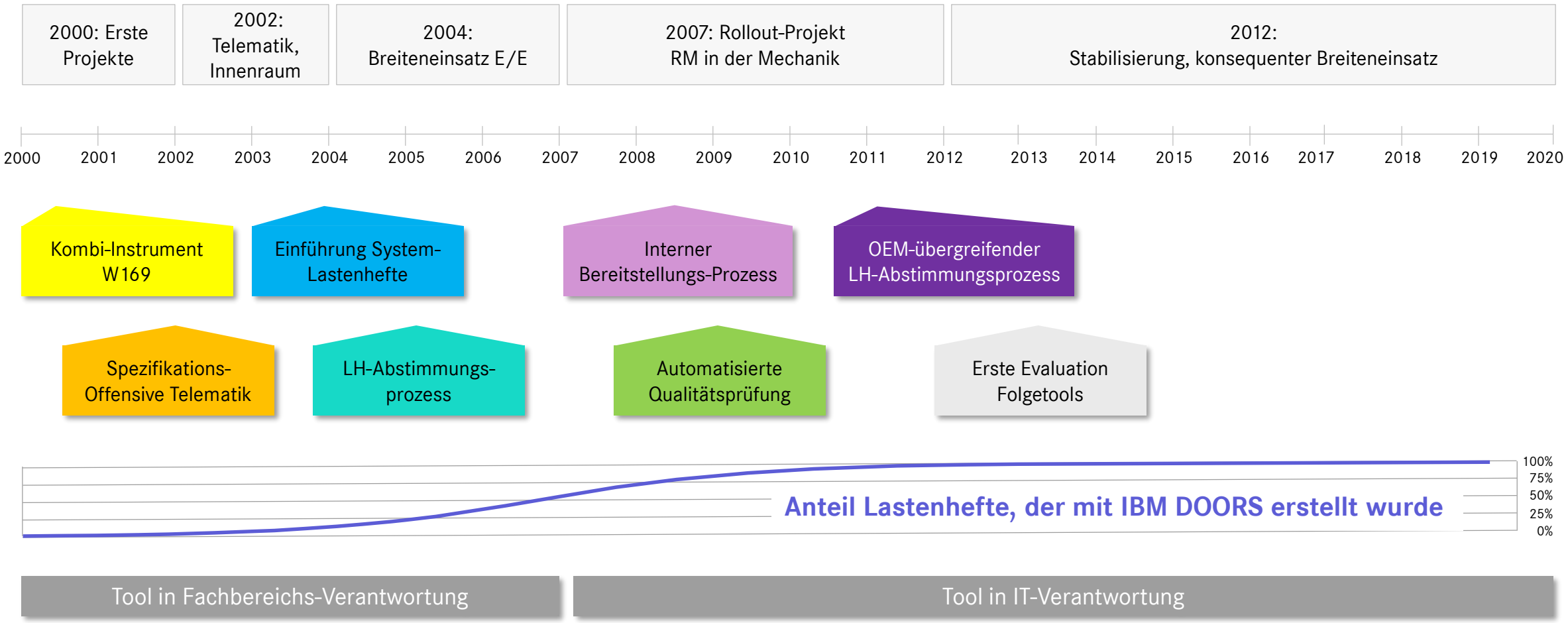
Ein RM-Tool für alle Projekte – Das war einmal …

Dr. Frank Houdek

Mercedes-Benz
The best or nothing.

