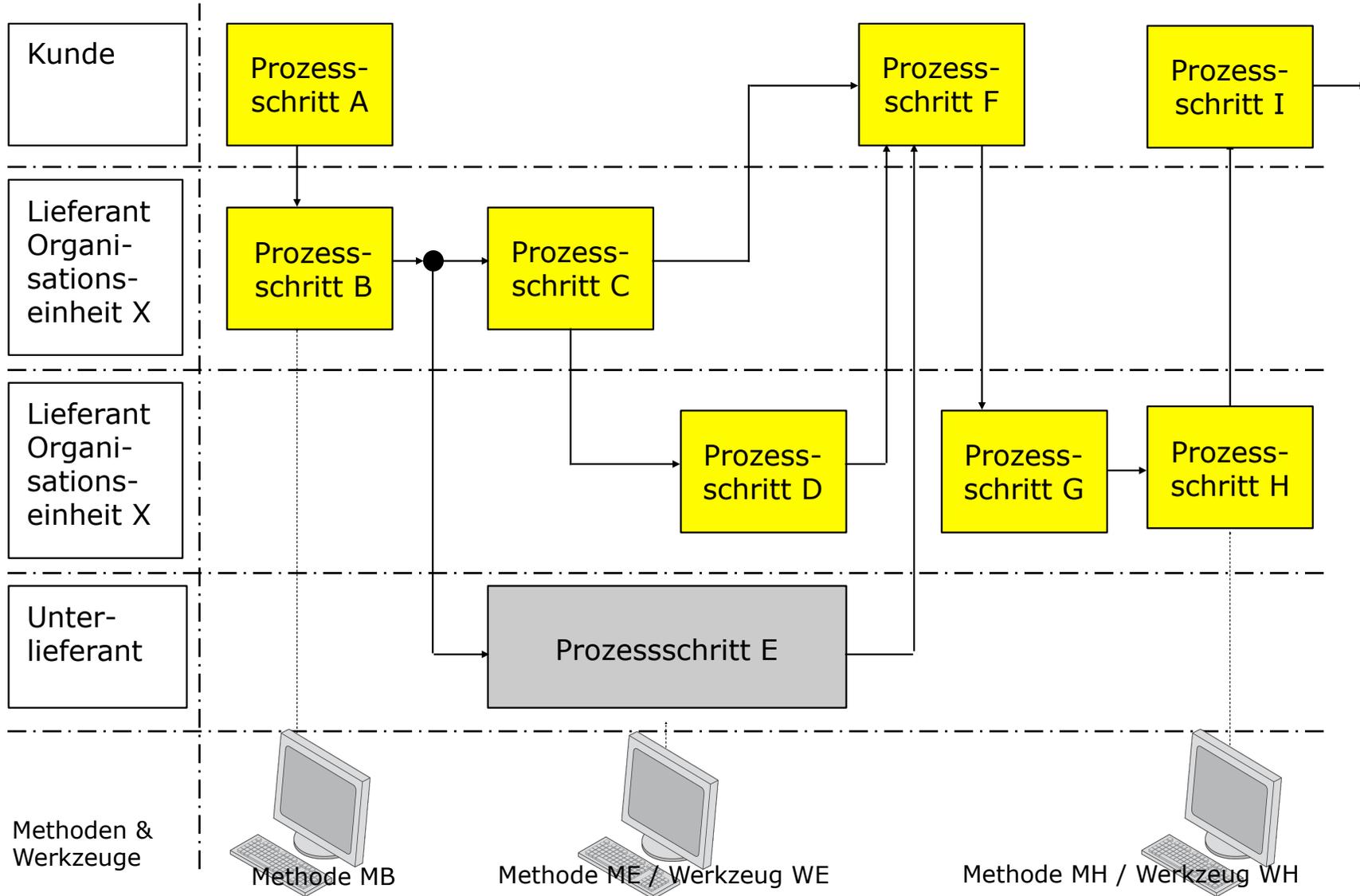


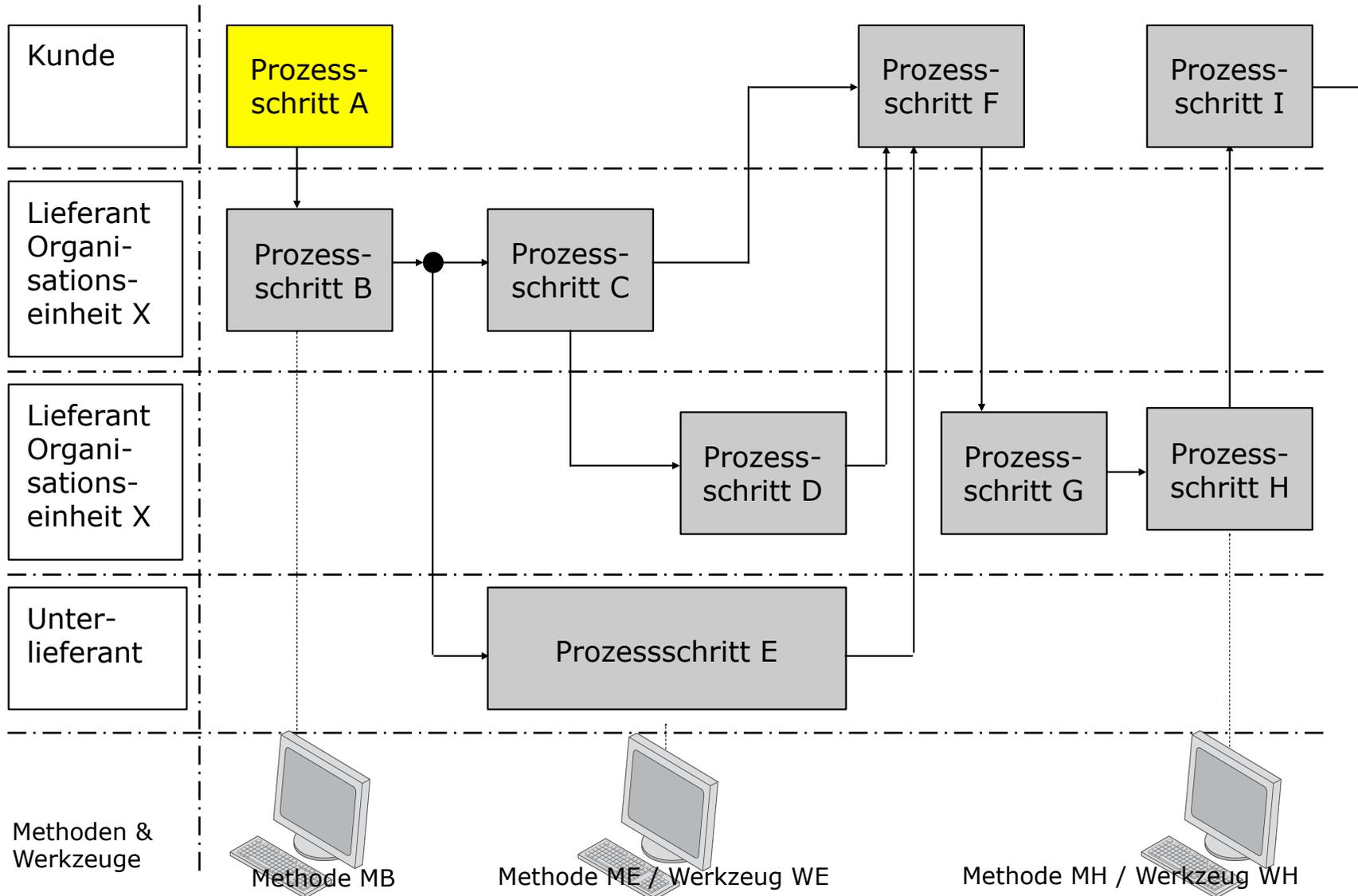


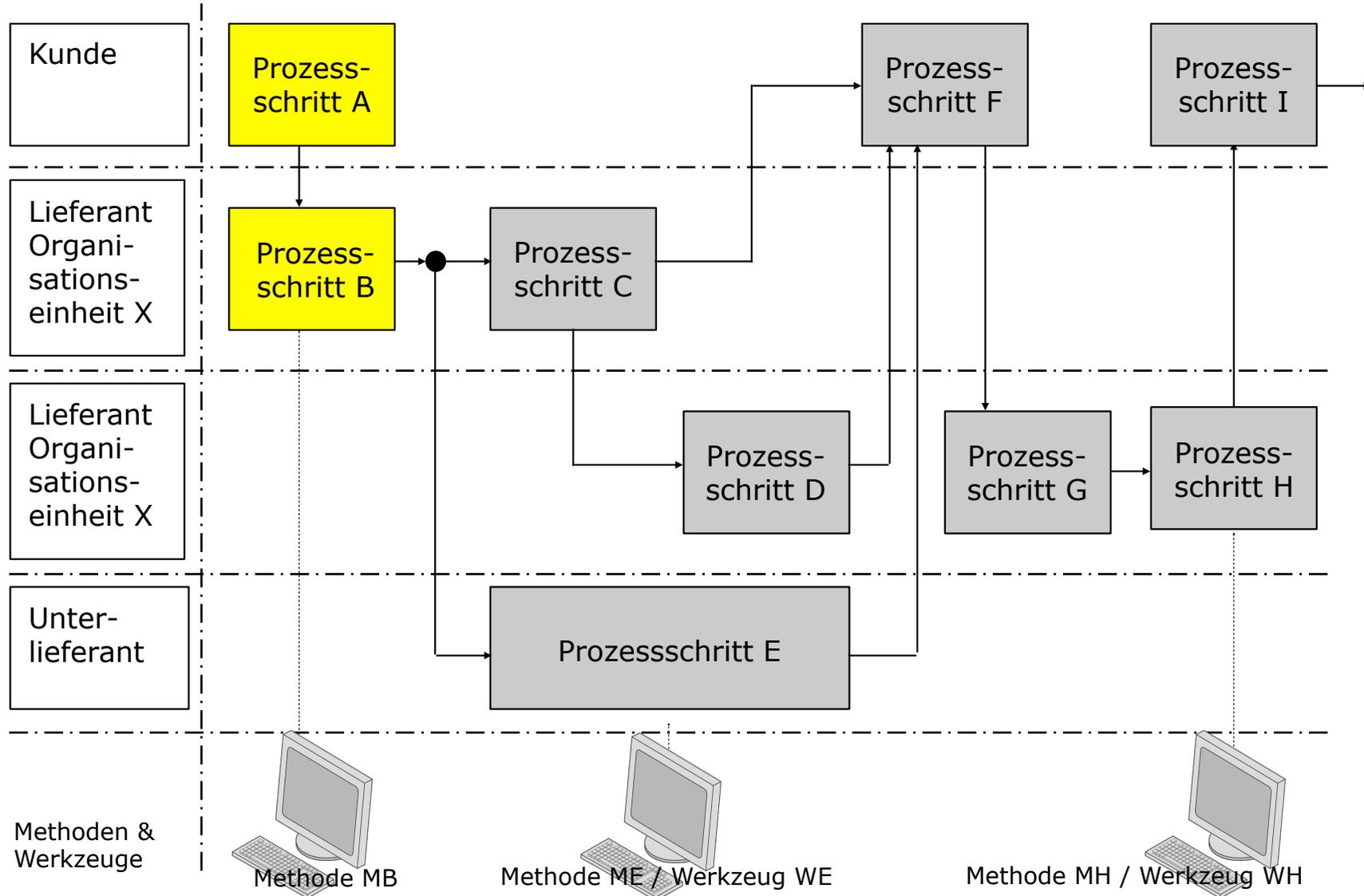


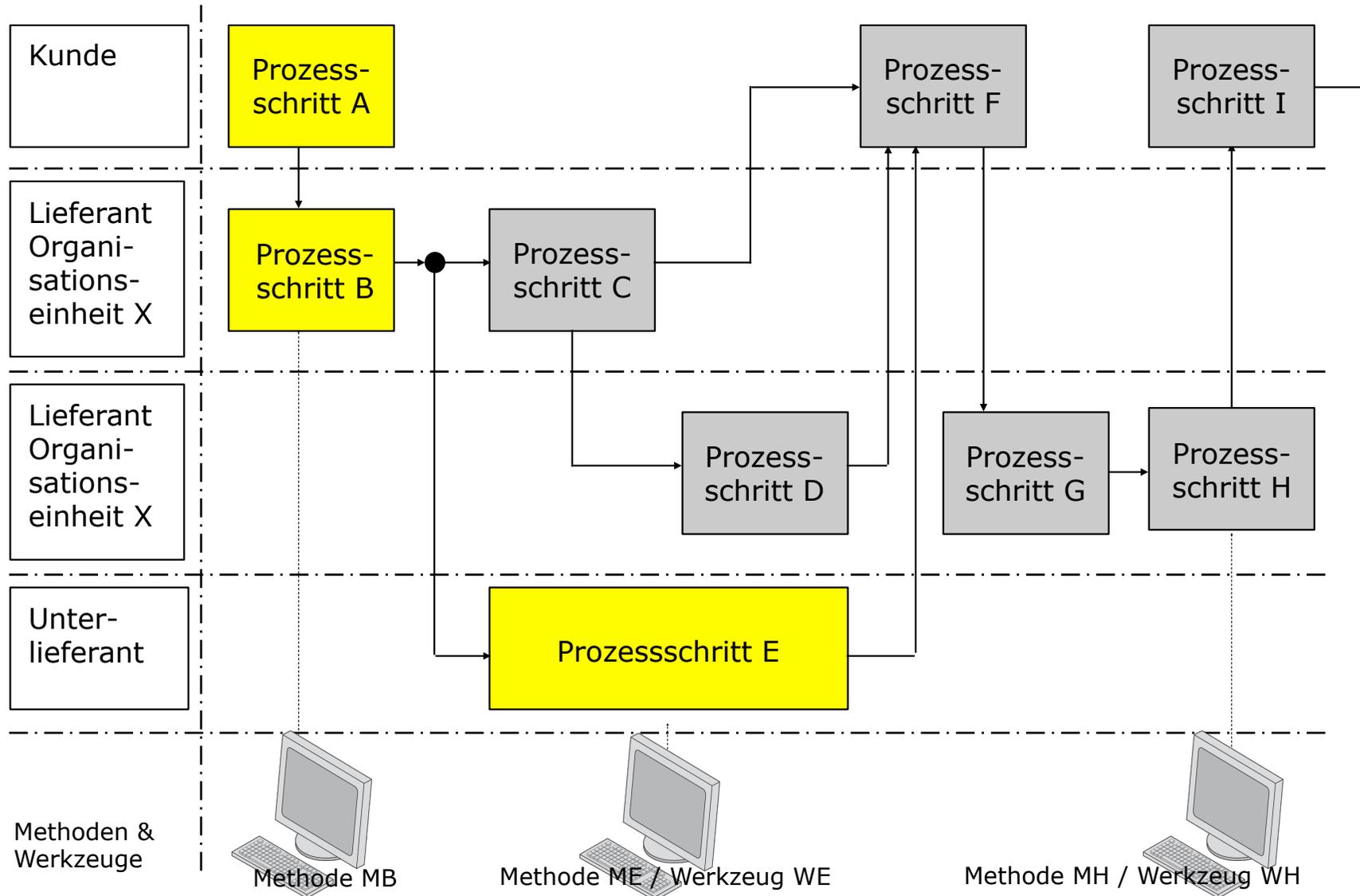


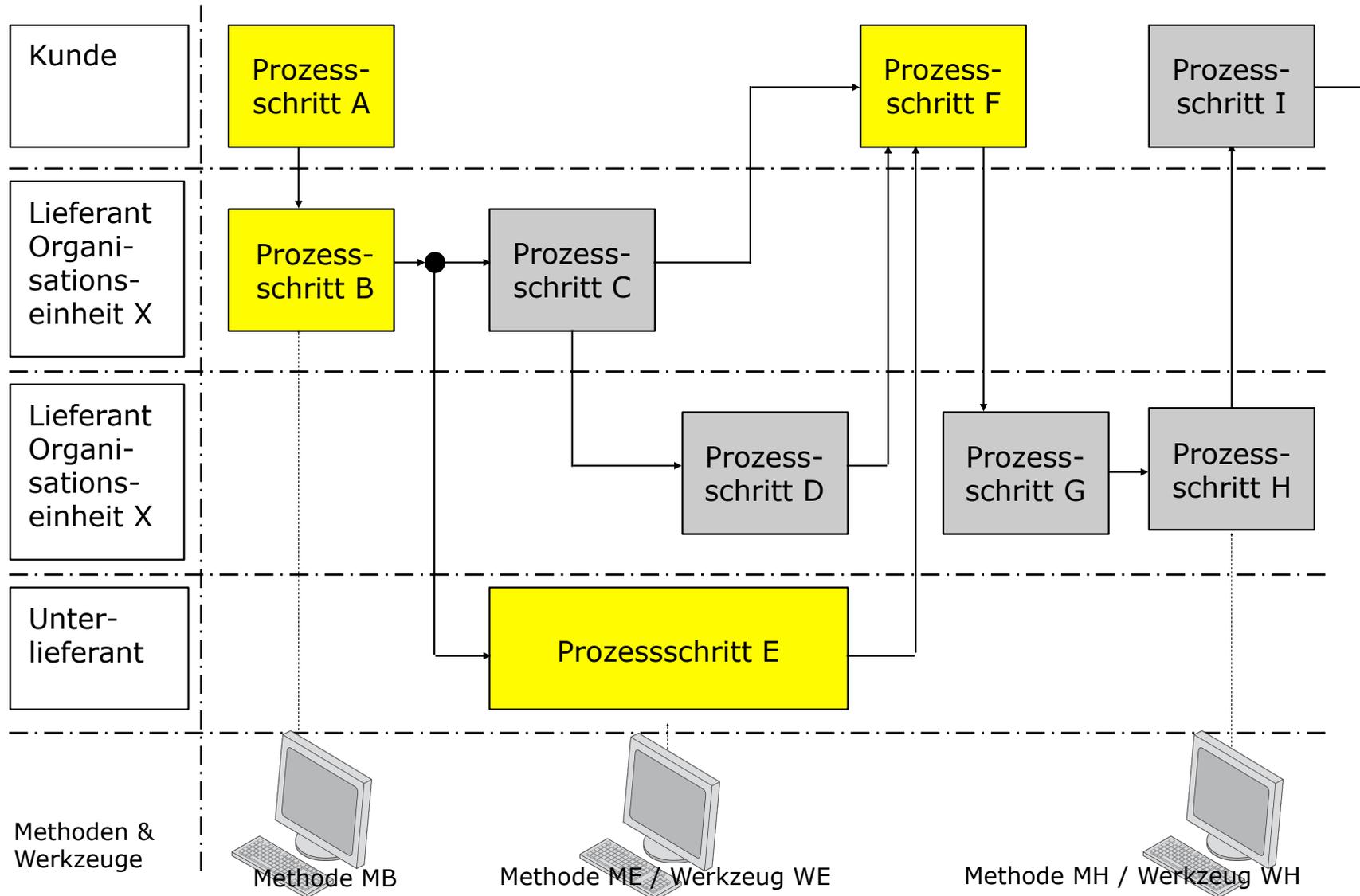


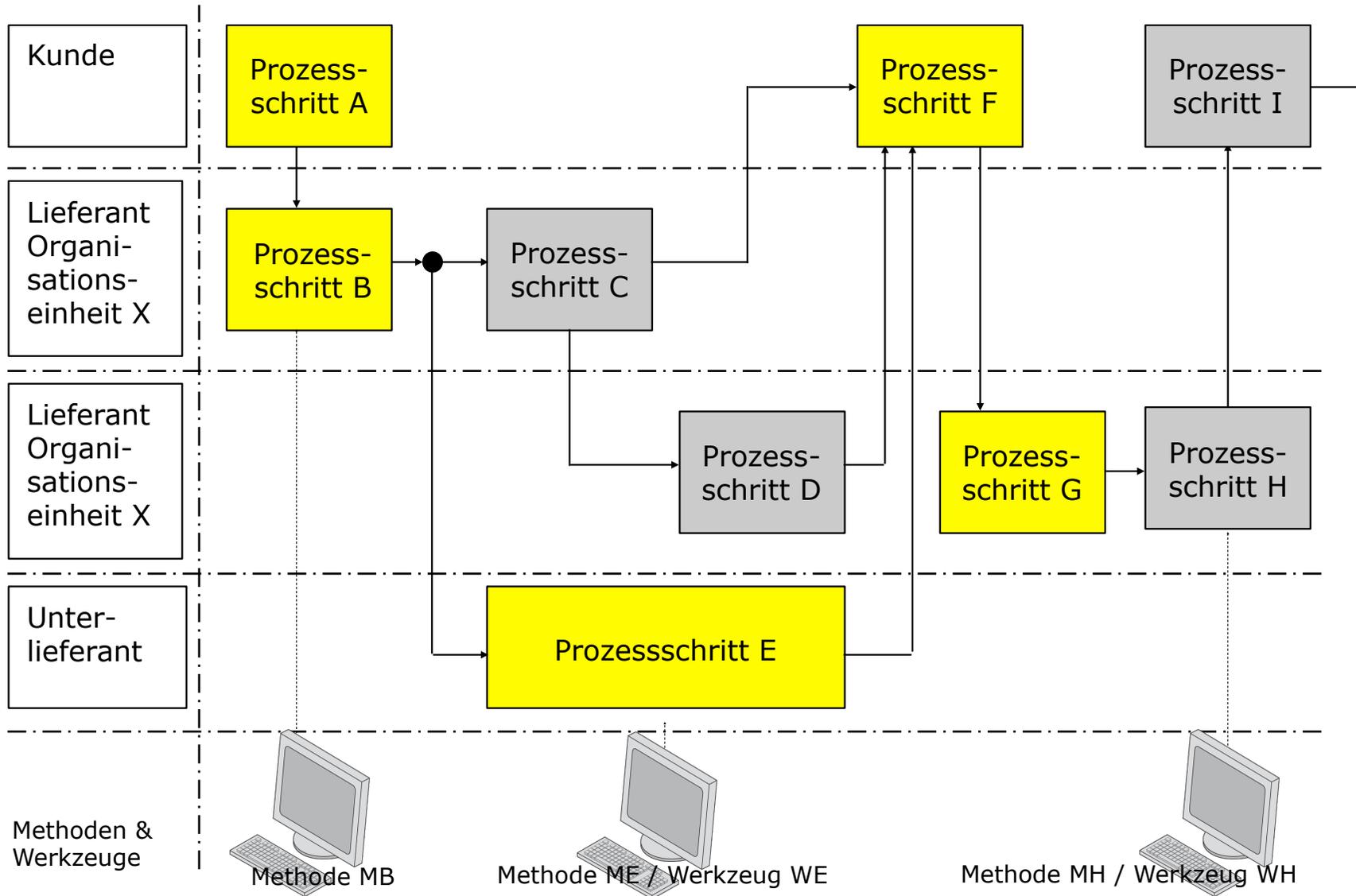


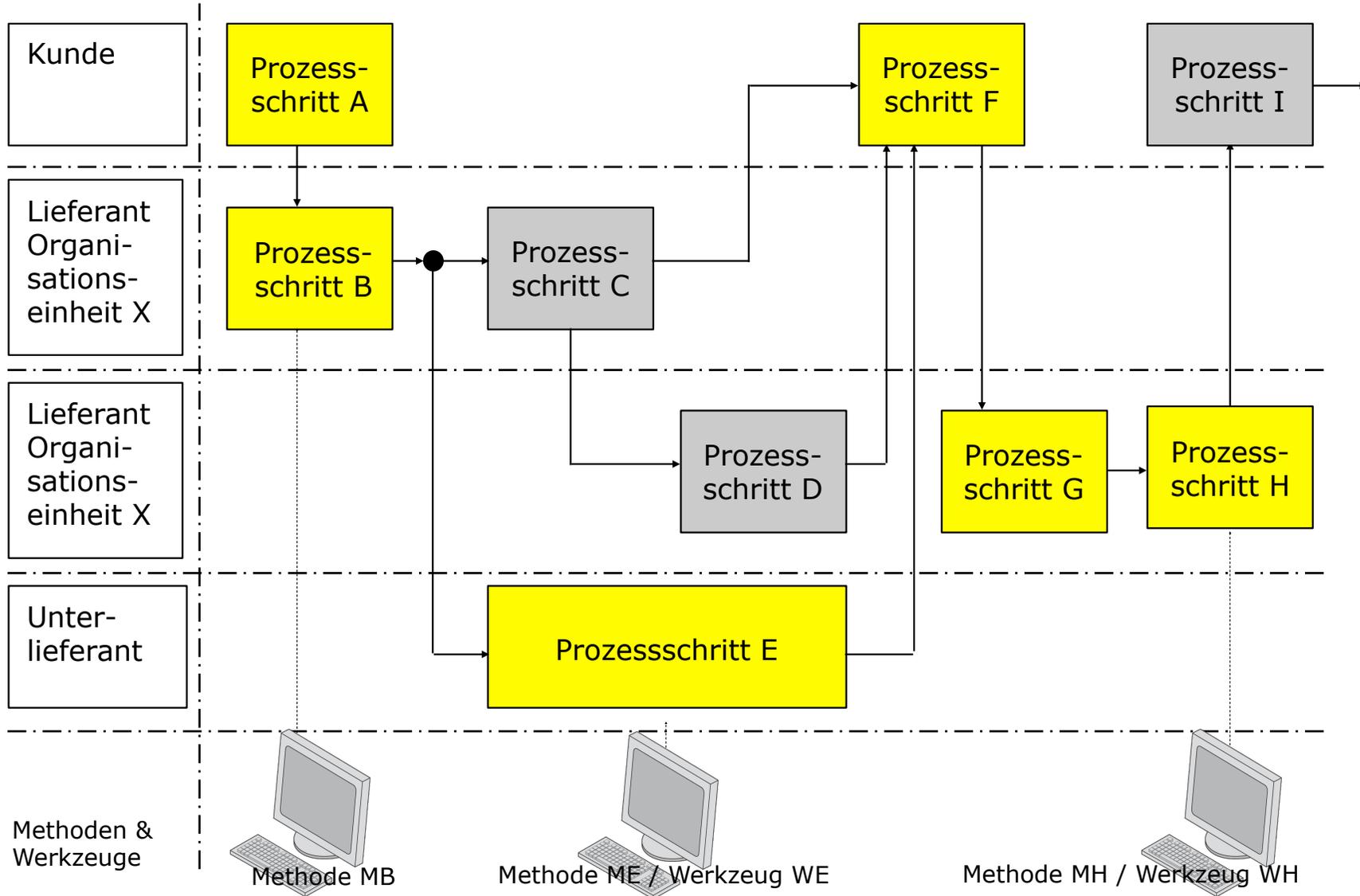


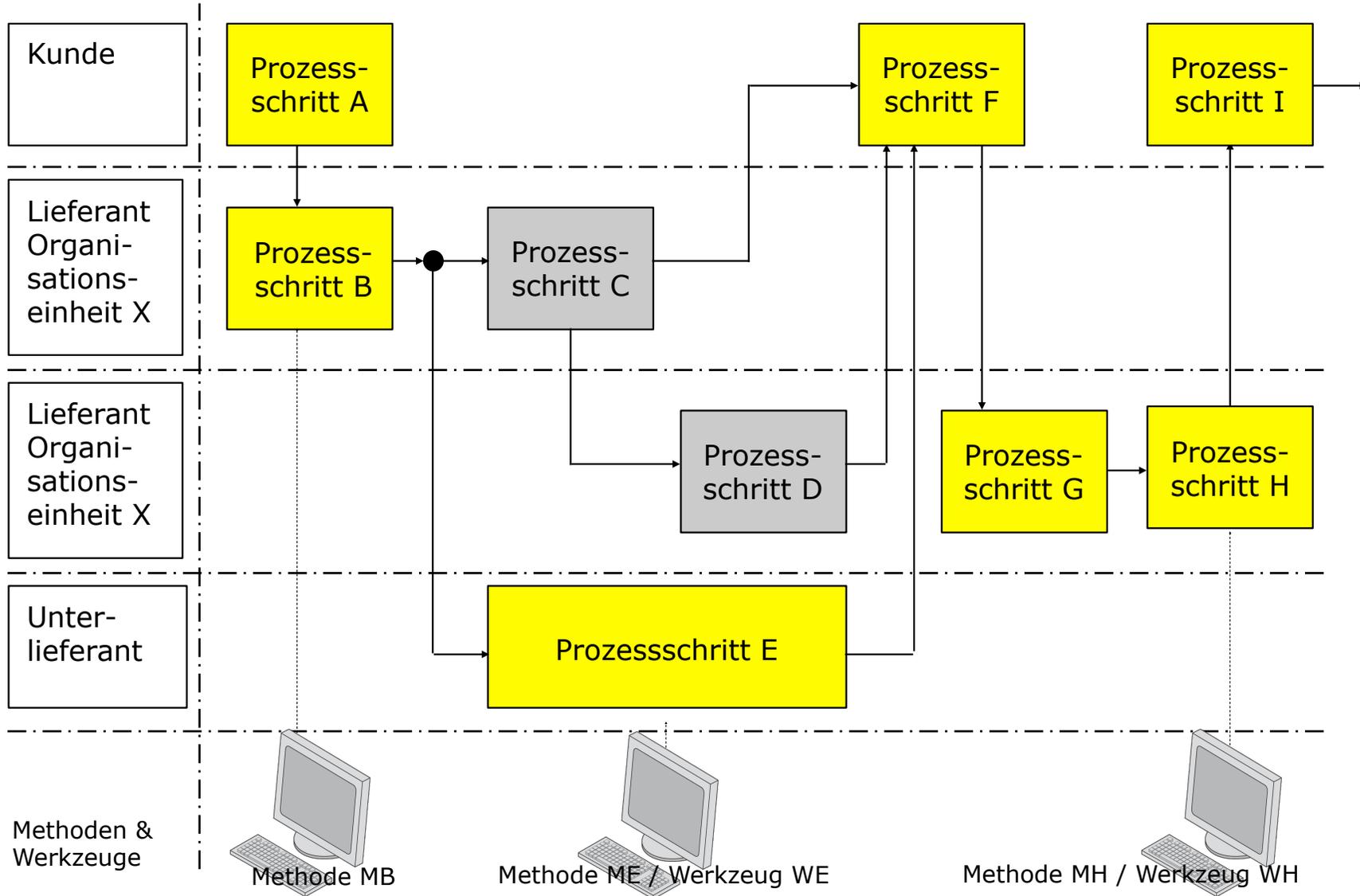












6. SW-Entwicklung / 3. Unterstützungsprozesse

Unterstützungsprozesse für die Embedded Software Entwicklung

1. Vorgehensmodelle und Standards
2. Konfigurationsmanagement
3. Projektmanagement
4. Lieferantenmanagement
5. **Anforderungsmanagement**
 1. **Erfassen von Benutzeranforderungen**
 2. Verfolgen von Anforderungen
6. Qualitätssicherung

