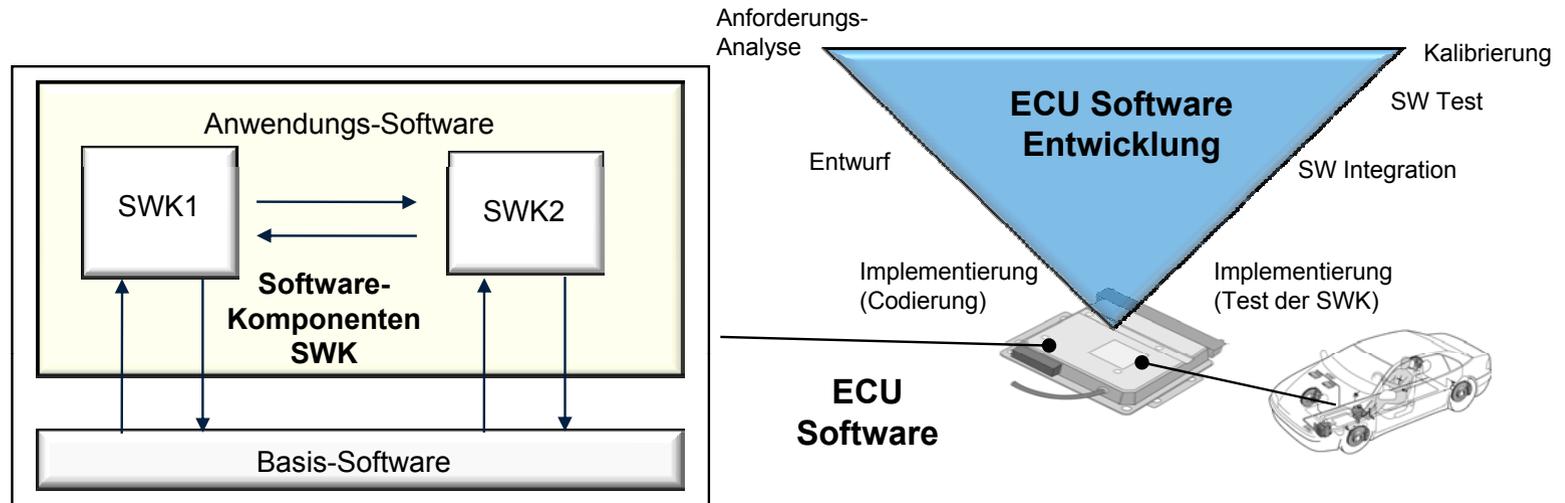




Produktlinien für die ECU Software- Entwicklung

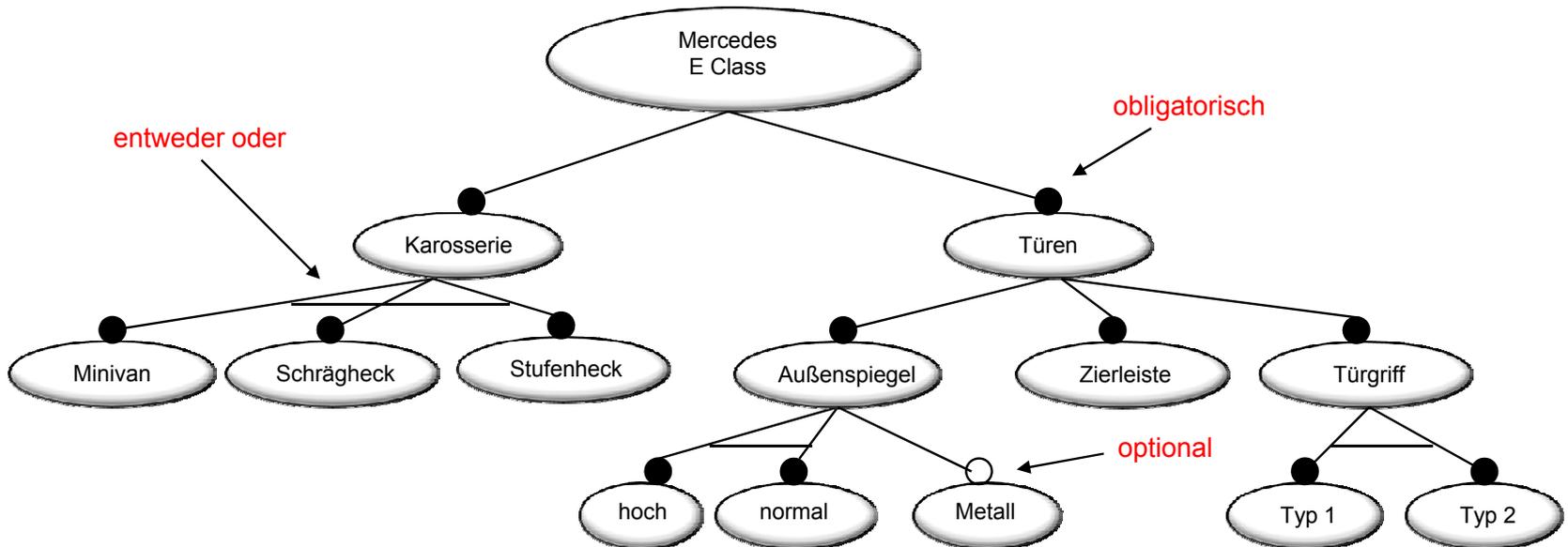
Dr. Thomas Zurawka, TU Darmstadt, 16.06.2009