Requirements Management @ Mercedes-Benz Cars

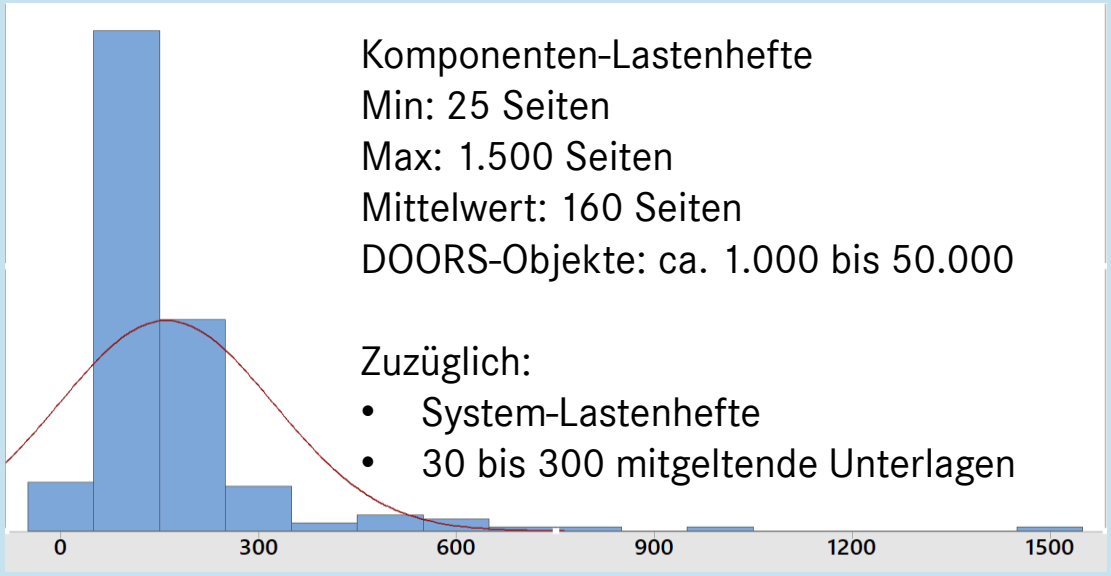
Ein kurzer historischer Abriss ...



Alle Abbildungen in dieser Präsentation: © Mercedes-Benz AG oder frei verfügbare Abbildungen unter Creative Commons Lizenz

Einblicke in die Spezifikations-Welt der Mercedes-Benz PKW Entwicklung – Zahlen und Fakten

Spezifikationsgröße



Komponenten-Lastenhefte
Min: 25 Seiten
Max: 1.500 Seiten
Mittelwert: 160 Seiten
DOORS-Objekte: ca. 1.000 bis 50.000