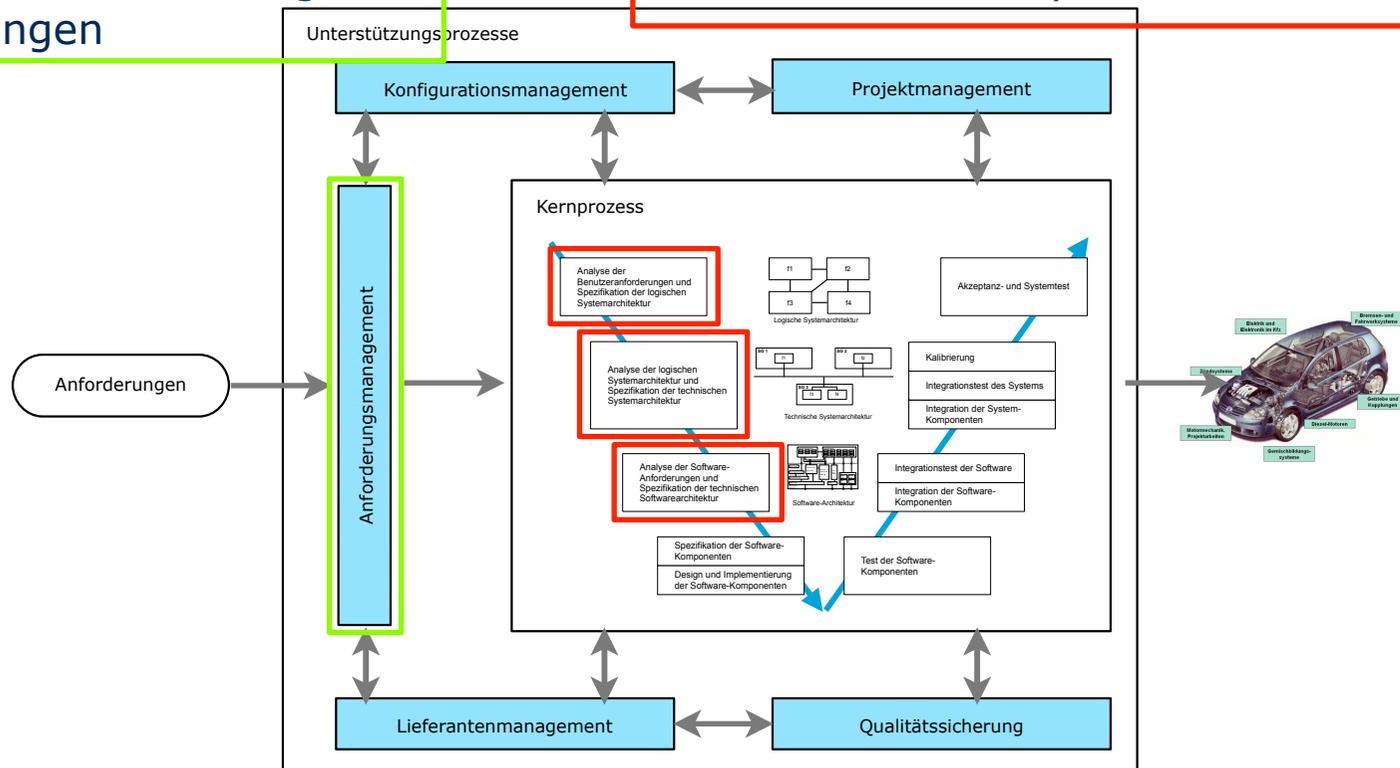
Anforderungsmanagement (1)

- Anforderungsmanagement ist - wie Konfigurationsmanagement - nicht grundsätzlich automobilspezifisch
 - Einsatz von Standardwerkzeugen wie Telelogic DOORS (Jetzt IBM)
 - <http://www-03.ibm.com/software/products/en/ratidoor/>
- Aber: Berücksichtigung von besonderen Anforderungen in Fahrzeugprojekten
 - Unterstützung der unternehmens- und standortübergreifenden Zusammenarbeit im Anforderungsmanagement
 - Versionierung von Anforderungen
 - Zusammenspiel von Anforderungsmanagement und Konfigurationsmanagement
 - Lange Produktlebenszyklen
 - Änderung der Anforderungen

Anforderungsmanagement (2)

- Unterstützungsprozess
Anforderungsmanagement
 - Erfassen von Benutzeranforderungen
 - Verfolgen der Umsetzung von Anforderungen

- Kernprozess der System- und Softwareentwicklung
 - Analyse der Anforderungen
 - Spezifikation der logischen und technischen Systemarchitektur



Erfassen von Benutzeranforderungen

- Benutzer sind alle Personen, deren Wünsche Einfluss auf das Fahrzeug haben können
- Benutzergruppen eines Fahrzeugs
 - Fahrer
 - Insassen
 - Andere Verkehrsteilnehmer
 - Fahrzeuge
 - Fahrer
 - Radfahrer
 - Reiter
 - Fussgänger
 - ...
 - ...
 - ...
 - Mitarbeiter im Service
- Gesetzgeber
 - Gesetzliche Vorgaben werden auch als Randbedingungen bezeichnet











Erfassen von Benutzeranforderungen

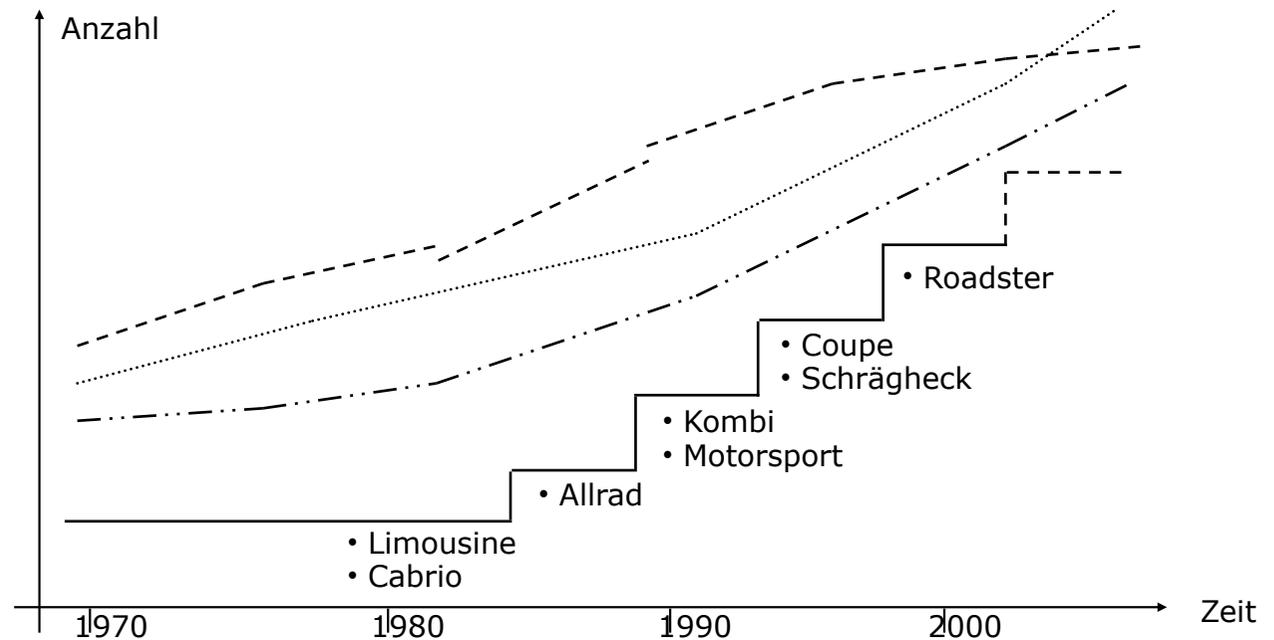
- Benutzer sind alle Personen, deren Wünsche Einfluss auf das Fahrzeug haben können
- Benutzergruppen eines Fahrzeugs
 - Fahrer
 - Insassen
 - Andere Verkehrsteilnehmer
 - Fahrzeuge
 - Fahrer
 - Radfahrer
 - Reiter
 - Fussgänger
 - Diebe, Vandalen
 - Marder
 - Insekten, Wild(-unfälle)
 - Mitfahrende Haustiere
- ...
- Mitarbeiter im Service
- Gesetzgeber
 - Gesetzliche Vorgaben werden auch als Randbedingungen bezeichnet

Drei Arten von Benutzeranforderungen

- Unterscheidung nach Zeitpunkt der Erfassung
 - Anforderungen, die zu Beginn des Projektes gestellt werden
 - Anforderungen, die im Laufe des Projektes gestellt werden
 - Änderungswünsche
 - Zusätzliche Anforderungen
 - Anforderungen, die nach Übergabe des Produktes gestellt werden
 - Neue Anforderungen
 - Fehlermeldungen
 - Verbesserungsvorschläge
- Erfassung von Benutzeranforderungen durch
 - Ableitung aus existierenden Systemen
 - Technische Entwicklungen
 - Rückmeldungen
 - Interviews
 - Workshops
 - ...

Anzahl der Benutzeranforderungen

- Zunahme mit jeder Fahrzeuggeneration
 - Gestiegene Anzahl der Fahrzeugfunktionen
 - Zunahme der Fahrzeugvarianten
 - Gestiegene Kundenerwartungen
 - Gestiegene Gesetzliche Anforderungen



Legende:	----	Anzahl gesetzlicher Anforderungen
	Optionale Funktionen der Fahrzeuge
	- . - .	Produzierte Fahrzeuge pro Jahr
	—	Karosserievarianten + Motorvarianten + Getriebevarianten

Die Zunahme an Variantenvielfalt auf Produktebene schlägt sich direkt auf die Teileebene durch

Evolution der Produktvarianten



1983 Mercedes 190
1 Karosserievariante
4 Motoren



1993 Mercedes C-Class
4 Karosserievariante
7 Motoren
4 Designlinien



2000 Mercedes C-Class
5 Karosserievariante
7 Motoren
3 Designlinien



2007 Mercedes C-Class
7 Karosserievariante
9 Motoren
3 Designlinien



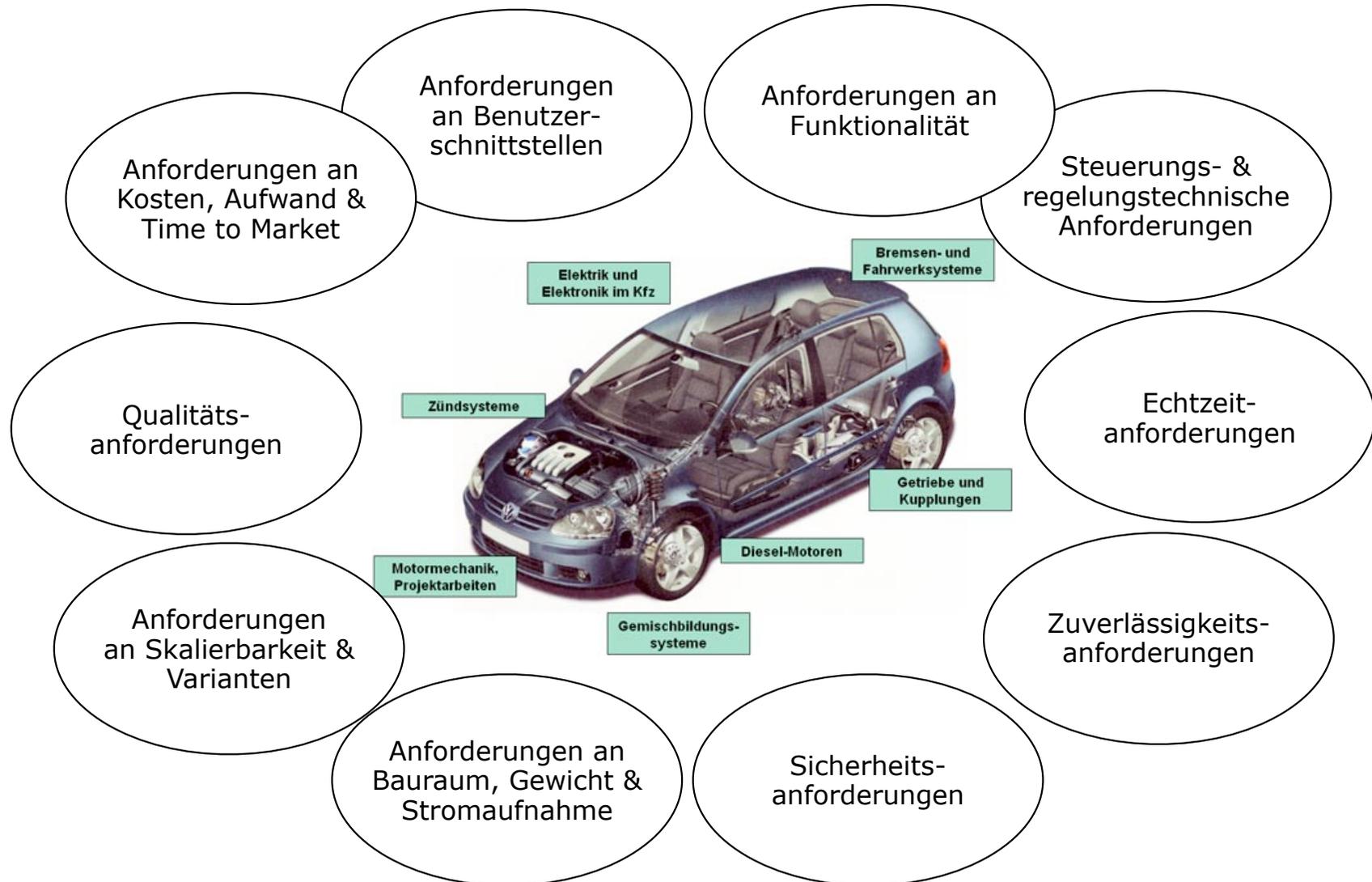
- 80 Sonderausstattungen
- Wiederholrate identische Fahrzeuge: 1,4 p.a.
- 20 000 Zukaufteile
- 12 000 selbstergestellte Teile
- 75% der Teile sind für Sonderausstattungen