ECU SW Architektur & SW Entwicklungsprozess



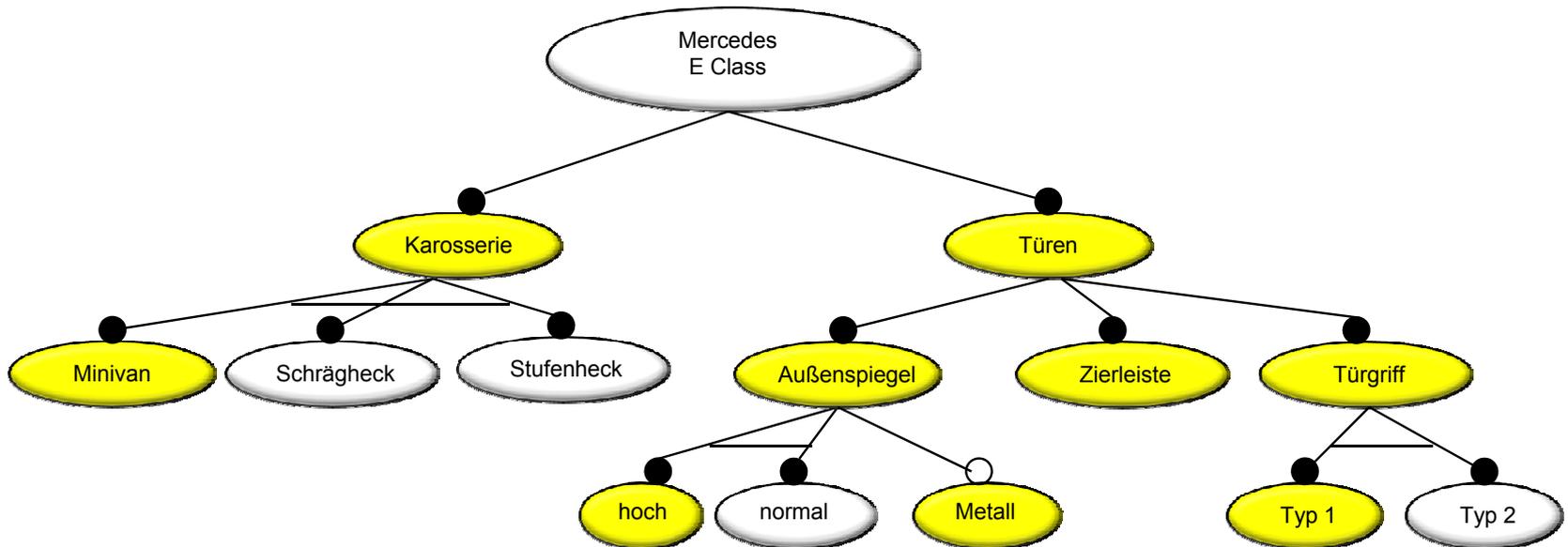
- **Ziele:** geringe Entwicklungskosten, sehr gute Software Qualität
- Lösung: **Wiederverwendung von SWK** und anderen Artefakten

Wiederverwendung: Varianten-Management & Features



Feature-Modellierung: Definition aller Features aller Produkte einer Anwendungsdomäne sowie deren Abhängigkeiten

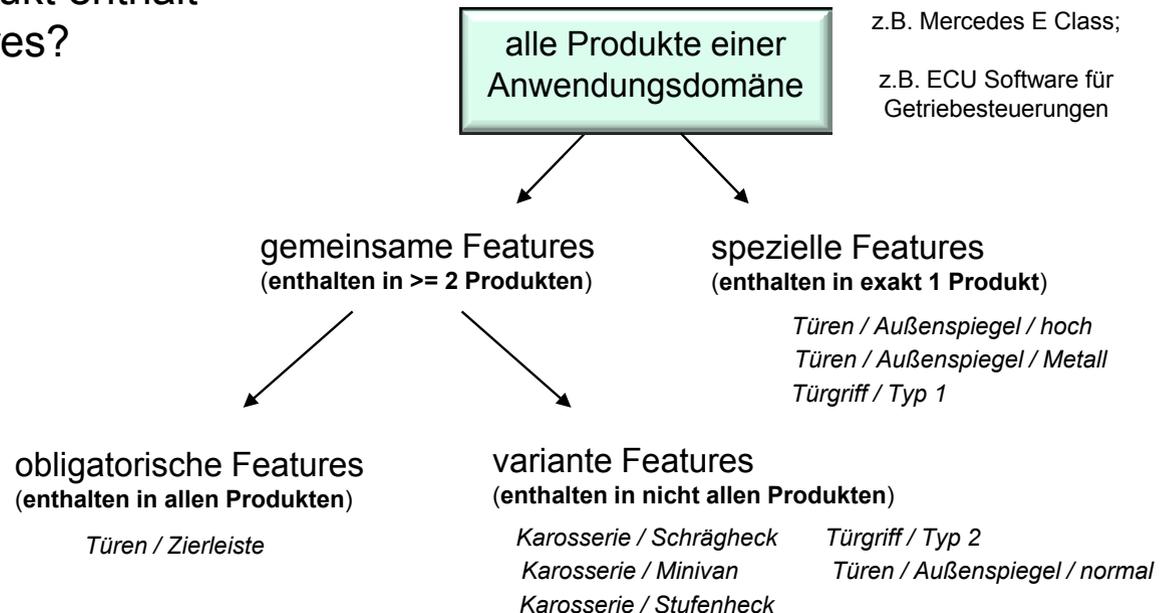
Wiederverwendung: Varianten-Management & Features



Feature-Konfiguration: Auswahl der “gelben” Features für ein bestimmtes Produkt

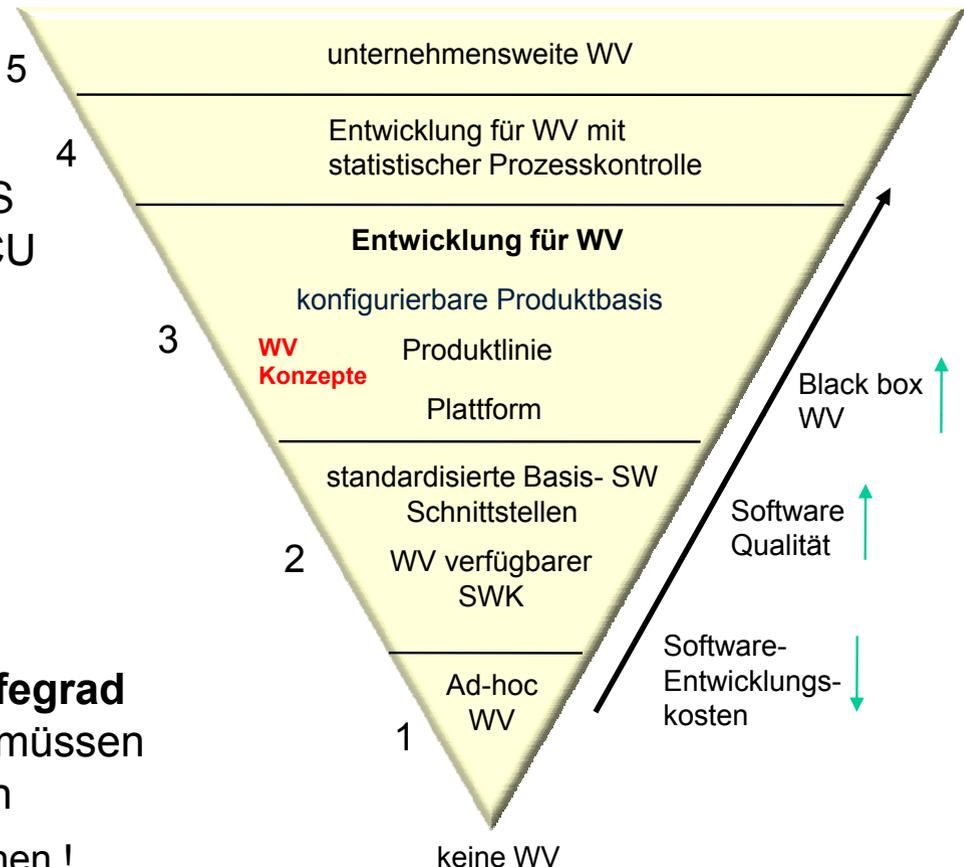
Wiederverwendung: Scoping

- **Identifikation** der gemeinsamen (obligatorisch & variant) und speziellen Features aller Produkte einer Anwendungsdomäne
- ist Antwort auf die Frage: Welches Produkt enthält welche Features?