Zuzüglich:
- System-Lastenhefte
- 30 bis 300 mitgeltende Unterlagen

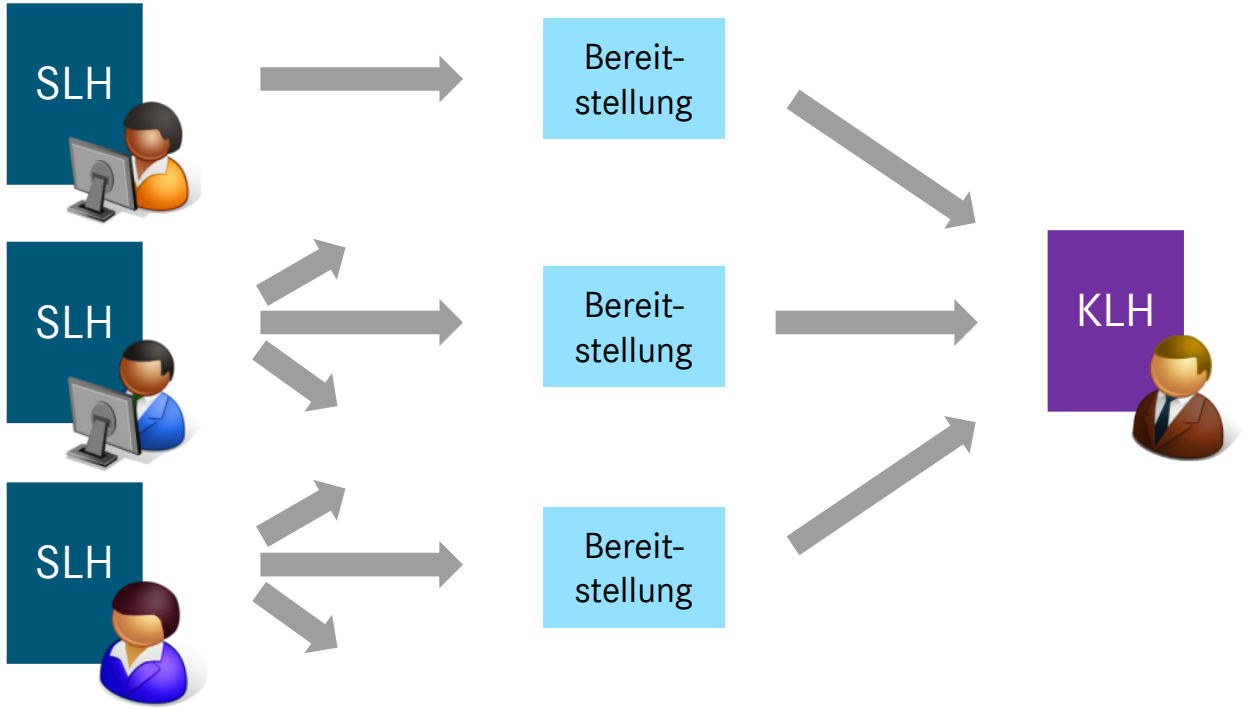
Spezifikations-Stil

- Basis: Umfangreiches LH-Template (Standard-Texte)
- Im wesentlichen natürliche Sprache (Deutsch, Englisch)
- Kein Einsatz von Satz-Schablonen
- Wo angemessen:
  - Tabellen
  - Abbildungen, Zeichnungen
  - Zustandsautomaten
  - etc.
- Typischerweise keine formale Spezifikation
- Kennzeichnung: Anforderung, Information, Überschrift

Ein RM-Tool = Eine RM-Kultur? **Ja und Nein!**
Gleiche Vorgaben und Randbedingungen, aber viele lokale Erweiterungen und Besonderheiten

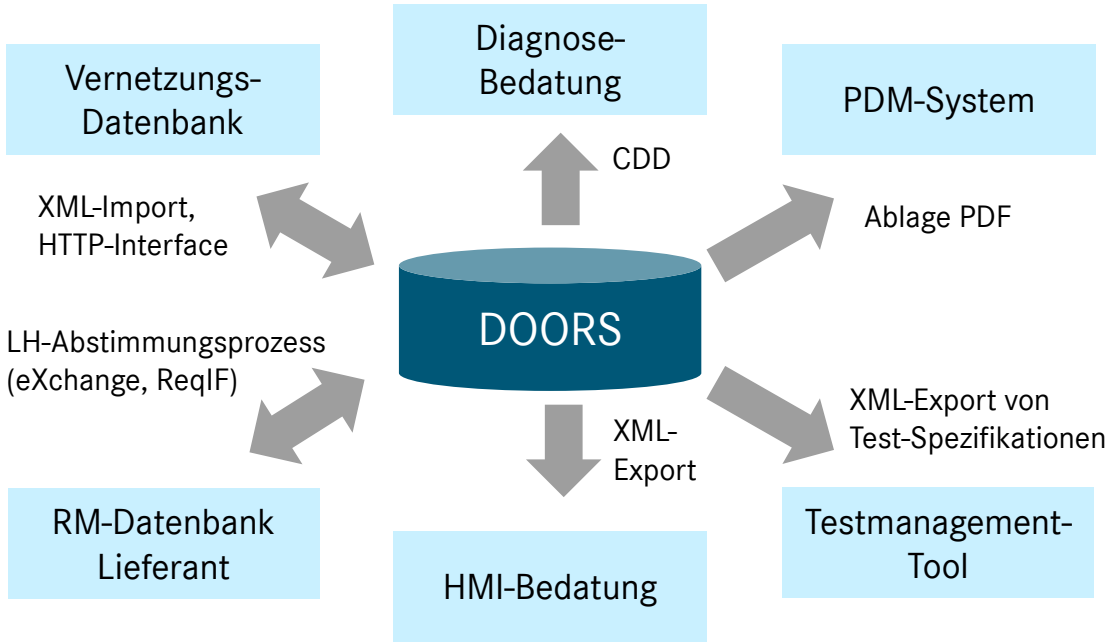
Einblicke in die Spezifikations-Welt der Mercedes-Benz PKW Entwicklung – Prozesse und Schnittstellen