Quelle: Katzenbach, Daimler AG

G. Rock 04/06/09 – Seite/Page 5

Quelle: Dr. Georg Rock, PROSTEP IMP GmbH:
Variantenmanagement: Forschung und industrieller Einsatz
Darmstädter „Automotive Software Engineering“-Kolloquium 2009

Anforderungsklassen



6. SW-Entwicklung / 3. Unterstützungsprozesse

Unterstützungsprozesse für die Embedded Software Entwicklung

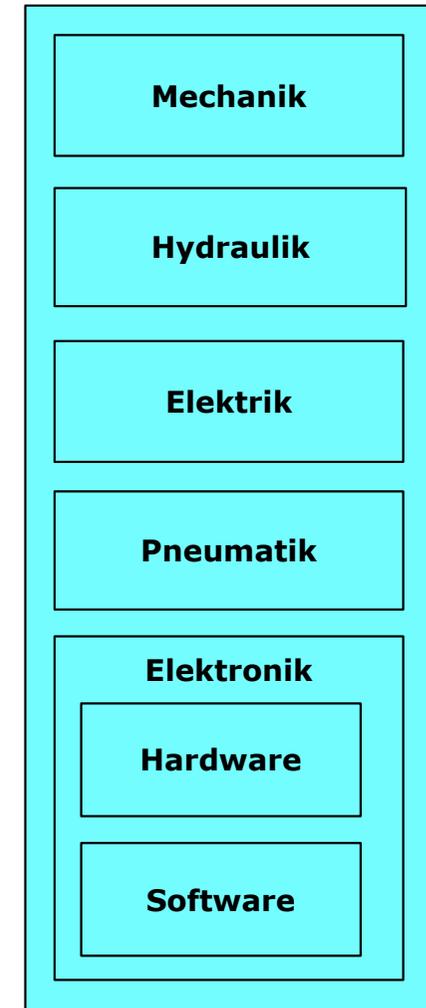
1. Vorgehensmodelle und Standards
2. Konfigurationsmanagement
3. Projektmanagement
4. Lieferantenmanagement
5. **Anforderungsmanagement**
 1. Erfassen von Benutzeranforderungen
 2. **Verfolgen von Anforderungen**
6. Qualitätssicherung