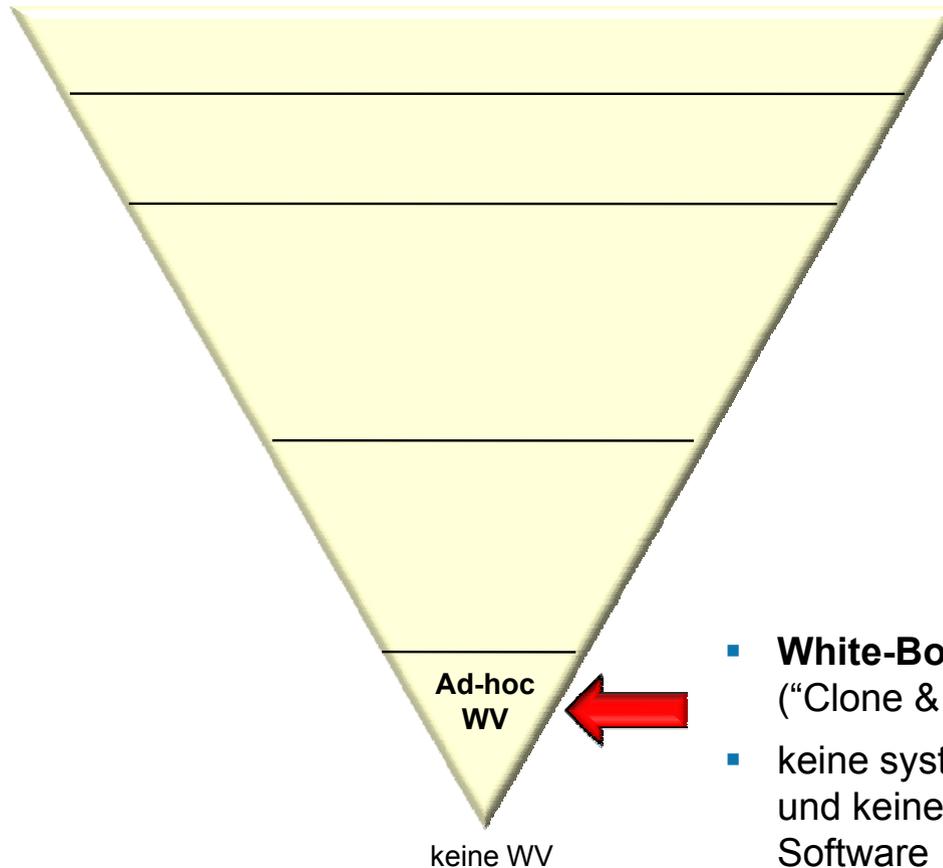


Maßnahmen

- neues **Reifegradmodell** für die WV von Software
 - entwickelt von SYSTECS & einigen automotive ECU Zulieferern
 - Organisationen kommen Schritt-für-Schritt auf ein höhere Stufe der WV
 - Migrationspfad, **Ziel-Reifegrad & Stufe 3 WV Konzept** müssen sorgfältig geplant werden
 - um die Ziele zu erreichen !