Handover-Prozess



Vernetzte LH-Landschaft, die über definierte (versionierte) Stände gesteuert wird

Schnittstellen zu anderen Tools (Auswahl)



„Mühsame“ Vernetzung mit benachbarten Systemen

KLH: Komponentenlastenheft, SLH: Systemlastenheft

Need for Change (1/2)

Tool-Technische Randbedingungen
- Ankündigung Hersteller
- Security
- Integrierbarkeit in IT-Landschaft

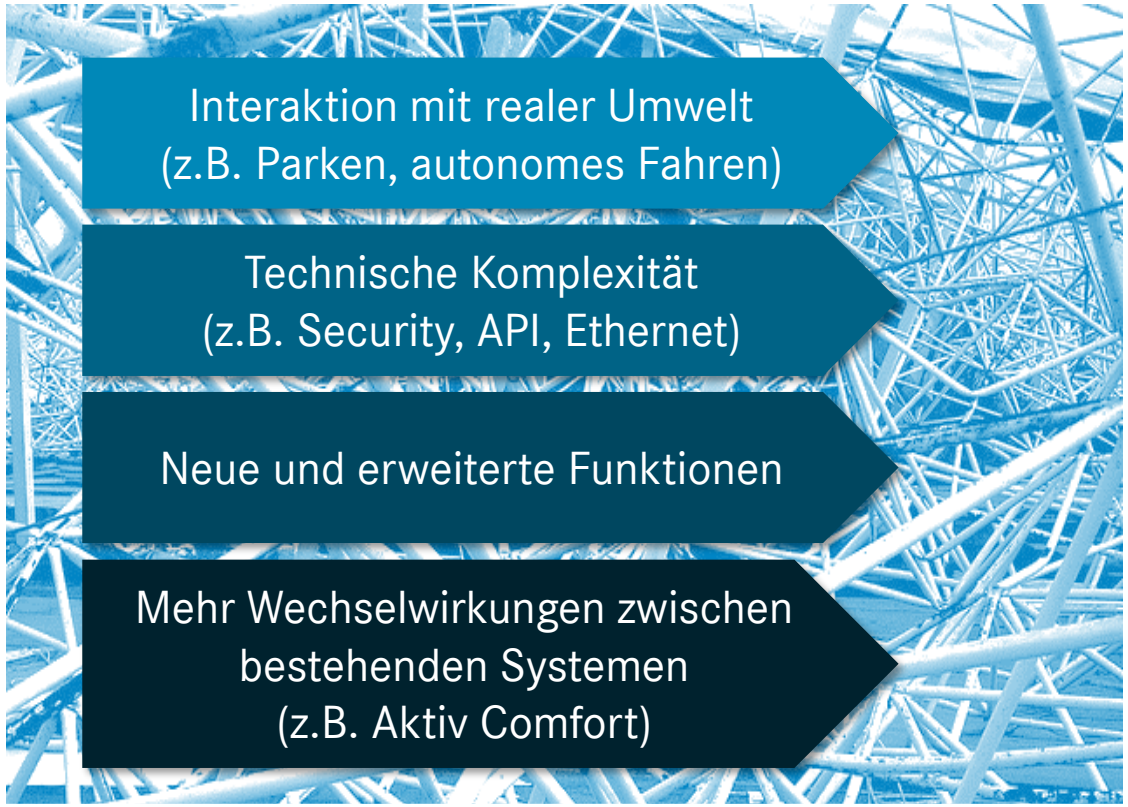
Implikationen
- Schulung 5.000+ User
- Methodenentwicklung und –dokumentation
- Tool-Customizing
- Coaching/Support