Benutzeranforderungen



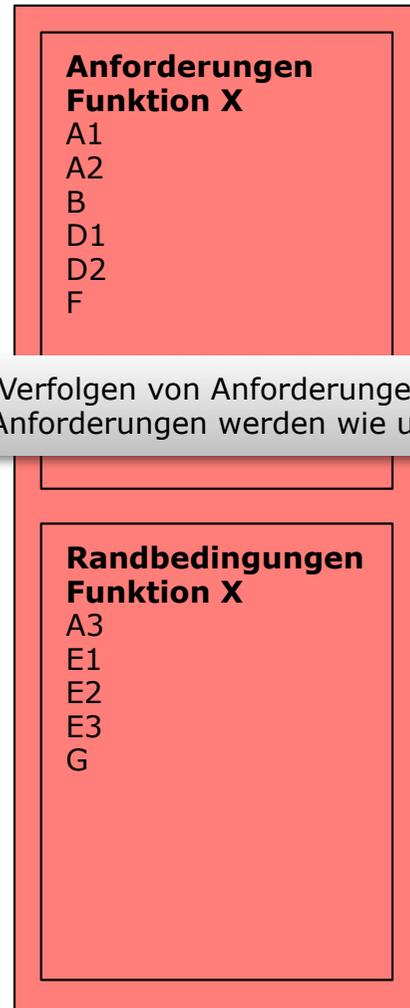
Logische Systemarchitektur



Technische Systemarchitektur

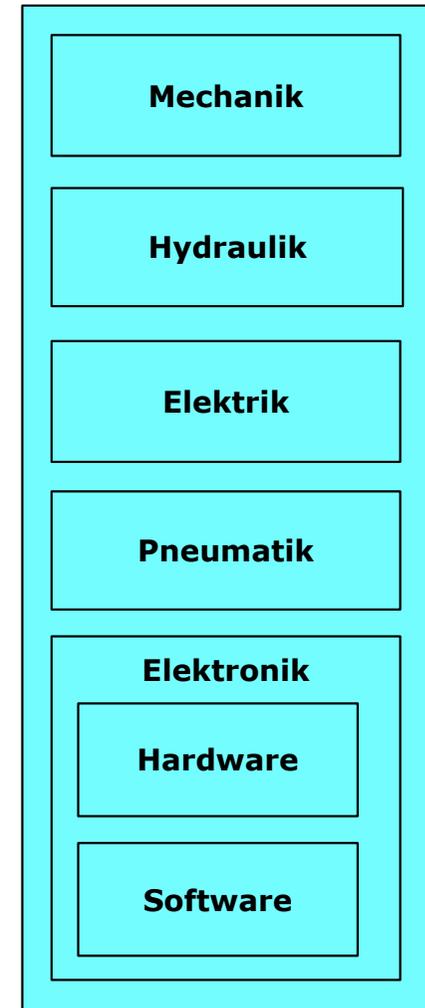


Benutzeranforderungen

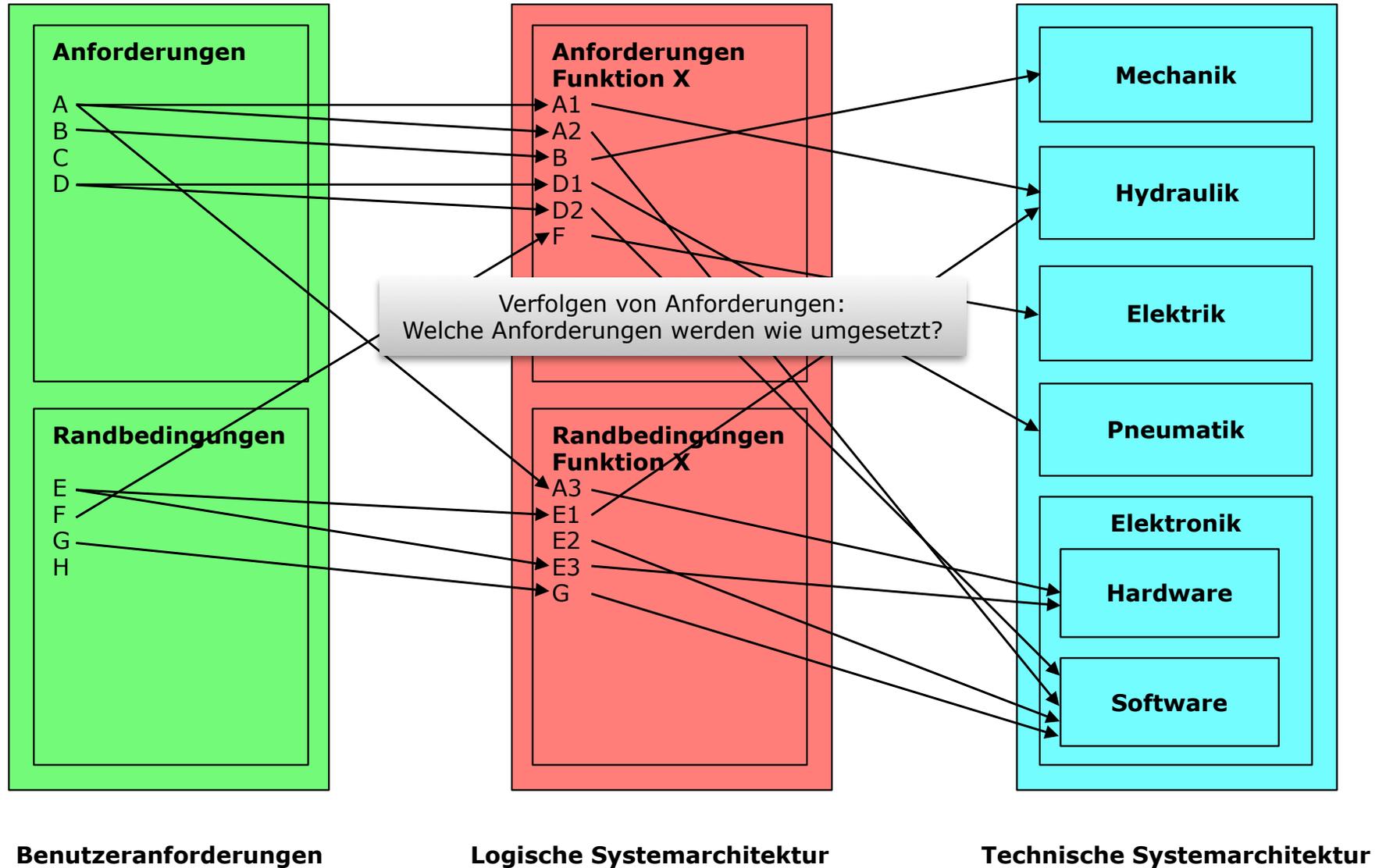


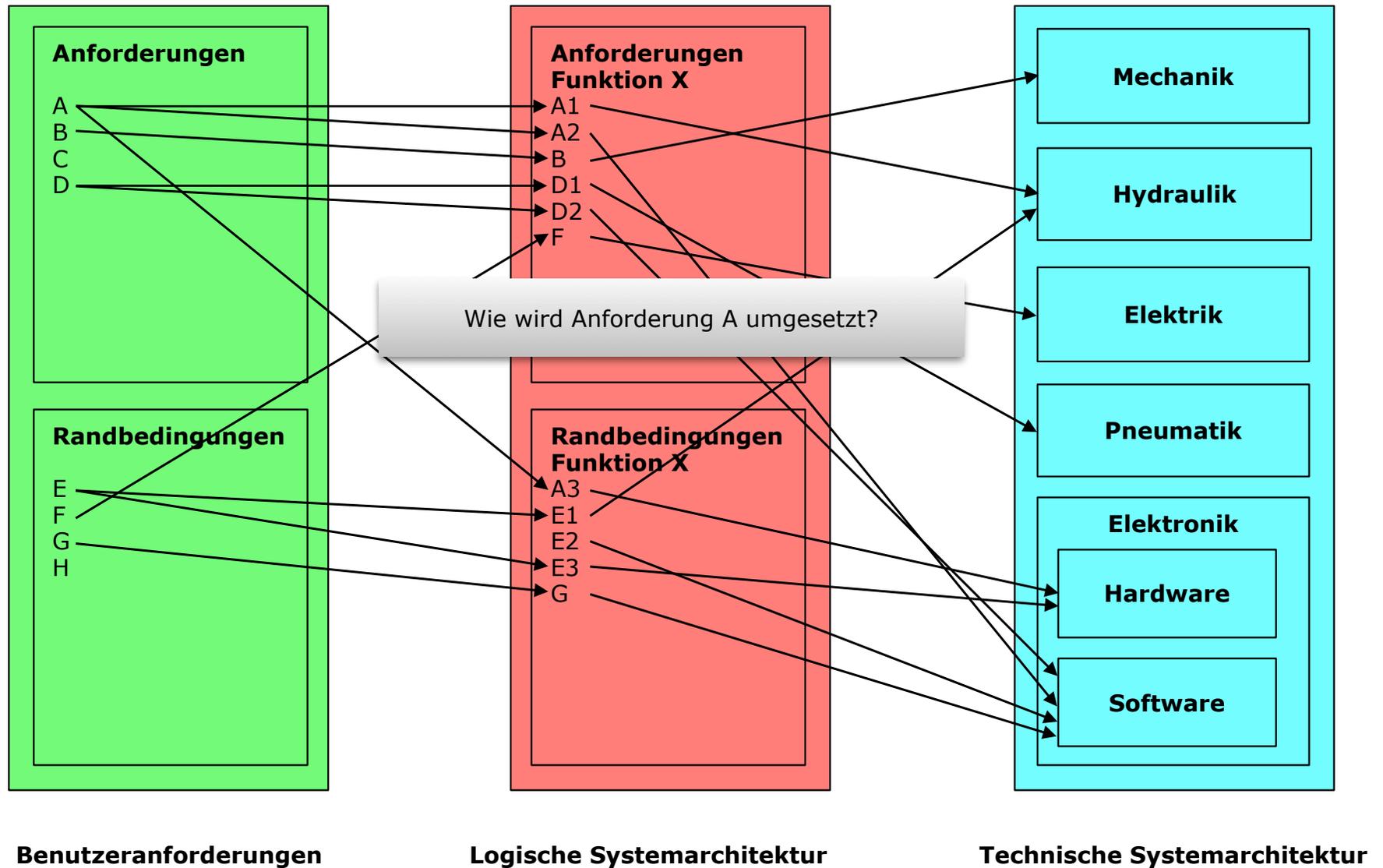
Logische Systemarchitektur

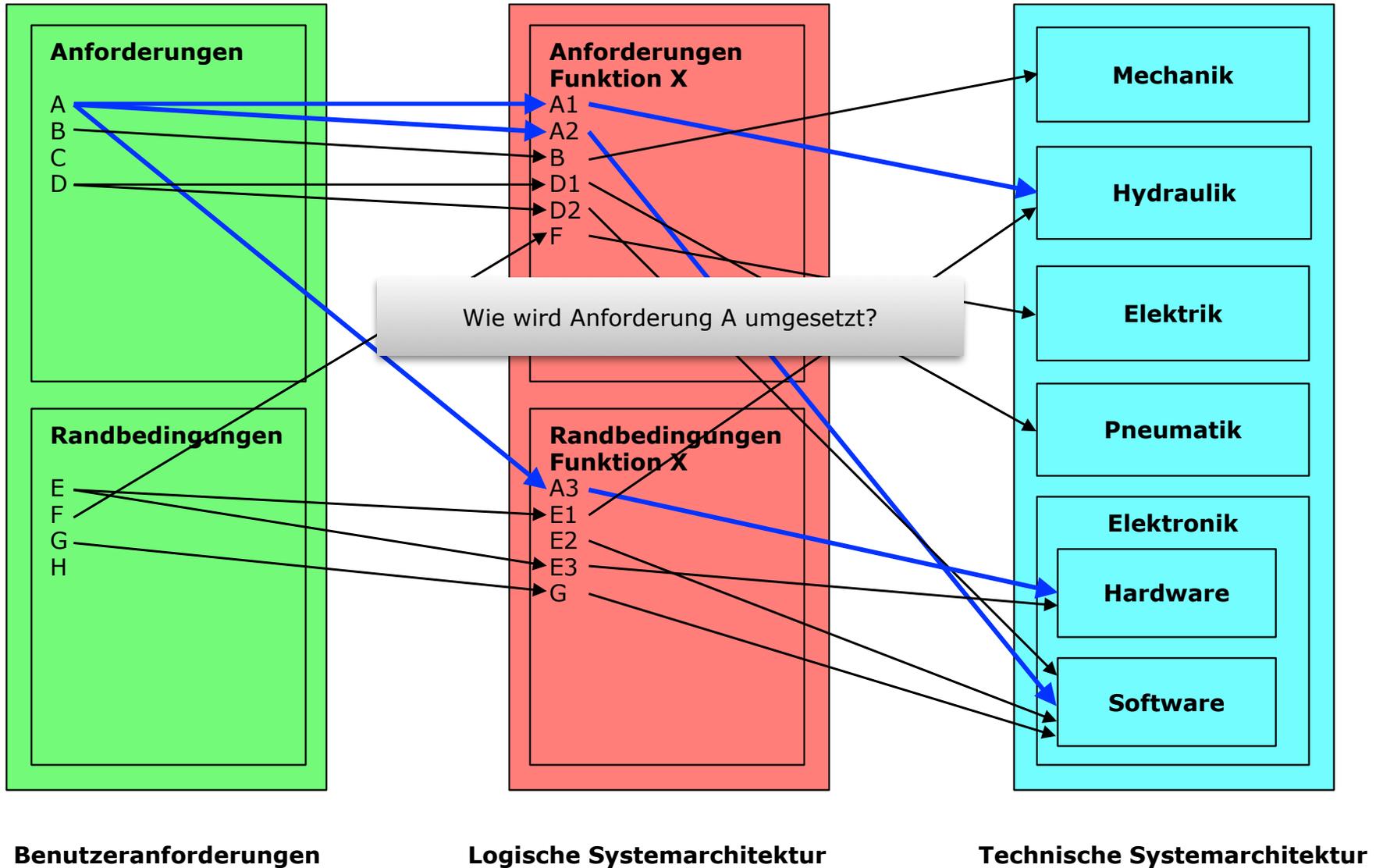
Verfolgen von Anforderungen:
Welche Anforderungen werden wie umgesetzt?