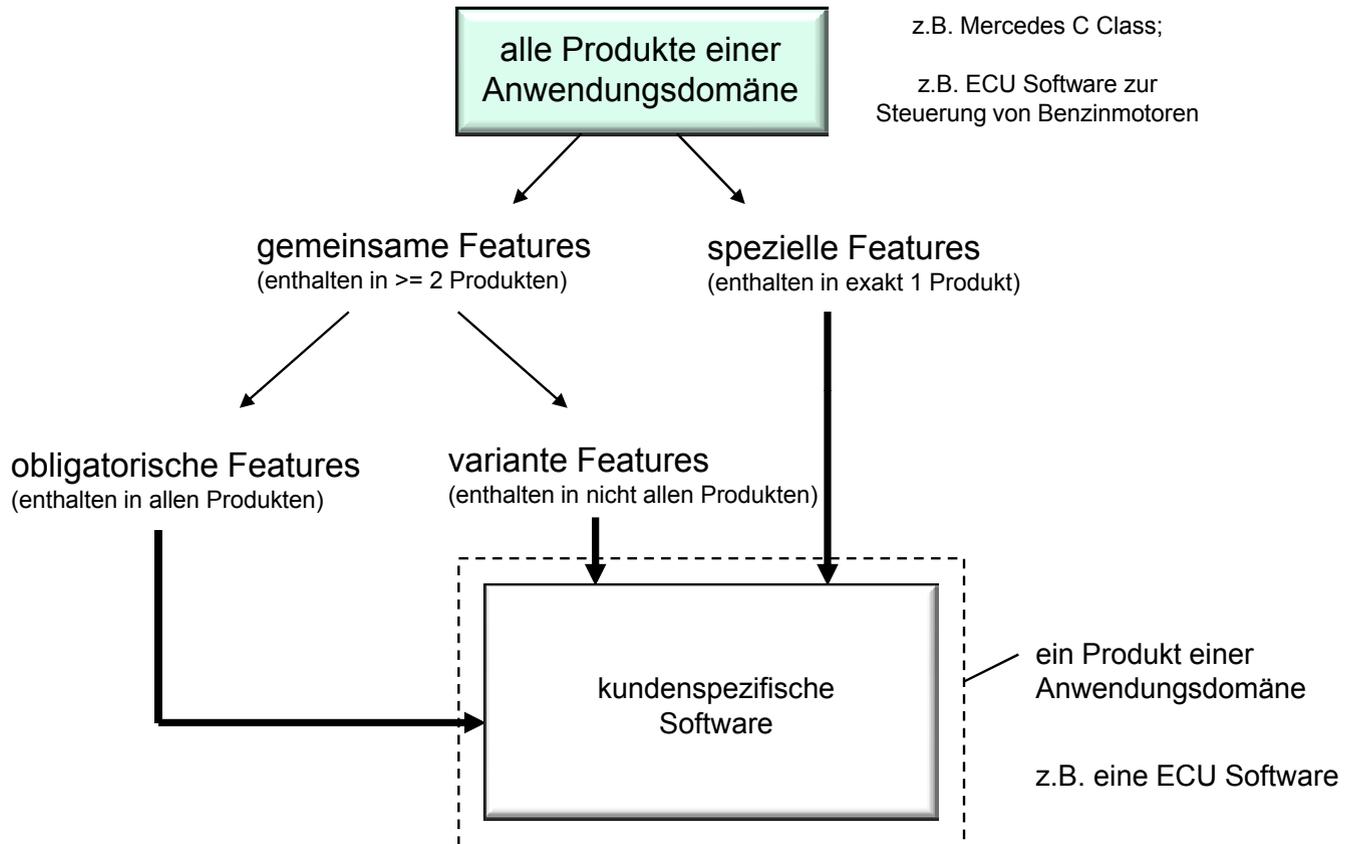


Reifegradmodell – Stufe 1

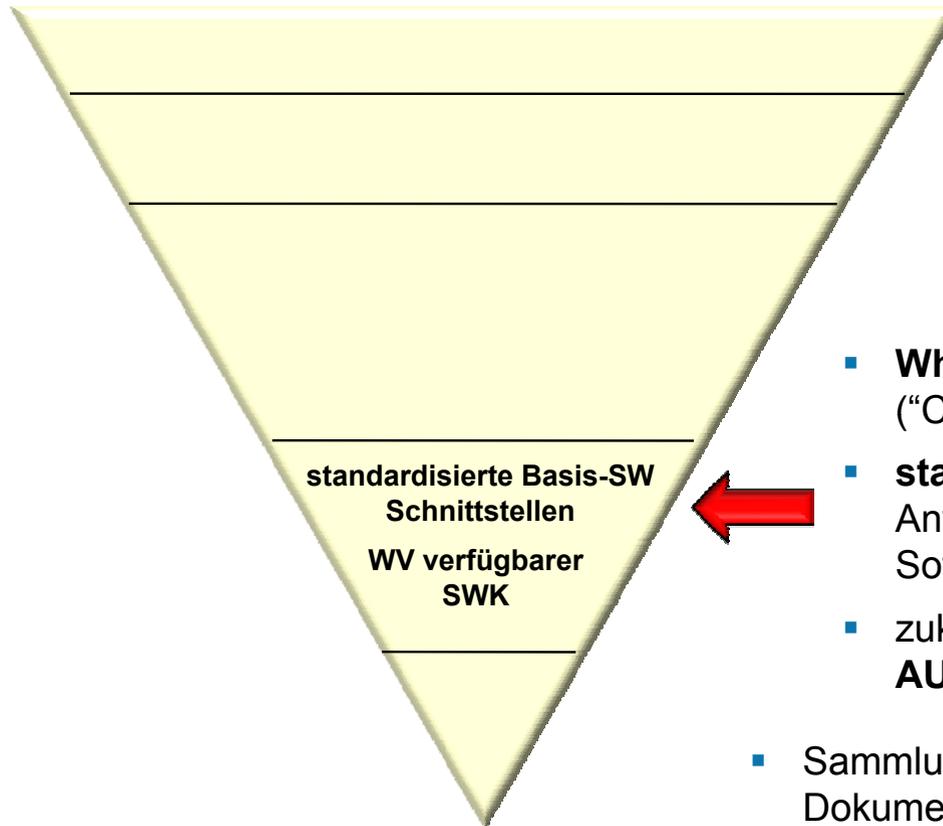


- **White-Box WV** von ECU Software (“Clone & own”)
- keine systematische, keine koordinierte und keine dokumentierte WV von ECU Software

Reifegradmodell – Stufe 1 (und 2)