Ablösung von DOORS wird zweistelligen Millionen-Betrag kosten

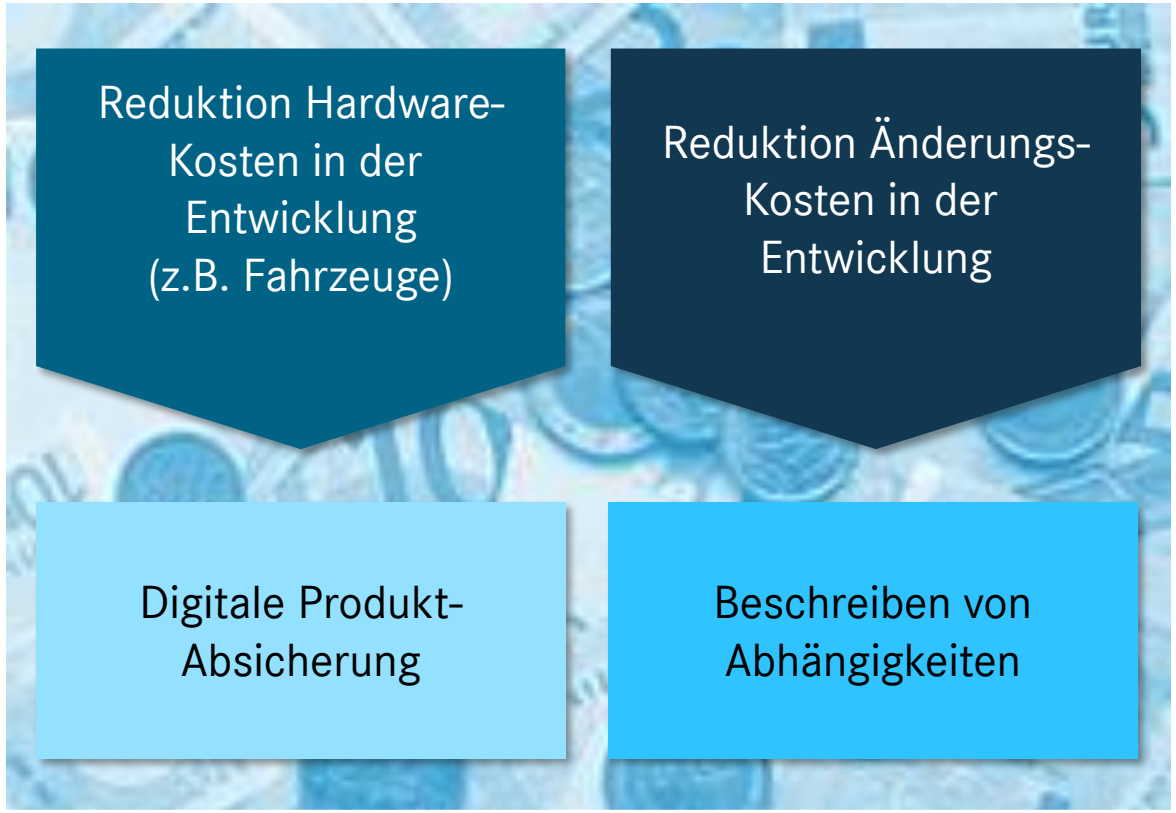
... nur um eine Schreibmaschine durch eine andere zu ersetzen?

Need for Change (2/2)

Steigende Produktkomplexität



Cost-to-Market



- Weniger rein textuelle Spezifikation
- Mehr modellbasierte Beschreibung

Bis hierher war es einfach …

Ziele ✓

Konkreter Weg ?