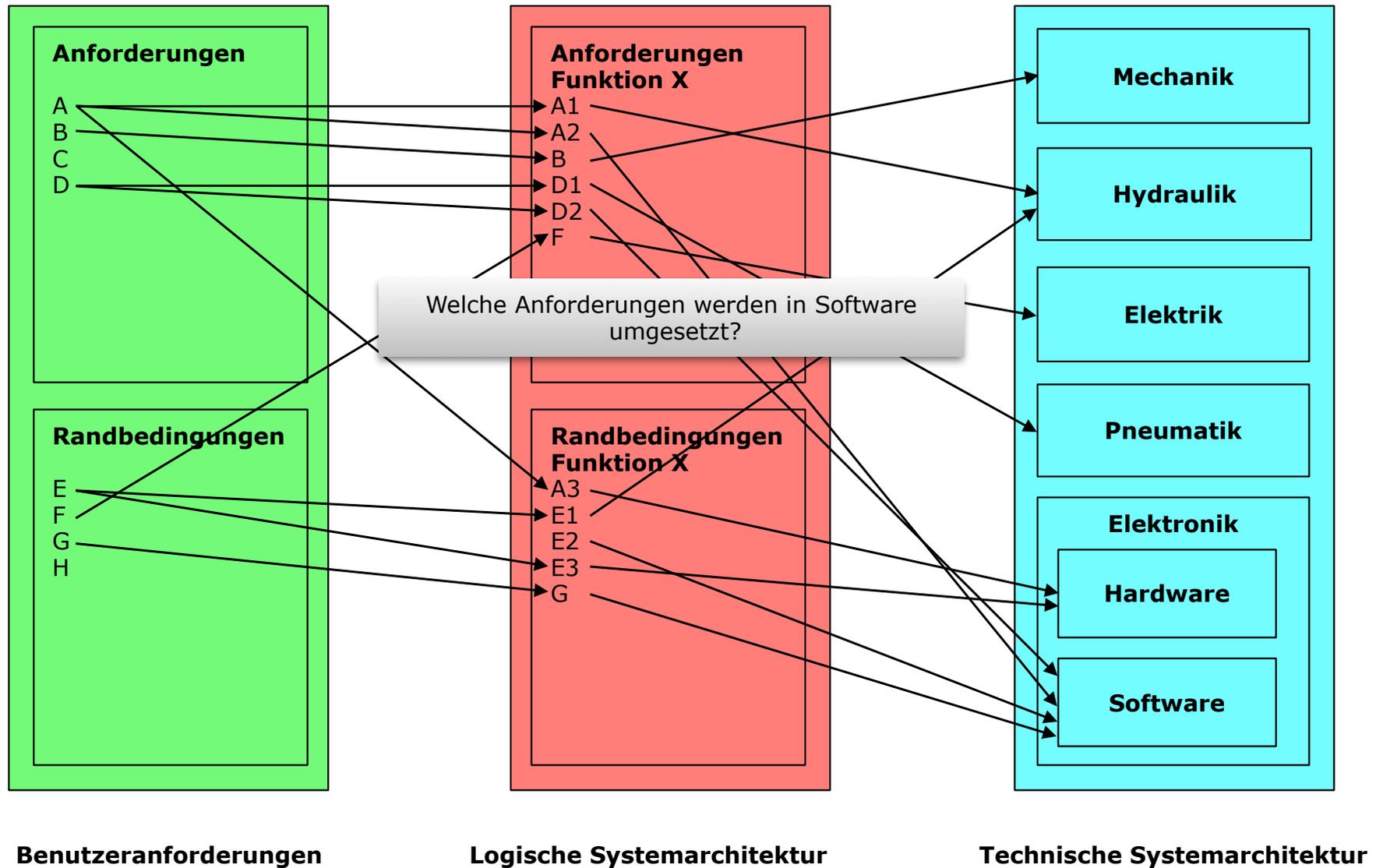


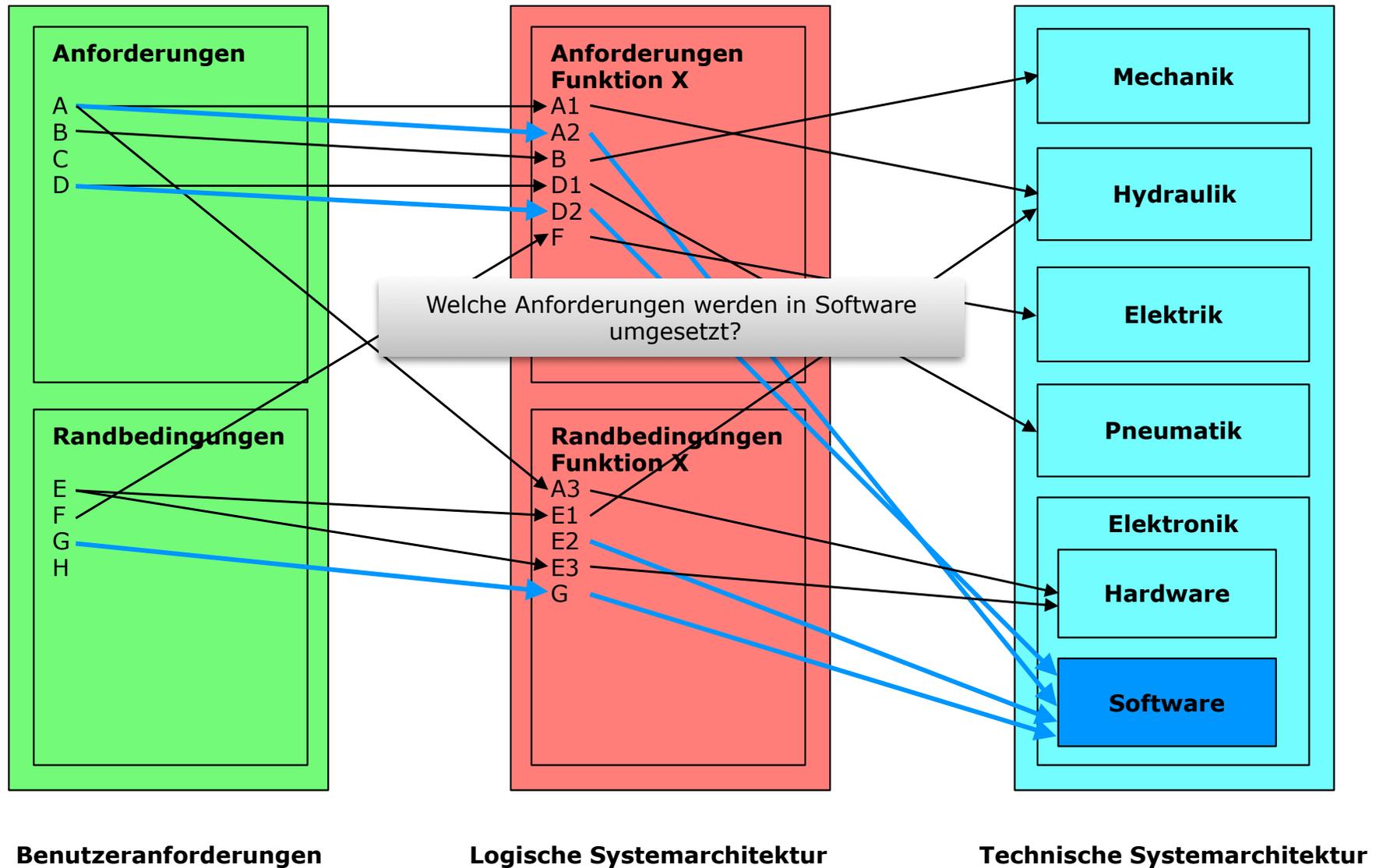
Technische Systemarchitektur

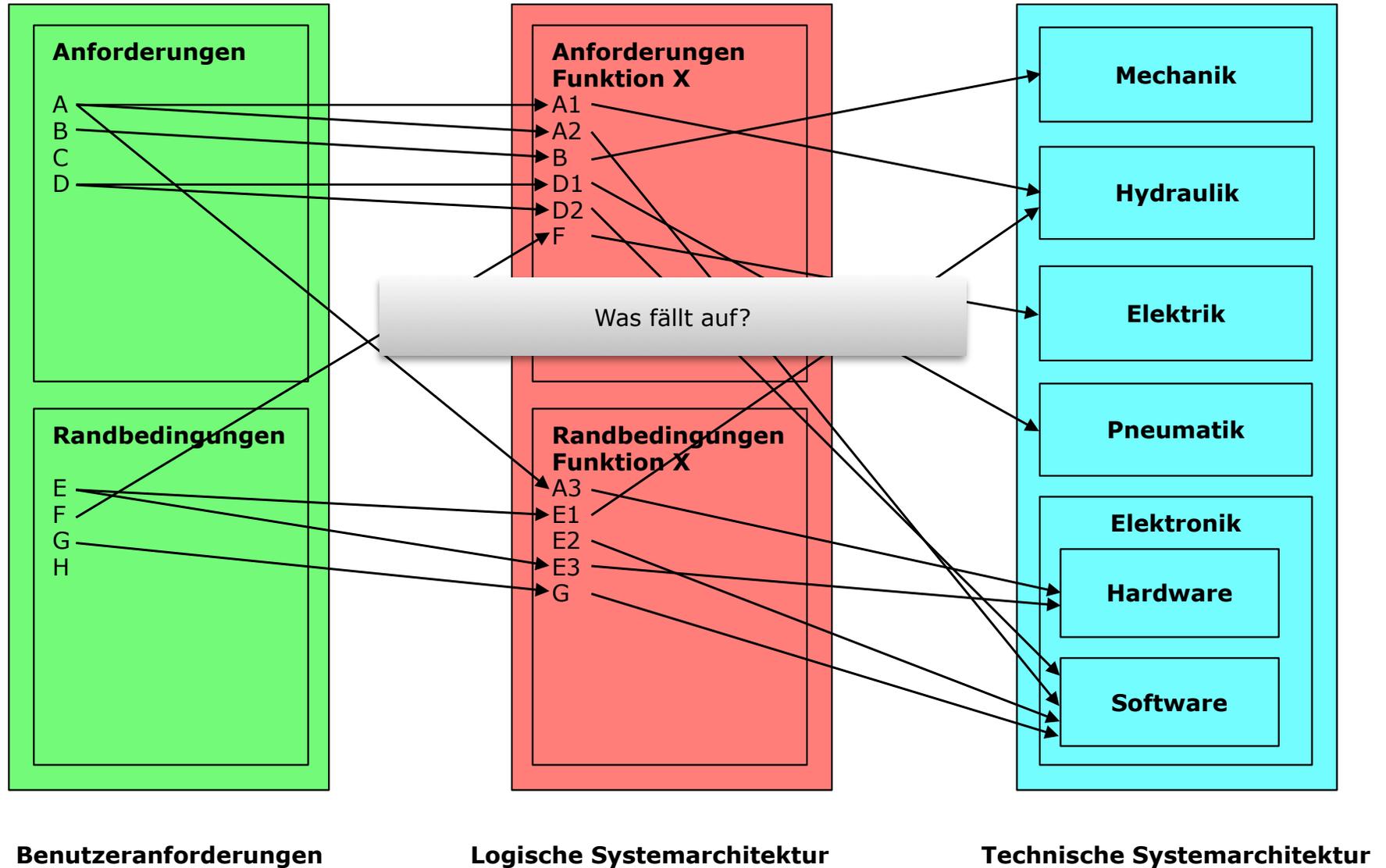


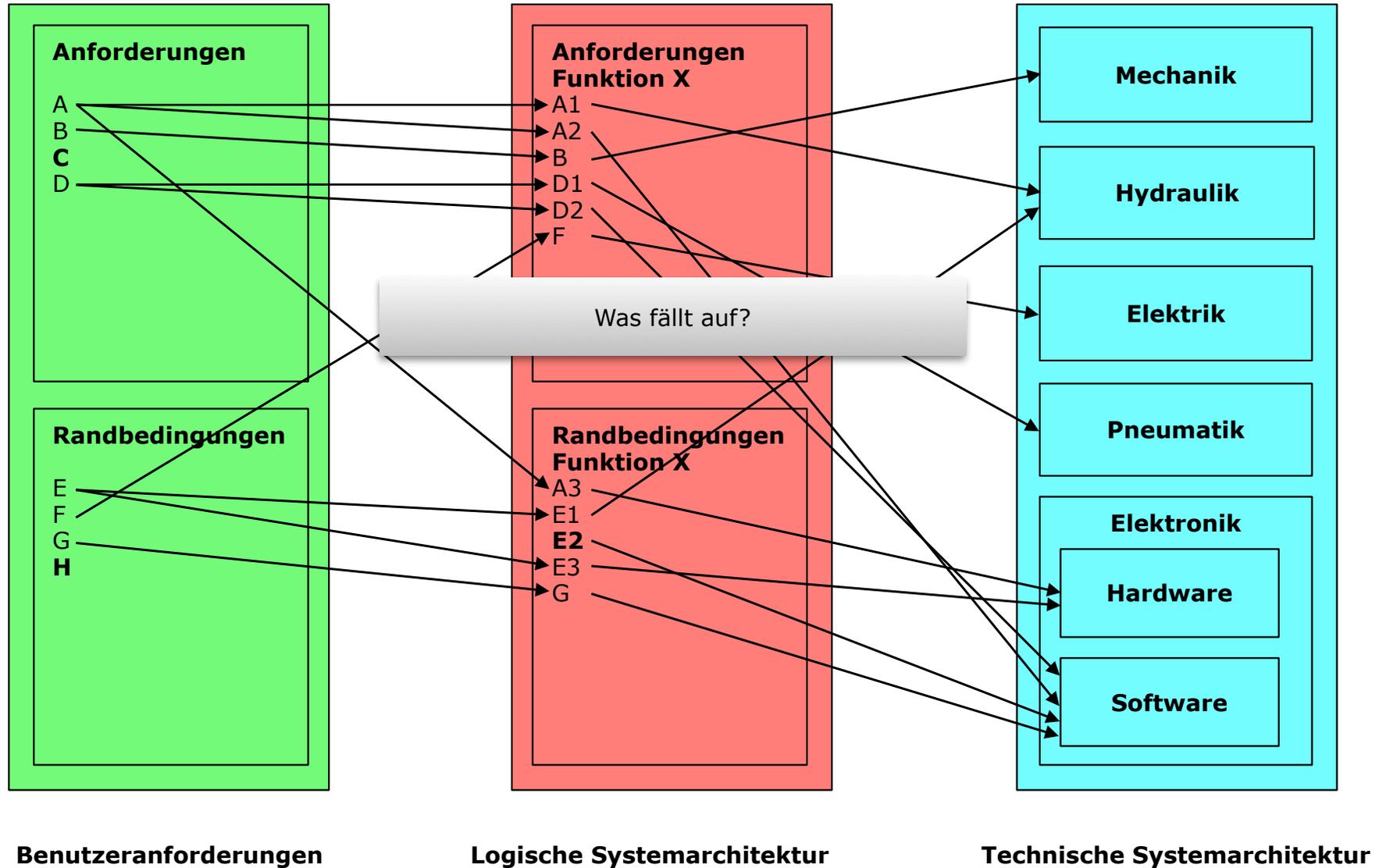








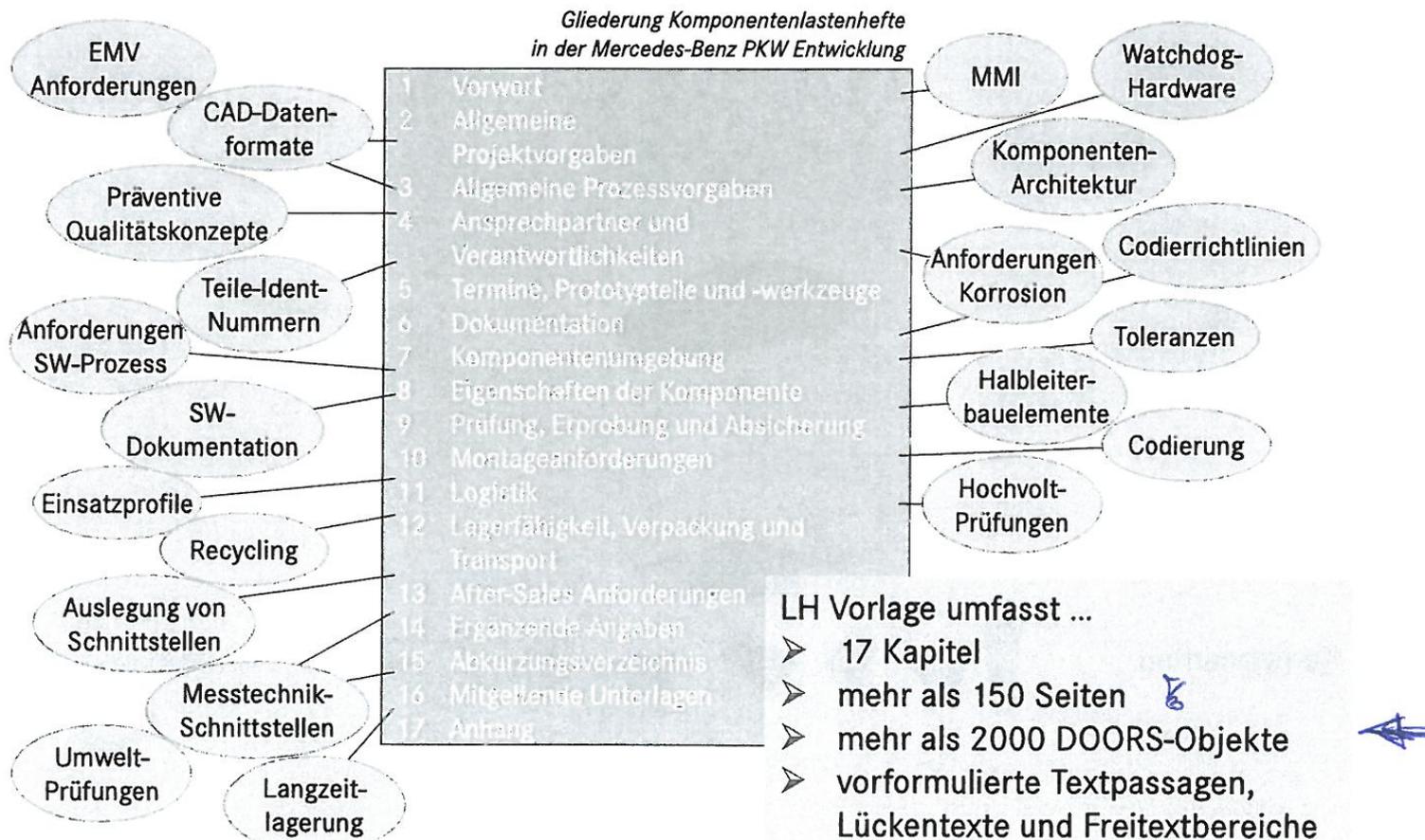




Wie sieht es in der Praxis aus?

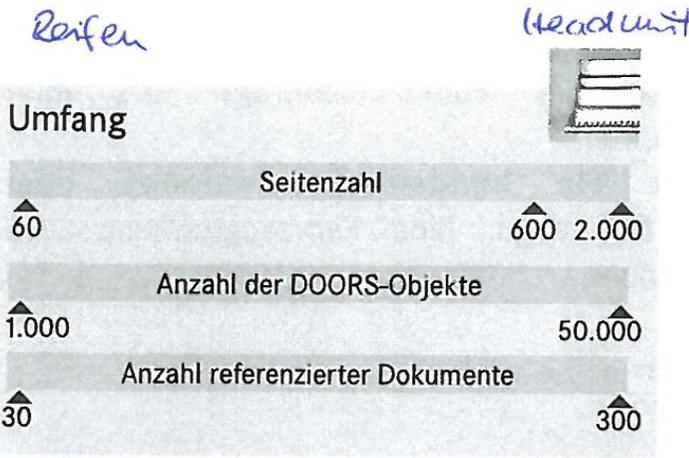
- Quelle:
Durchgängiges Anforderungsmanagement vom Fahrzeug bis zur Komponente,
Dr. Matthias Recknagel, Daimler AG,
Ringvorlesung: Forum Software und Automatisierung,
IAS, Universität Stuttgart, 9. Januar 2014

Komponentenlastenhefte – Aufbau und Inhalt



*Standards Recycling Mercedes-Benz
gesetz, Material, Kreislaufspflicht
z.B. für Metalle 5 Jahre 25 Mio€*

Zahlen, Daten, Fakten Lastenhefte @ Mercedes-Benz



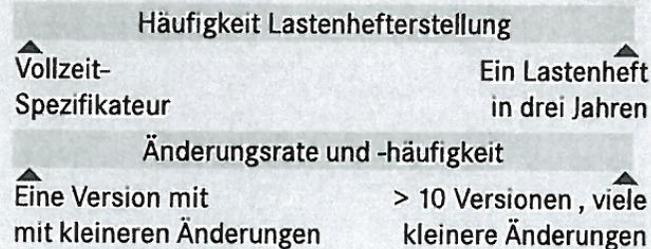
Lastenheft-Vorlage

- 17 Kapitel
- > 150 Seiten
- > 2000 DOORS Objekte
- Viele Standard-Formulierungen

1	Vorwort
2	Allgemeine Projektvorgaben
3	Allgemeine Prozessvorgaben
4	Ansprechpartner und Verantwortlichkeiten
5	Termine, Prototypen und -werkzeuge
6	Dokumentation
7	Komponentenumgebung
8	Eigenschaften der Komponente
9	Prüfung, Erprobung und Absicherung
10	Montageanforderungen
11	Logistik
12	Lagerfähigkeit, Verpackung und Transport
13	After-Sales Anforderungen
14	Ergänzende Angaben
15	Abkürzungsverzeichnis
16	Mitgeltende Unterlagen
17	Anhang

40.000 Seiten relevant worst case

Häufigkeit



Spezifikationsinhalte

- Überwiegend natürlichsprachlich (German, English)
- Tabellen und Zeichnungen nach Bedarf
- Formale Spezifikation bei
 - ausführbaren Modellen (Matlab/Simulink)
 - CAD Zeichnungen

6. SW-Entwicklung / 3. Unterstützungsprozesse

Unterstützungsprozesse für die Embedded Software Entwicklung

1. Vorgehensmodelle und Standards
2. Konfigurationsmanagement
3. Projektmanagement
4. Lieferantenmanagement
5. Anforderungsmanagement
6. **Qualitätssicherung**
 1. **Integrations- und Testschritte**
 2. Maßnahmen zur Qualitätssicherung von Software
 3. Qualitätsstandards

Qualitätssicherung

- Qualitätssicherung
 - Alle Maßnahmen, die sicherstellen, dass das Produkt die geforderten Anforderungen erfüllt.
- Richtlinien zur Qualitätssicherung: Vorbeugende Maßnahmen
 - Einsatz von erfahrenen und geschulten Mitarbeitern
 - Geeigneter Entwicklungsprozess mit definierten Testschritten
 - Richtlinien, Maßnahmen und Standards zur Unterstützung des Prozesses
 - Werkzeugumgebung zur Unterstützung des Prozesses
 - Automatisierung manueller und fehlerträchtiger Arbeitsschritte
- Maßnahmen zur Qualitätssicherung: Finden von Fehlern
 - Nach jedem Schritt des Entwicklungsprozesses
 - Ins V-Modell integriert, siehe „Kernprozess“
- Software-Produkte
 - Spezifikationsfehler
 - Implementierungsfehler
 - Ergebnis von zahlreichen Untersuchungen:
 - In den meisten Projekten überwiegen die Spezifikationsfehler

Validation und Verifikation

- Validation: Prüfen auf Spezifikationsfehler
- Validation ist der Prozess zur Beurteilung eines Systems oder einer Komponente mit dem Ziel festzustellen, ob der Einsatzzweck oder die Benutzererwartungen erfüllt werden. Funktionsvalidation ist demnach die Prüfung, ob die Spezifikation die Benutzeranforderungen erfüllt, ob überhaupt die Benutzerakzeptanz durch eine Funktion erreicht wird.
- Verifikation: Prüfen auf Implementierungsfehler
- Verifikation ist der Prozess zur Beurteilung eines Systems oder einer Komponente mit dem Ziel festzustellen, ob die Resultate einer gegebenen Entwicklungsphase den Vorgaben für diese Phase entsprechen. Software-Verifikation ist demnach die Prüfung, ob eine Implementierung der für den betreffenden Entwicklungsschritt vorgegebenen Spezifikation genügt.

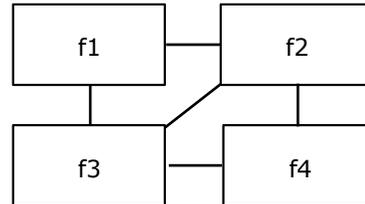
Verifikation und Validation

- Verifikation:
 - „Does the system things right?“ / „Erfüllt das System seine Aufgabe richtig?“
 - Prüfung z.B. gegen Technische Anforderungen
 - Beispiel: Analyse des Zeitverhaltens durch Untersuchung der Worst Case Execution Time (WCET)
- Validation:
 - „Does the system the right things?“ / „Erfüllt das System die richtige Aufgabe?“
 - Prüfung z.B. gegen Benutzeranforderungen
 - Beispiel: Simulation der Benutzeroberfläche
- Ähnlich:
 - Effektiv: Die richtigen Dinge tun.
 - Effizient: Die Dinge richtig tun.