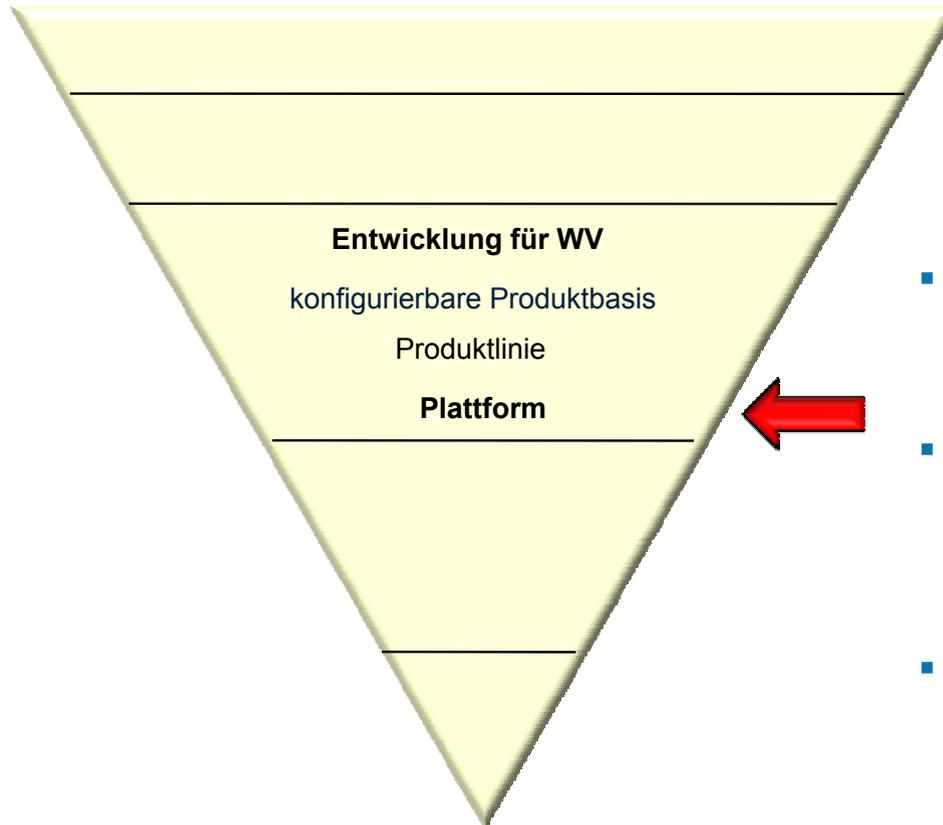


Reifegradmodell – Stufe 2



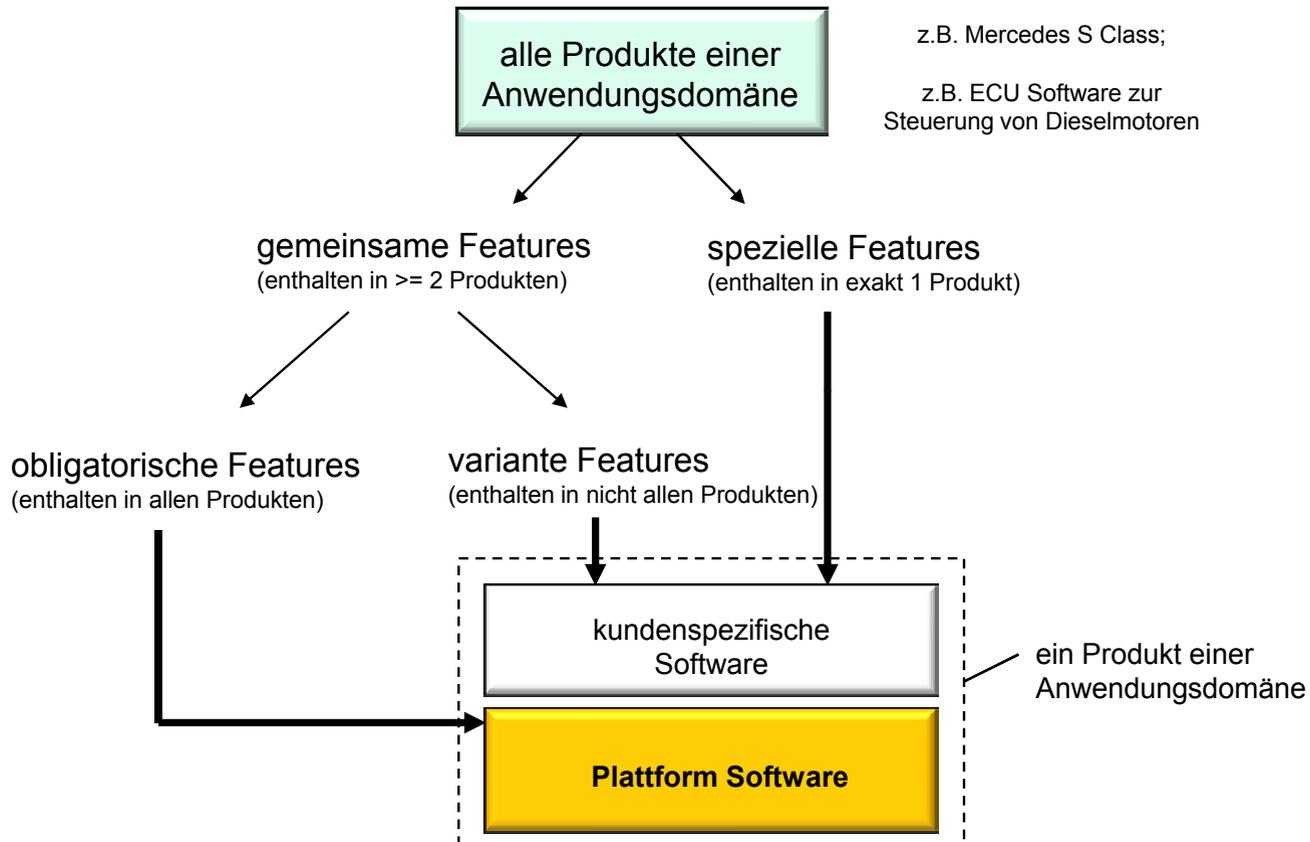
- **White-Box WV** von ECU Software (“Clone & own”)
- **stabile Schnittstellen** zwischen Anwendungs-Software und Basis-Software
- zukünftig: Standardisierung über **AUTOSAR**
- Sammlung, Strukturierung und Dokumentation verfügbarer SWK in systematischer Weise

Reifegradmodell – Stufe 3 (Plattform)

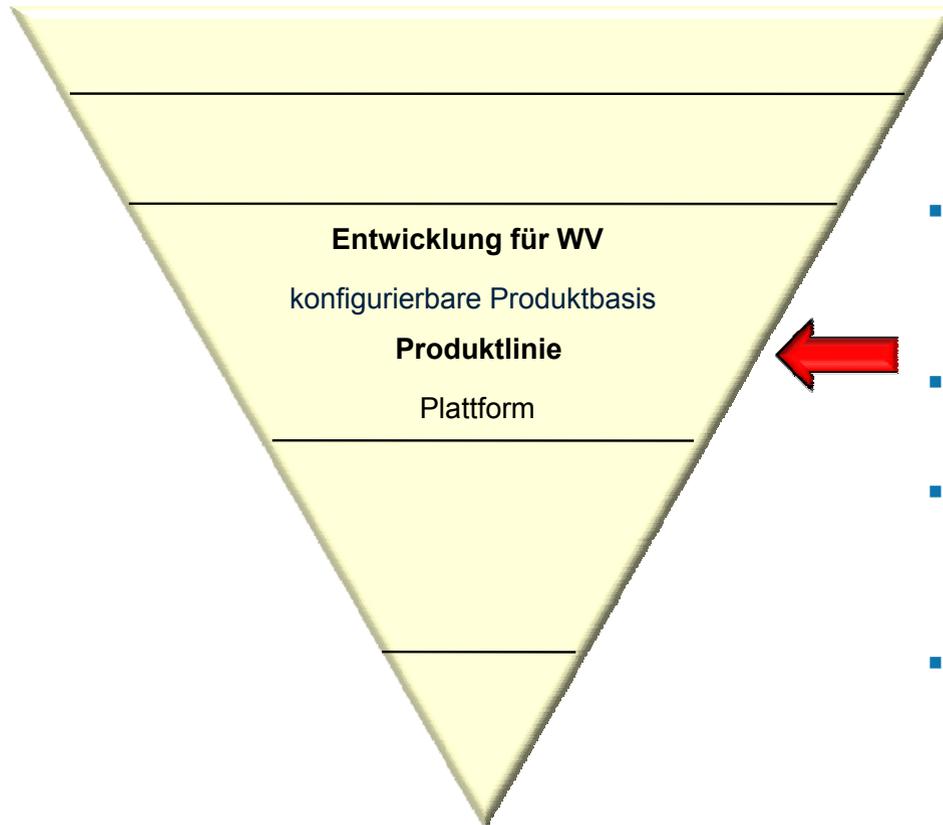


- **Black-Box & White-Box WV** von ECU Software
- besteht aus den gemeinsamen SWK, die in **allen** Produkten eingesetzt werden
- geeignet für **reife** Anwendungsdomänen
 - gemeinsame Features aller Produkte wurden bereits identifiziert

Reifegradmodell – Stufe 3 (Plattform)