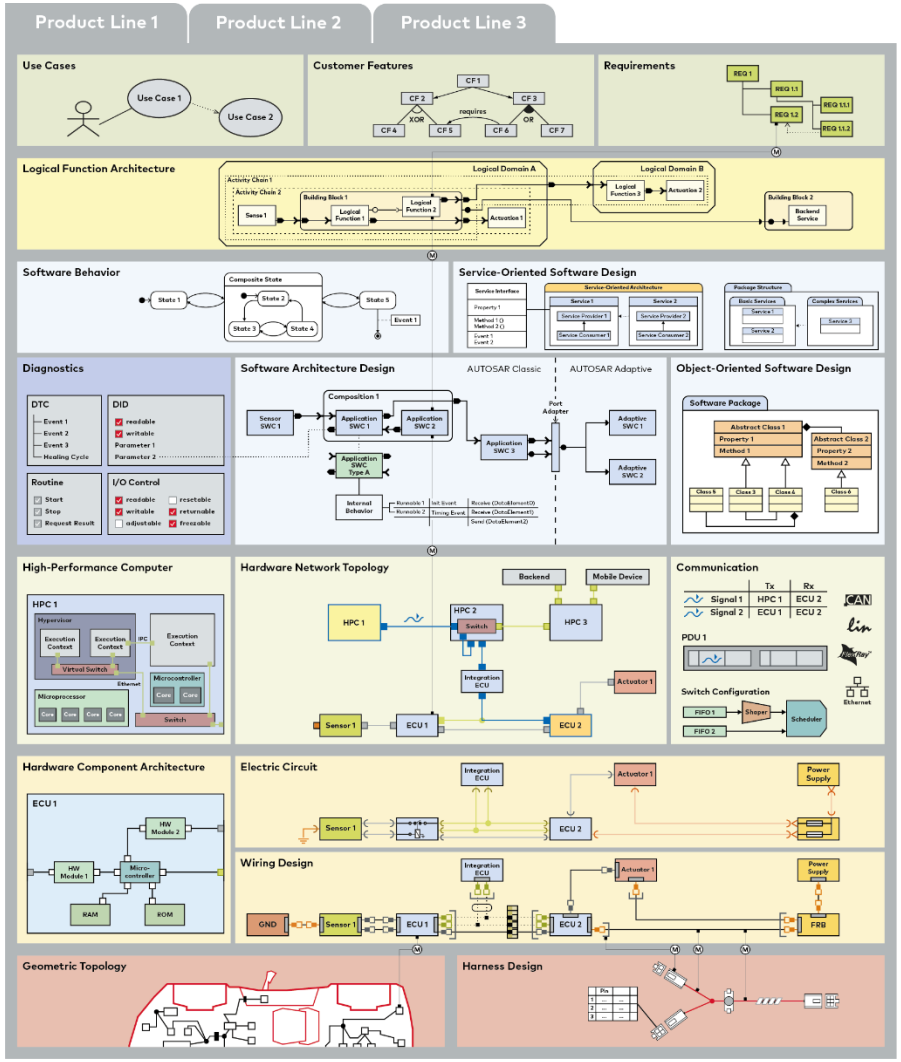
Fall 1: Innenraum-Elektronik-Entwicklung

- Komplexe Funktionsarchitektur
- Nutzung Kommunikationsdaten
- Logische Architektur
- Technische Architektur (AUTOSAR-SW-Komponenten)
- Diagnose-Daten-Prozess
- Übergabe Daten an Leitungssatzentwicklung
- Funktionsmodellierung mit Matlab/Simulink (=Implementierung)

- Viele verschiedene Datentypen (z.B. Signale, Pins, Parameter, DTCs, SWCs)
- Viele (definierte) Abhängigkeiten zwischen Daten
- (Relativ) klare Vorstellung, wohin der Weg gehen soll