Integrations- und Testschritte

- Das V-Modell unterscheidet fünf verschiedene Testschritte:
 1. Beim Komponententest wird eine Komponente gegen die Spezifikation der Komponente getestet.
 2. Beim Integrationstest der Software wird die Software gegen die Spezifikation der technischen Softwarearchitektur getestet
 3. Beim Integrationstest wird das System gegen die Spezifikation der technischen Systemarchitektur getestet.
 4. Beim Systemtest wird das System gegen die Spezifikation der logischen Systemarchitektur getestet.
 5. Beim Akzeptanztest wird das System gegen die Benutzeranforderungen getestet.
- Validation und Verifikation

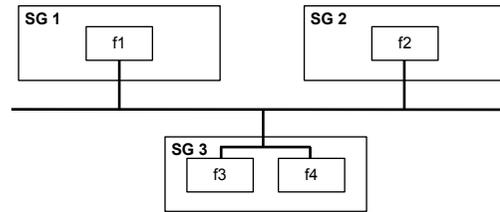
Analyse der Benutzeranforderungen und Spezifikation der logischen Systemarchitektur



Logische Systemarchitektur

Akzeptanz- und Systemtest

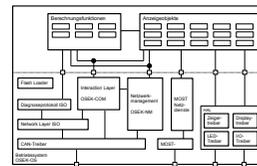
Analyse der logischen Systemarchitektur und Spezifikation der technischen Systemarchitektur



Technische Systemarchitektur

Kalibrierung
Integrationstest des Systems
Integration der System-Komponenten

Analyse der Software-Anforderungen und Spezifikation der technischen Softwarearchitektur



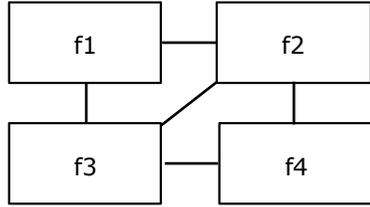
Software-Architektur

Integrationstest der Software
Integration der Software-Komponenten

Spezifikation der Software-Komponenten
Design und Implementierung der Software-Komponenten

Test der Software-Komponenten

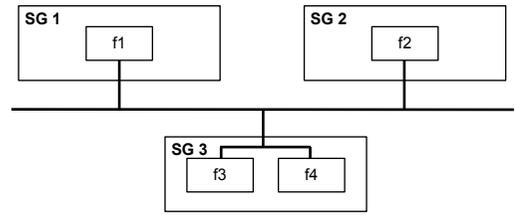
Analyse der Benutzeranforderungen und Spezifikation der logischen Systemarchitektur



Logische Systemarchitektur

Akzeptanz- und Systemtest

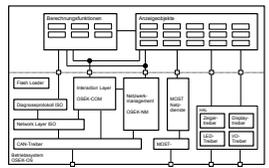
Analyse der logischen Systemarchitektur und Spezifikation der technischen Systemarchitektur



Technische Systemarchitektur

Kalibrierung
 Integrationstest des Systems
 Integration der System-Komponenten

Analyse der Software-Anforderungen und Spezifikation der technischen Softwarearchitektur



Software-Architektur

Integrationstest der Software
 Integration der Software-Komponenten

Spezifikation der Software-Komponenten
 Design und Implementierung der Software-Komponenten

Test der Software-Komponenten

6. SW-Entwicklung / 3. Unterstützungsprozesse

Unterstützungsprozesse für die Embedded Software Entwicklung

1. Vorgehensmodelle und Standards
2. Konfigurationsmanagement
3. Projektmanagement
4. Lieferantenmanagement
5. Anforderungsmanagement
6. **Qualitätssicherung**
 1. Integrations- und Testschritte
 2. **Maßnahmen zur Qualitätssicherung von Software**
 3. Qualitätsstandards

Motivation

- Tests können die Anwesenheit von Fehlern zeigen, nie aber deren Abwesenheit.
 - Edsger W. Dijkstra
Ein fehlender Programmzweig ... wird auch durch Austesten aller vorhandenen Programmzweige nicht gefunden.
 - Bernhard Hohlfeld
- Software wird hauptsächlich getestet
- Testverfahren können nie alle Systemzustände erreichen
 - Wurde ausreichend getestet?
- Software muss für Tests vorhanden und ausführbar sein
 - Ist die Spezifikation korrekt?
- Interne Qualitätseigenschaften werden selten geprüft
- Ist die Software zuverlässig, wartbar, änderbar etc. ?
- Ist die Software effizient?
- Wie viel Speicher/Zeit verbraucht das System maximal?

Maßnahmen zur Qualitätssicherung von Software

- Verifikation
 - Statische Techniken
 - Review
Walkthrough, Fagan-Inspektion, Code-Inspektion, Peer-Review, ...
 - Analyse
Statische Analyse, Formale Prüfung, Kontroll- und Datenfluss, ...
 - Dynamischer Test
Komponenten-/Integrationstest
 - Black-Box-Test
Funktionale Leistungsfähigkeit, Stress, Grenzwert, Fehlererwartung, ...
 - White-Box-Test
Struktur, Pfad, Zweig, Bedingung, Abdeckung, ...
- Validation
 - Animation, Formale Spezifikation, Modellierung, Simulation, Rapid Prototyping, ...
 - Systemtest/Akzeptanztest
Funktionale Leistungsfähigkeit, Stresstests, Grenzwerttests, Fehlererwartungstests, Ursache-Wirkungs-Graph, Äquivalenzklassentests, ...
 - siehe auch <http://campar.in.tum.de/twiki/pub/Chair/TeachingWs04MedInform/Softwaresicherheit.pdf>

6. SW-Entwicklung / 3. Unterstützungsprozesse

Unterstützungsprozesse für die Embedded Software Entwicklung

1. Vorgehensmodelle und Standards
2. Konfigurationsmanagement
3. Projektmanagement
4. Lieferantenmanagement
5. Anforderungsmanagement
6. **Qualitätssicherung**
 1. Integrations- und Testschritte
 2. Maßnahmen zur Qualitätssicherung von Software
 3. **Qualitätsstandards**

Qualitätsstandards (1)

- EN ISO 9000 und 9001
 - Darstellung eines QM-Systems
 - Grundlagen
 - Begriffe
 - Branchenneutral
- QS 9000
 - Automotive Anpassung der ISO 9000 durch GM, Ford und Chrysler
 - Höhere Forderungen
 - Kundenzufriedenheit
 - Kaizen
 - Ergebnisorientierung
 - Interdisziplinäre Teamarbeit
 - Kunden-Lieferanten-Beziehung

Qualitätsstandards (2)

- VDA 6.x

Die deutsche Automobilindustrie hat die in der ISO 9001 festgelegten Anforderungen weiterentwickelt und in den Regelwerken VDA 6.1, VDA 6.2 und VDA 6.4 dokumentiert.

Quelle: <http://www.vda-qmc.de/>

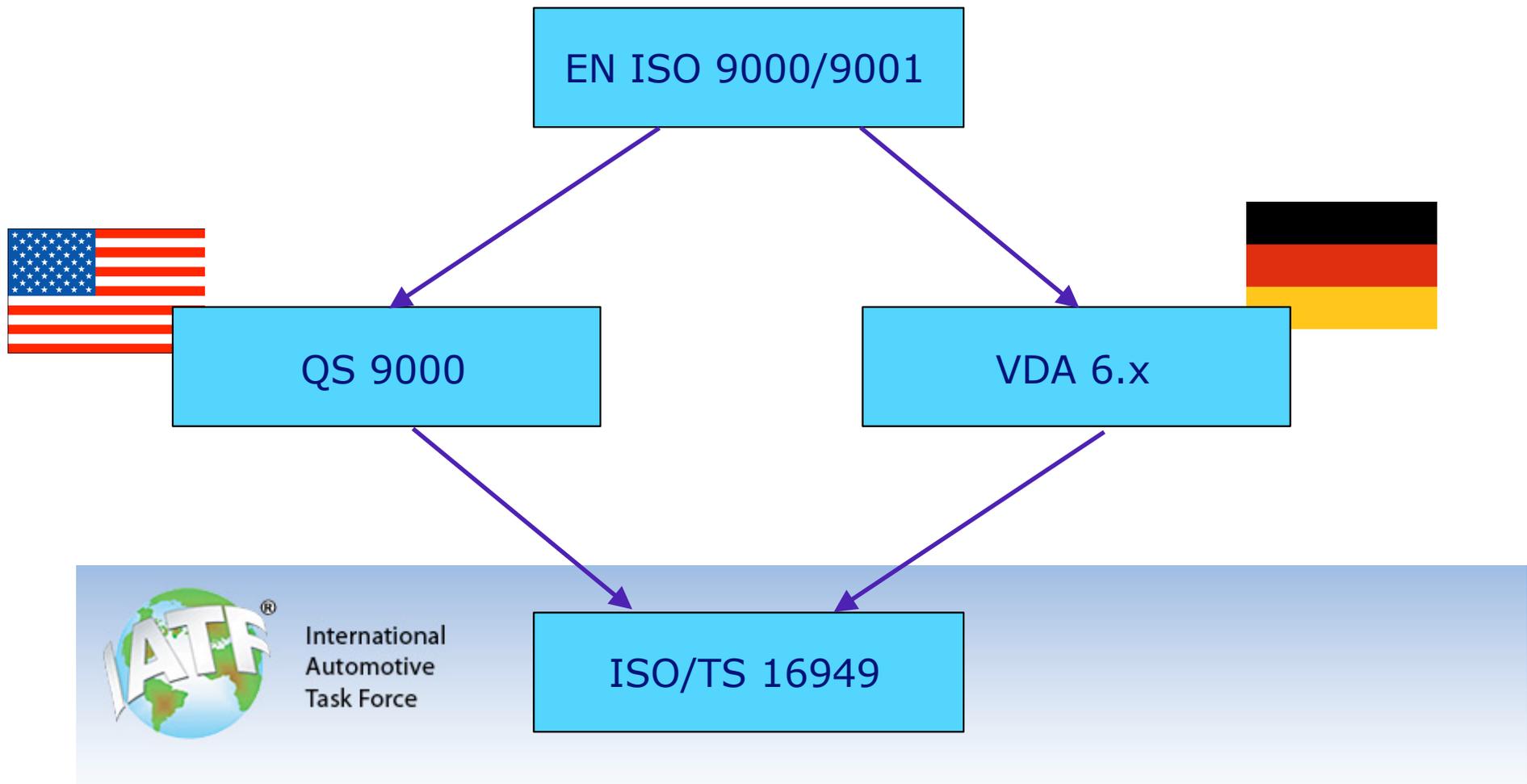
- ISO/TS 16949

Mit der ISO/TS 16949 wurden von den Vertretern der International Automotive Task Force (IATF) Formulierungen gefunden, die inhaltlich eine größtmögliche Identifikation mit den nationalen Standards AVSQ, EAQF, QS-9000 und VDA 6.1 ermöglichen.

Um den Übergang von VDA 6.1 zu ISO/TS 16949:1999 transparenter zu gestalten, hatte VDA-QMC 1999 den Textvergleich zwischen VDA 6.1 4. Auflage und ISO/TS 16949:1999 erstmalig veröffentlicht. Nun stehen viele Unternehmen vor der Aufgabe, ihr QM-System von VDA 6.1 oder ISO/TS 16949:1999 ausgehend, zur ISO/TS 16949:2002 weiterzuentwickeln.

Quelle: http://webshop.vda.de/QMC/product_info.php?products_id=49

Qualitätsstandards



EN ISO 9000 und 9001

- EN ISO 9000
- definiert Grundlagen und Begriffe zu Qualitätsmanagementsystemen.
- Erläutert werden die Grundlagen für Qualitätsmanagementsysteme und die in der Normenreihe EN ISO 9000 ff. verwendeten Begriffe. Die europäische Norm ISO 9000:2000 wurde in drei offiziellen Fassungen in englischer, deutscher und französischer Sprache veröffentlicht. Auch der prozessorientierte Ansatz des Qualitätsmanagements wird erklärt, basierend auf dem nach William Edwards Deming benannten Demingkreis (engl. auch Deming Cycle bzw. PDCA).
- EN ISO 9001
- legt die Anforderungen an ein Qualitätsmanagementsystem (QM-System) für den Fall fest, dass eine Organisation ihre Fähigkeit darlegen muss, Produkte bereitzustellen, welche die Anforderungen der Kunden und allfällige behördliche Anforderungen erfüllen, und anstrebt, die Kundenzufriedenheit zu erhöhen.
- Diese Norm beschreibt modellhaft das gesamte Qualitätsmanagementsystem und ist Basis für ein umfassendes Qualitätsmanagementsystem.

Quelle: Wikipedia

EN ISO 9000 und 9001

- Acht Grundsätze des Qualitätsmanagements
 - Kundenorientierung
 - Verantwortlichkeit der Führung
 - Einbeziehung der beteiligten Personen („Stakeholder“)
 - Prozessorientierter Ansatz
 - Systemorientierter Managementansatz
 - Kontinuierliche Verbesserung
 - Sachbezogener Entscheidungsfindungsansatz
 - Lieferantenbeziehungen zum gegenseitigen Nutzen

- Quelle: Wikipedia

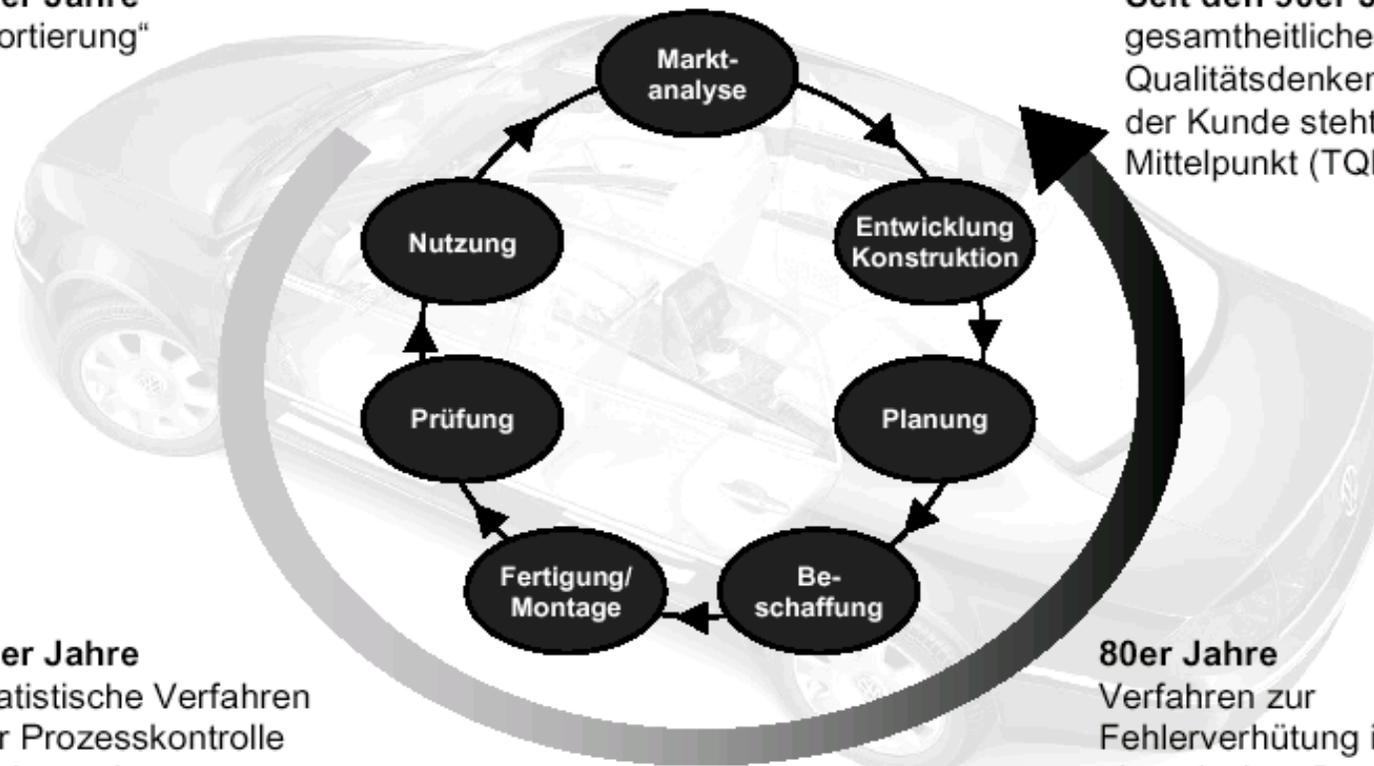
QS9000

- INTRODUCTION TO QS 9000
- Abstract: QS-9000 is the name given to the Quality System Requirements of the automotive industry which were developed by Chrysler, Ford, General Motors and major truck manufacturers and issued in 1994. QS-9000 is sometimes seen as being identical to ISO 9000. However, QS-9000 adds clauses to many of the ISO 9000 elements. Some of the differences in QS-9000 and ISO-9000 and the responsibilities of each employee under QS-9000 are given in this paper. This paper also describes the two types of audits, how employees should respond to audits, and the QS-9000 Quality Statement. Other terminology unique to QS-9000 is also given.
- Quelle:
<http://technologyinterface.nmsu.edu/summer97/manufacturing/qs9000.html>

Qualitätssicherung: Standards und Methoden

Entwicklung des Qualitätsmanagements

20er Jahre
„Sortierung“



Seit den 90er Jahren
gesamtheitliches
Qualitätsdenken -
der Kunde steht im
Mittelpunkt (TQM,..)