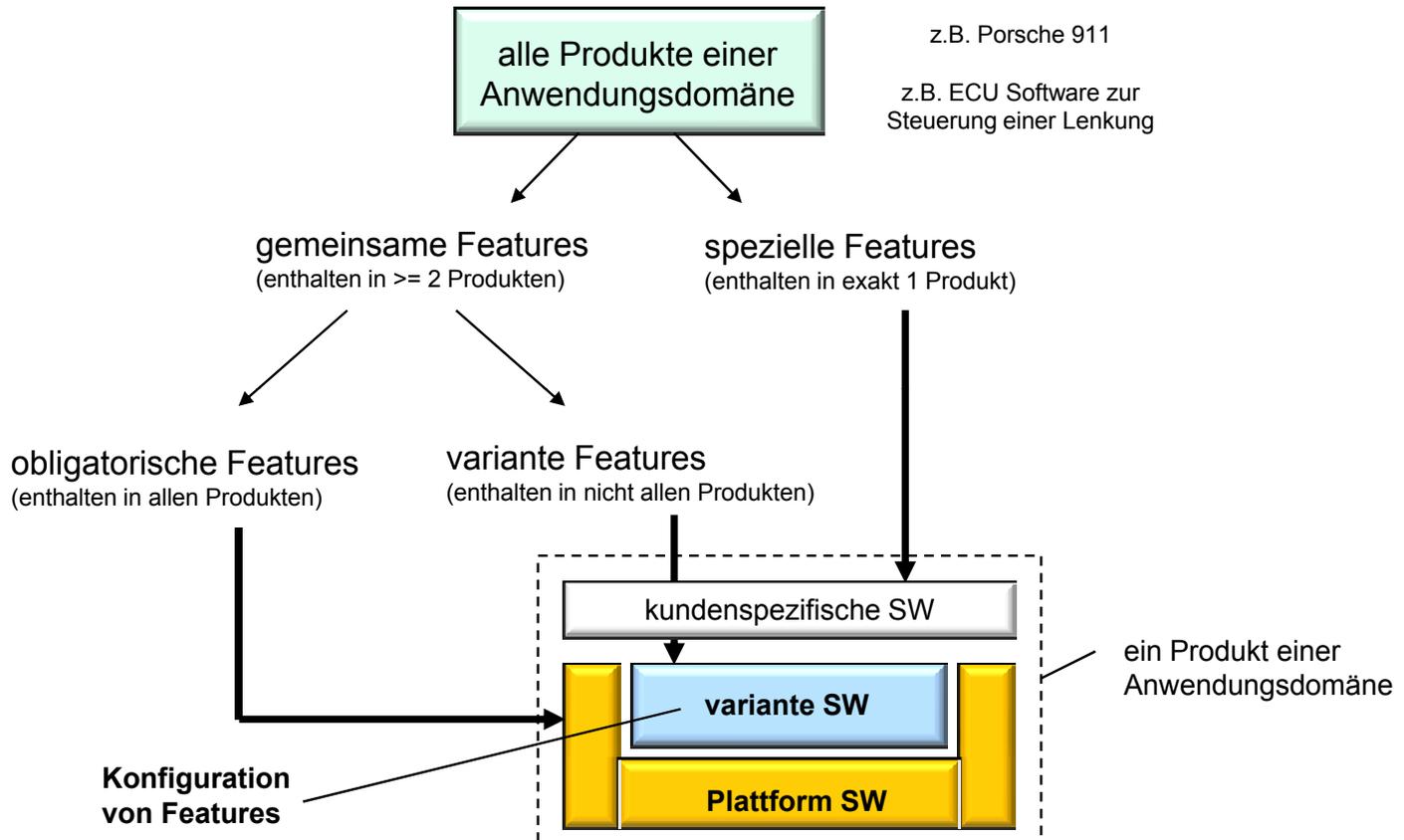


Reifegradmodell – Stufe 3 (Produktlinie)

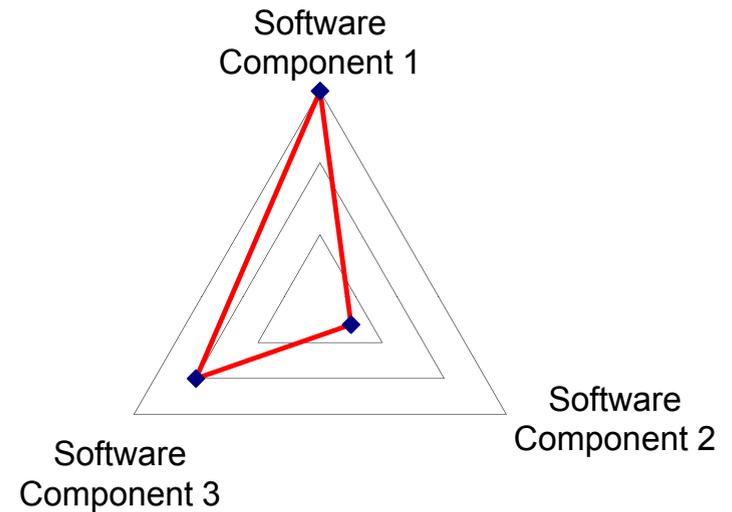
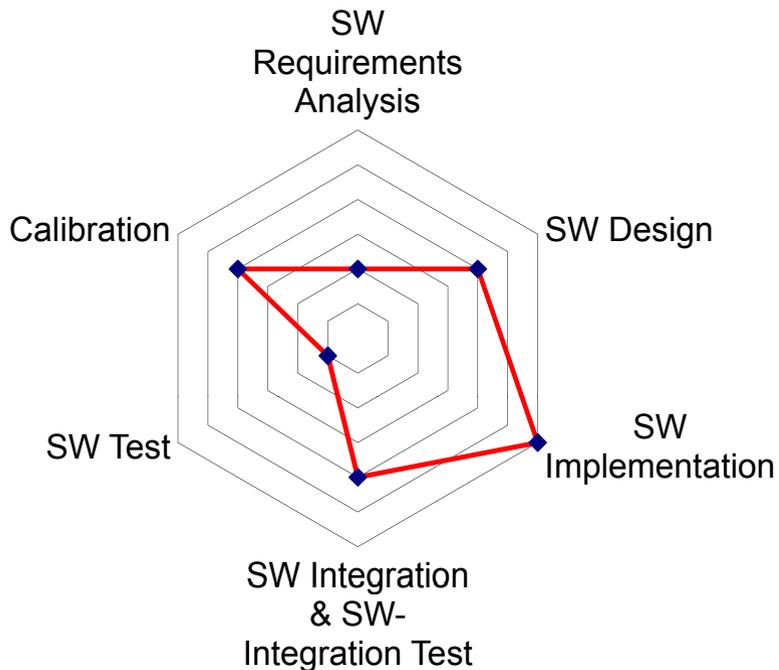


- **Black-Box & White-Box WV** von ECU Software
- SWK, welche in nicht allen Produkten verwendet werden
- diese SWK sind nicht Teil der Plattform
- für sehr reife Anwendungsdomänen
 - **gemeinsame** und **variante** Features aller Produkte wurden bereits identifiziert

Reifegradmodell – Stufe 3 (Produktlinie)



Reifegradmodell – Beispiel aus der Praxis



- **Produkt:** ECU Software zur Steuerung von Getrieben
- **Kontext:** PKW Getriebe, Speichergröße < 4 MByte Flash/EEPROM, ...
- **Organisation:** Abteilung X, Y der Firma Z

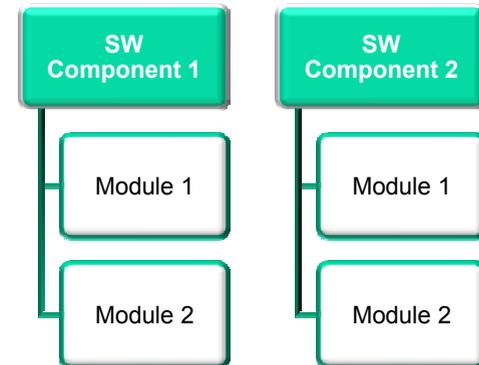
Reifegradmodell – Erfahrungen aus der Praxis