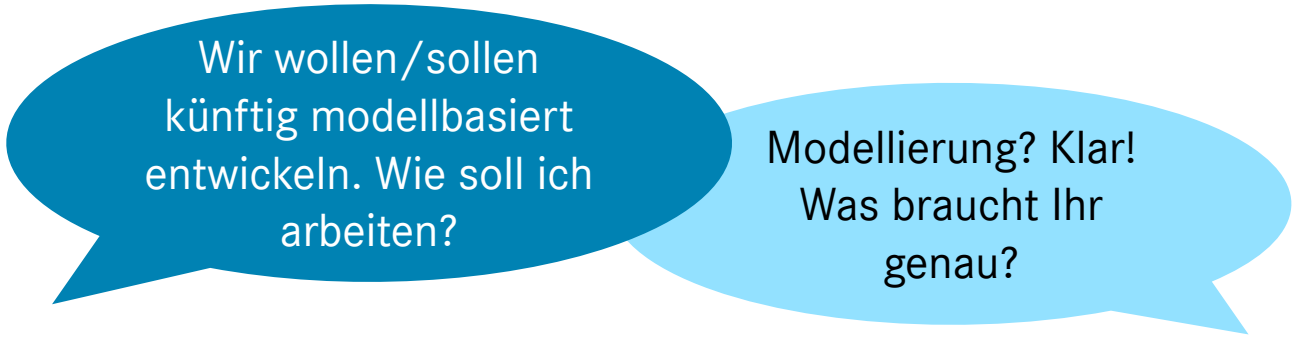
Ergebnis: Tool PREEvision von Vector

- Entscheidung, Tool PREEvision von Vector einzusetzen
- Tool kennt ca. 6000 E/E Datentypen und deren Beziehungen zueinander
- Ein „Tool-Vater“: E/E-Konzept-Tool von Daimler …
- Tool bietet Out-of-the-Box bereits viel, trotzdem Customizing notwendig, z.B.
  - Anforderungsmanagement
  - Handover-Prozess
  - Diagnose-Daten-Prozess
  - LH-Abstimmungsprozess
- Sehr mächtiges Tool, aber auch entsprechend anspruchsvoll



© Vector Informatik GmbH. Verwendung mit freundlicher Genehmigung der Vector Informatik GmbH

Fall 2: Mechanik-Entwicklung



- Deutlich mehr Unsicherheit, was modellbasierte Entwicklung bedeutet

- Sich abzeichnende Benefits:
  - Modellierung von Wirkketten
  - → Erkennen von Abhängigkeiten (z.B. bei Änderungen)

Sinnvolle Abstraktionsebene hängt von Anwendung ab!

- Tool-Entscheid: IBM Rational JAZZ Plattform mit DOORS Next Generation und IBM Rational Rhapsody

Wäre nicht ein RM-Tool für alle besser?

Fachliche Bedarfe
(z.B. Modellierung Wirkketten,
Unterstützung E/E-Entwicklung)

Unterschiedliche Einsatzszenarien
(Vollzeit-User vs. Ein LH in 5 Jahren)

Zunehmend Software-
Eigenentwicklung
(=weitere Funktionen)

Ein
RM-Tool
für alle

Technische Constraints
(Web-based? Thick-Client?)