30er Jahre
Statistische Verfahren
zur Prozesskontrolle
und -regelung

80er Jahre
Verfahren zur
Fehlerverhütung in
planerischen Bereichen
(FMEA,..)

Qualitätssicherung: Standards und Methoden

Beispiel FMEA - Motorenentwicklung



- **Motor „Typ 12“**
- **Biturbo-System mit
mit zwei Ladeluftkühlern**
- **405 kW / 550 PS**
- **900 Nm**

Failure Mode and Effects Analysis								Blatt Nr.:			
Produkt- feature	Möglicher Fehler	Mögliche Folgen	Mögliche Fehler- ursache	Aktueller Status				Maß- nahmen	Ver- ant- wortlich	Termin	
				Aktuelle Maß- nahme	Auftreten						RPZ
					Bedeutung	Entdeckung					
Feder Nr. 103-5	Bruch	Zylinder- ausfall	Ermüdung	Festigkeits- test	6	7	10	420	versch. R.B.Shaw	08/07/01	
Öldicht- schraube	Leck	Ölverlust, Über- hitzung	Dichtung nicht fest genug	Höheres Montage- moment	7	9	9	567	dickere Dichtung R.Frost	05/09/01	

Bewertungszahlen:

A - Auftretenswahrscheinlichkeit

1 (unwahrscheinlich)
10 (hoch)

B - Bedeutung

1 (keine Bedeutung)
10 (sehr hohe Bedeutung)

E - Entdeckungswahrscheinlichkeit

1 (hoch)
10 (unwahrscheinlich)

**FMEA:
Fehler-Möglichkeits- und Einflussanalyse
Failure Mode and Effect Analysis**

Qualitätssicherung: Standards und Methoden

Beispiel FMEA - Motorenentwicklung



- Motor „Typ 12“
- Biturbo-System mit mit zwei Ladeluftkühlern
- 405 kW / 550 PS
- 900 Nm

Failure Mode and Effects Analysis								Blatt Nr.:		
Produktfeature	Möglicher Fehler	Mögliche Folgen	Mögliche Fehlerursache	Aktueller Status				Maßnahmen	Verantwortlich	Termin
				Aktuelle Maßnahme	Auftreten		RPZ			
					Bedeutung	Entdeckung				
Feder Nr. 103-5	Bruch	Zylinderausfall	Ermüdung	Festigkeits-test	6	7	10	420	versch. R.B.Shaw	08/07/01
Öldichtschraube	Leck	Ölverlust, Überhitzung	Dichtung nicht fest genug	Höheres Montage-moment	7	9	9	567	dickere Dichtung R.Frost	05/09/01

Bewertungszahlen:

A - Auftretenswahrscheinlichkeit

- 1 (unwahrscheinlich)
- 10 (hoch)

B - Bedeutung

- 1 (keine Bedeutung)
- 10 (sehr hohe Bedeutung)

E - Entdeckungswahrscheinlichkeit

- 1 (hoch)
- 10 (unwahrscheinlich)

**FMEA:
Fehler-Möglichkeiten- und Einflussanalyse
Failure Mode and Effect Analysis**

Qualitätssicherung: Standards und Methoden

Beispiel FMEA - Motorenentwicklung



- Motor „Typ 12“
- Biturbo-System mit mit zwei Ladeluftkühlern
- 405 kW / 550 PS
- 900 Nm

RPZ: Risiko-Prioritätszahl

Failure Mode and Effects Analysis								Blatt Nr.:			
Produktfeature	Möglicher Fehler	Mögliche Folgen	Mögliche Fehlerursache	Aktueller Status				RPZ	Maßnahmen	Verantwortlich	Termin
				Aktuelle Maßnahme	Auftreten						
					Bedeutung	Entdeckung					
Feder Nr. 103-5	Bruch	Zylinderausfall	Ermüdung	Festigkeits-test	6	7	10	420	versch.	R.B.Shaw	08/07/01
Öldichtschraube	Leck	Ölverlust, Überhitzung	Dichtung nicht fest genug	Höheres Montage-moment	7	9	9	567	dickere Dichtung	R.Frost	05/09/01

Bewertungszahlen:

A - Auftretenswahrscheinlichkeit

- 1 (unwahrscheinlich)
- 10 (hoch)

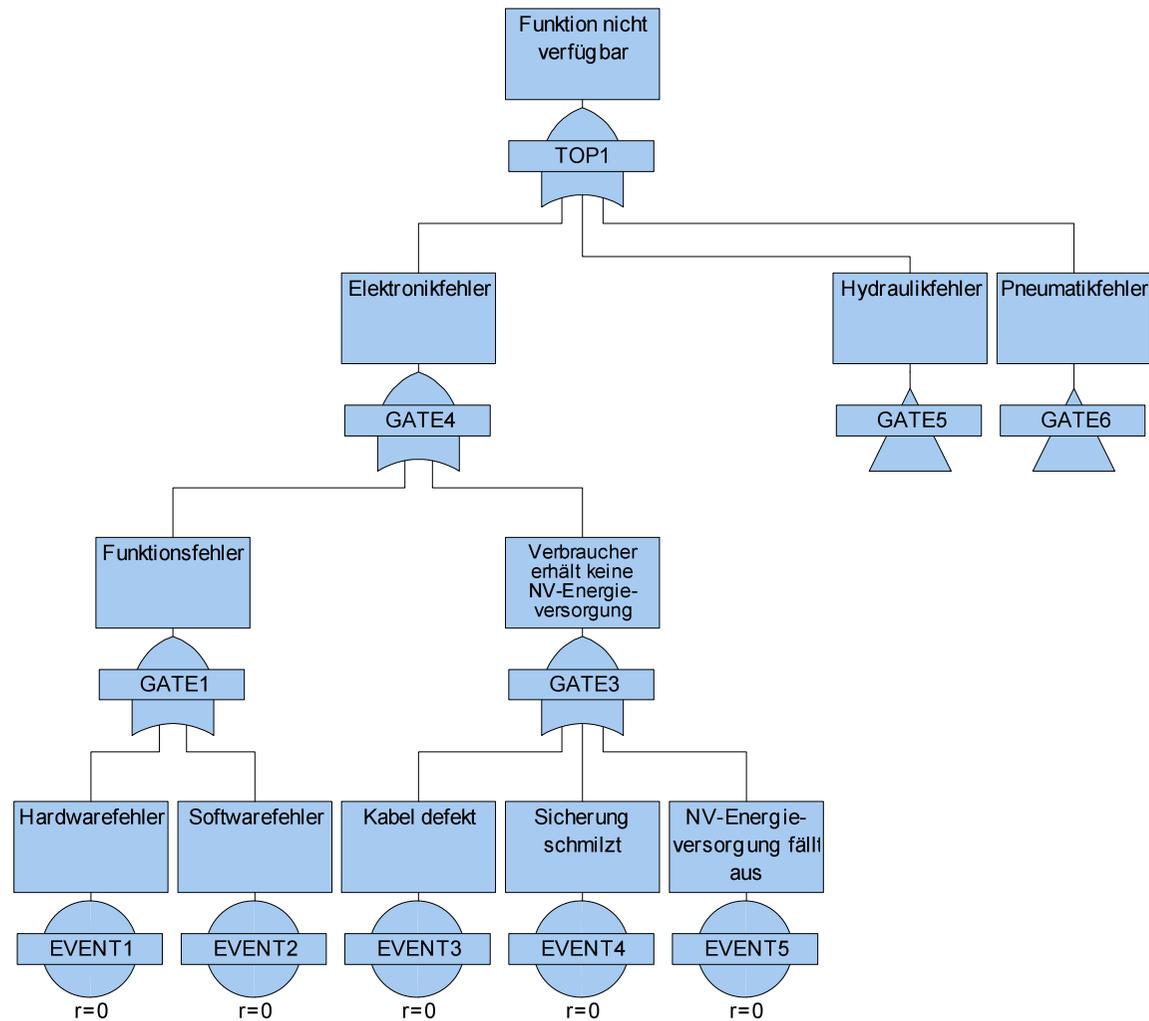
B - Bedeutung

- 1 (keine Bedeutung)
- 10 (sehr hohe Bedeutung)

E - Entdeckungswahrscheinlichkeit

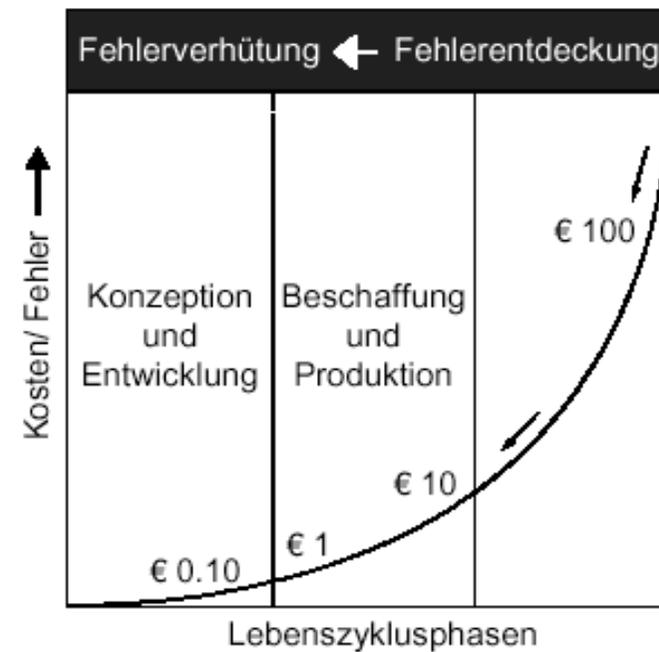
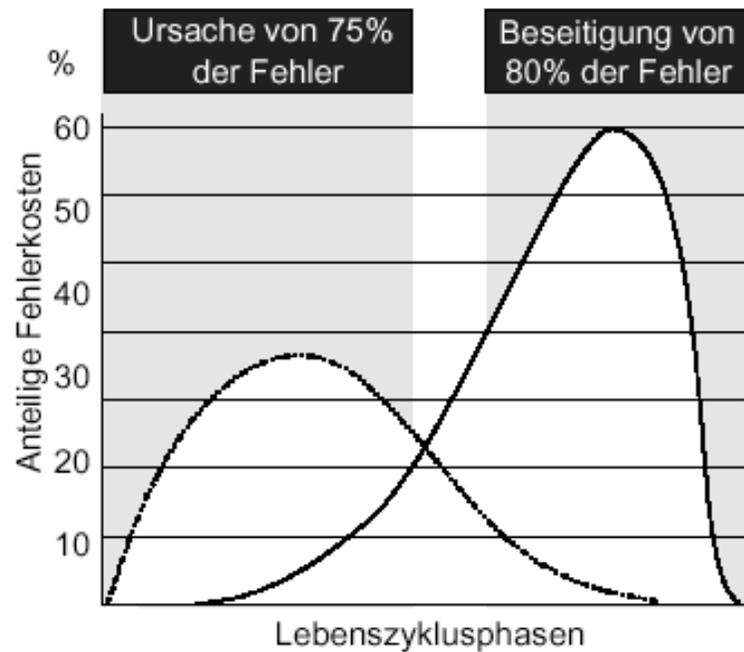
- 1 (hoch)
- 10 (unwahrscheinlich)

Fehlerbaum - Fault Tree



Qualitätssicherung: Standards und Methoden

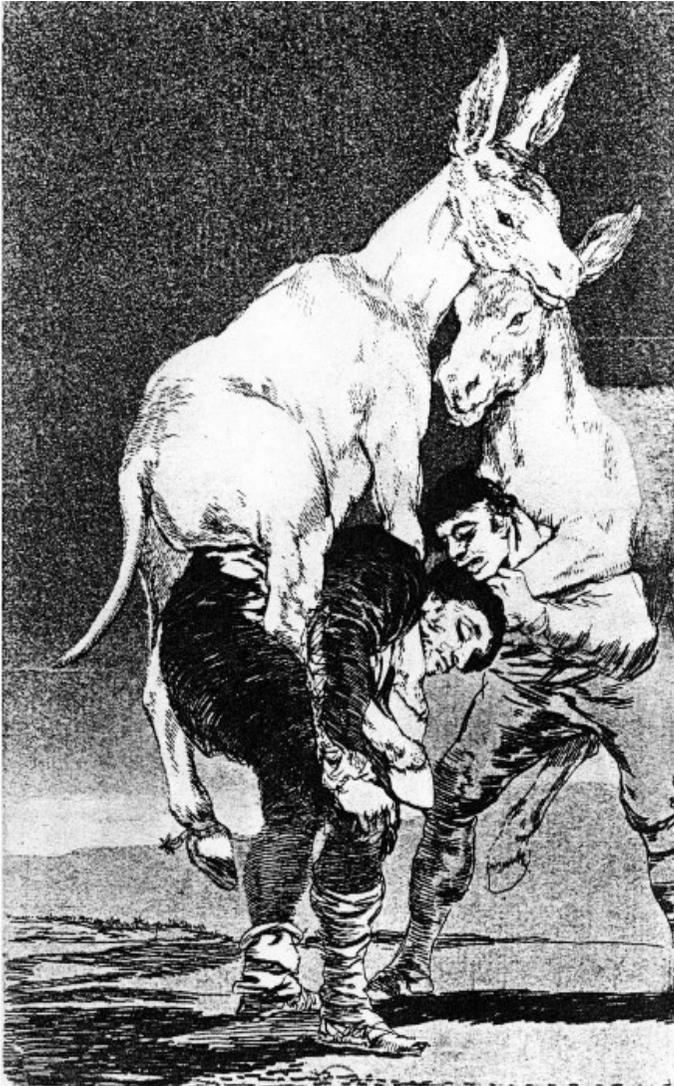
Fehler und deren Auswirkungen



Unterstützungsprozesse aus Sicht des Entwicklers



Unterstützungsprozesse aus Sicht des Entwicklers



Entwickler aus Sicht der Unterstützungsprozesse



Entwickler aus Sicht der Unterstützungsprozesse