- Reifegrad der untersuchten Organisationen zur automotivem SW-Entwicklung: **Stufe 1 bis Stufe 3**
 - RG teilweise unterschiedlich für jede V-Zyklus Phase !
 - **KRITISCH**: hoher Effizienzverlust !, (“Konvertierung” von SWK)
 - RG teilweise unterschiedlich für jede SWK !
 - Scoping wird nicht durchgeführt !
 - **KRITISCH**: evtl. hoher Effizienzverlust !
(falscher Ziel-Reifegrad & Stufe 3 WV Konzept) !
- **DRINGEND** beachten:
 - Scoping ist ein Muss !
 - liefert Ziel-Reifegrad und Stufe 3 WV Konzept !
 - für alle V-Zyklus Phasen: gleicher Reifegrad & gleiches Stufe 3 WV Konzept ist zwingend erforderlich !



Der Weg zu höheren Reifegradstufen

- Organisatorische Aspekte der WV
 - Einführung einer „**Kultur für WV**“
 - **SWK orientierte** anstatt rein projektorientierte Organisation
 - **Entwicklungsprozess** muss WV unterstützen
 - **Steuerkreis** um Termindruck in Projekten und WV von SWK auszubalancieren
 - primär Aufgabe des Managements



Der Weg zu höheren Reifegradstufen

- Organisatorische Aspekte der WV
 - **Daumen-Regeln:**
 - eine SWK muss **3x** entwickelt werden bevor diese wiederverwendet werden kann
 - Kosten/Nutzen ist erst dann < 1 , falls eine SWK **3x** wiederverwendet wird
 - man benötigt **mindestens 2 Jahre**, um die nächsthöhere Reifegradstufe zu erreichen
 - dies muss Schritt-für-Schritt erfolgen

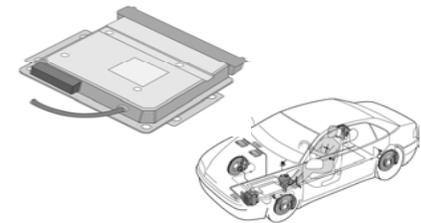


Der Weg zu höheren Reifegradstufen

- Erfahrung aus der Praxis
 - die organisatorischen Aspekte werden üblicherweise **nicht** bzw. **erst sehr spät beachtet** !
 - Konsequenz:
 - die nächsthöhere Reifegradstufe wird erst Jahre später oder nie erreicht !

Der Weg zu höheren Reifegradstufen

- Technologische Aspekte der WV
 - Reifegradstufen 1-2
 - „konventionelles“ Automotive Software Engineering, siehe einschlägige Literatur
 - Reifegradstufen ≥ 3
 - Produkt/Anwendungsdomäne & Produkt-Kontext
 - Beispiel:
 - Produkt/Domäne: ECU Software zur Steuerung von Dieselmotoren
 - Kontext: PKW mit Dieselmotoren, Speichergröße < 4 MByte Flash/EEPROM, ...
 - Kontext ist sehr wichtig, da ECU Software in dieser Umgebung laufen muss
 - eine Plattform der Produktlinie **wird für einen bestimmten Kontext entwickelt!**



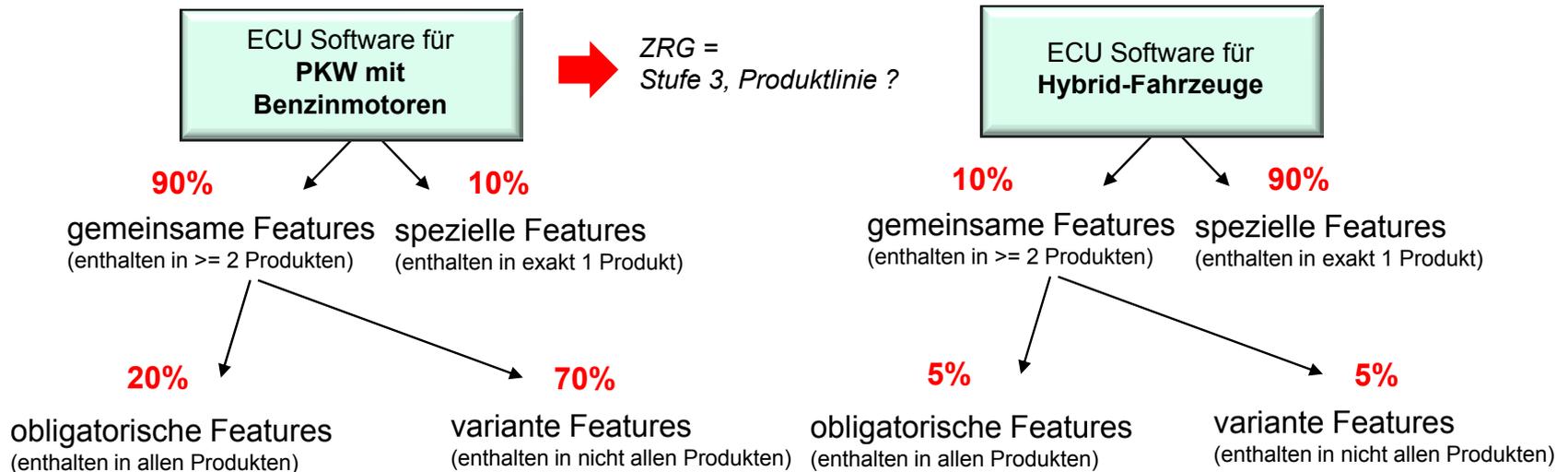
Der Weg zu höheren Reifegradstufen

- Technologische Aspekte der WV
 - Reifegradstufen ≥ 3 (Fortsetzung)
 - Management von Varianten/Features innerhalb aller V-Zyklus Phasen
 - Software Anforderungsanalyse ... Kalibrierung
 - abhängig von Entwicklungsmethodik & Tools
 - z.B. C/C++ Präprozessor-Makros
 - Software-Architektur
 - siehe einschlägige Literatur !
 - Feature-Modellierung und Feature-Konfiguration
 - siehe vorher und einschlägige Literatur

Der Weg zu höheren Reifegradstufen

- Technologische Aspekte der WV
 - **Scoping ist von größter Wichtigkeit** für alle Entscheidungen
 - Welcher ist der richtige Ziel-Reifegrad (ZRG) ?
 - Welches Stufe 3 WV Konzept soll eingesetzt werden ?

➔ ZRG = Stufe 1 ?