Beherrschbarkeit für
den Anwender

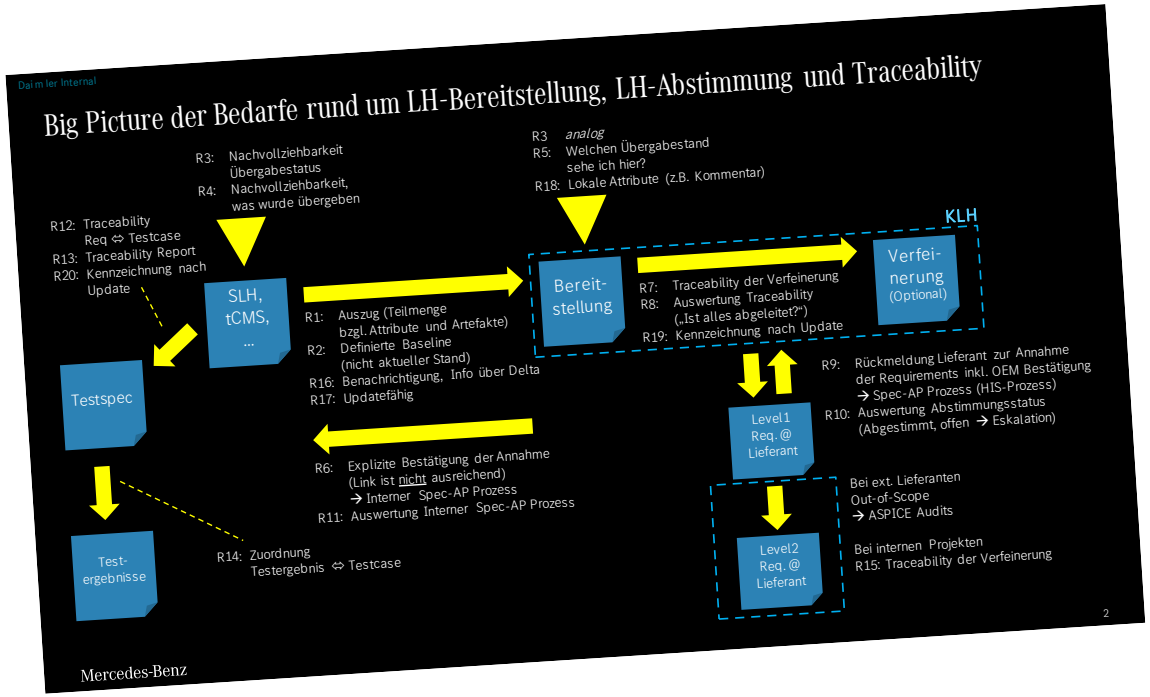
Neue Use Cases (z.B. Datenbank
für Regulatorische Vorgaben)

Theoretisch kann man alle benötigten
Funktionen in ein Tool integrieren …

… aber dieses wäre für viele Anwender
nicht mehr beherrschbar
→ Die Tool-Welt wird bunter

Und nun … ?

Definition der Zusammenarbeit bei bunterer Tool-Landschaft



- Trotz unterschiedlicher Tools gleiche Grundprinzipien der Arbeitsweise beibehalten

- Neue Tool-Features sind Chance und Risiko
  - Positiv: PREEvision Review & Vote
    → *Endlich Baselining trotz folgender Abstimmung*
  - Negativ: DNG nur ein Attribut mit Attachements
    → *Zweisprachigkeit wird aufwändiger*

Some take away …

Vorsicht Falle:
Powerpoint-
Engineering

RM-Tool-Welt
wird bunter

Gefahr des „Vergessens“
von Must-Have‘s
guter RM-Tools

ReqIF
Requirements Interchange Format

ReqIF ist nicht
gleich ReqIF

Belastbarkeit

DOORS Classic

DNG

Einige Beispiele …

Sinnvolle Requirement-IDs

Bisher:	*Jetzt (teilweise):*
STLH-1254	1313351
ILS-143	1313354
GDO-TS-3221	(und sie werden länger)

Echte, vollständige Baselines

In PreeVision ist z.B. der Workflow-Status normalerweise nicht Bestandteil einer Baseline

Views mit komplexen Filtern

Komplexe Filter (freie Kombination von AND, OR, NOT) werden oftmals nicht mehr unterstützt

Feingranulare Rechte

Setzen von Rechten auf Ebene von Attributen, Abschnitten, etc. wird oft nicht mehr unterstützt.

Requirements sind kein Designobjekt

Tools bieten oft umfangreiche Formatiermöglichkeiten (Schriftarten, -größe, etc.), die einheitliche „Requirements-Optik“ erschweren

Haben Sie Fragen?