Der Weg zu höheren Reifegradstufen

- Welcher ist der richtige Ziel-Reifegrad für eine Organisation ?
 - Welches Stufe 3 WV Konzept soll eingesetzt werden ?
- Lösung: Beziehung zwischen
- Software-Entwicklungskosten,
 - Grad der WV von Features bzw. SWK und
 - Reifegrad benötigt

Normierte SW-Entwicklungskosten

$$\text{SWEK} = f(\text{LOC}, \text{EL}, \text{CO}, \text{N}, \text{R})$$

- LOC **Lines of Code** für eine ECU ("Größe der ECU")
- EL **idealer Aufwand für 1 LOC**
(ohne Komplexitätszuwachs aufgrund WV Technologie)
EL = f (Performance des MA, Effizienz der Prozesse, Methoden, Tools)
- CO **Kosten für ein MA-Jahr;**
CO = f (Gehalt des MA, Infrastruktur)
- N **Anzahl der Projekte** (kumuliert)
- R = f (a, b, c, N); **normierte SWEK pro Projekt**
a – Grad der Black-Box WV der SWK
b – normierte SWEK bei **White-Box WV** einer SWK
b = f (Grad der Änderung einer SWK bei White-Box WV)
c – Aufwandszuwachs mit Grad der Black-Box WV aufgrund komplexerer WV-Technologie, z.B. c=0,1*a

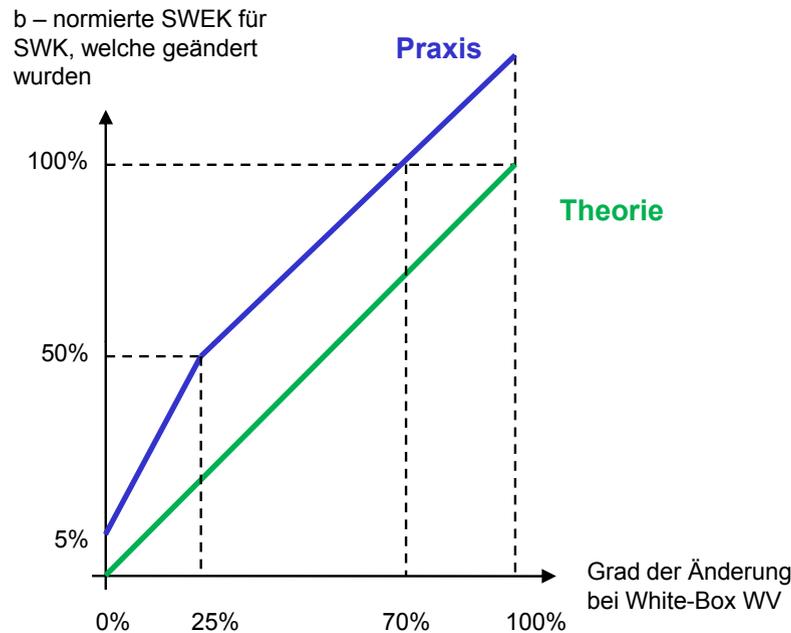
unabhängig vom Grad der WV

stark unabhängig vom MA und deren Arbeitsumgebung!

bekannte Größe für eine ECU SWE Organisation

WV von obligatorischen und varianten Features

Normierte SW-Entwicklungskosten (Faktor b)



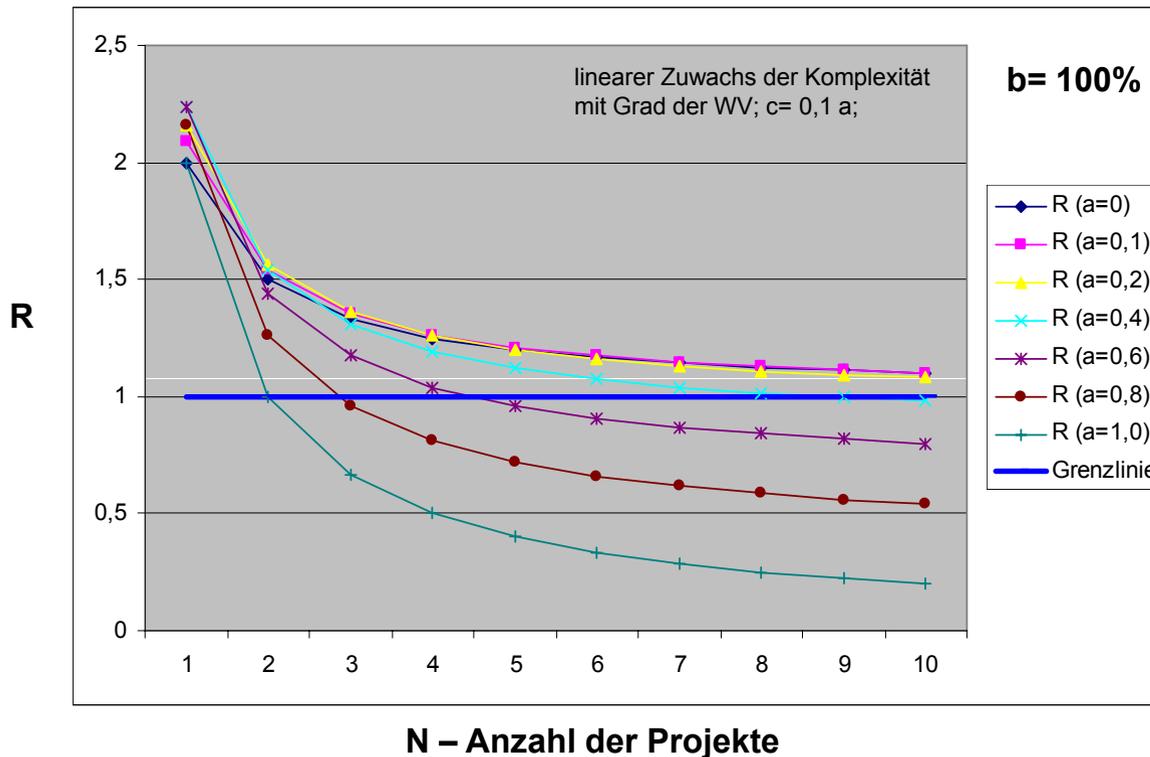
70% einer SWK geändert:

Kosten: **100%** der ursprünglichen Kosten zur Entwicklung der SWK!

0% einer SWK geändert:
(Black-Box WV):

Kosten: **> 5%** der ursprünglichen Kosten zur Entwicklung der SWK!

Normierte SW-Entwicklungskosten



< 30% Black-box WV:
keine Reduzierung der Kosten !

80% Black-box WV,
Projekte > 10:

50% Reduzierung der Kosten !

Assumption: Minimum of 70% of all white box reused SWC will be changed. All black box reusable SWC have already been developed and are stored in an archive (this is the reason why R equals roughly 2 if we have just one project to develop).

Zusammenfassung

Was ist der **Ziel-Reifegrad**?

Was ist das geeignete **Stufe 3 WV Konzept**?

