



Systematische Aufwandsschätzung für Software im Fahrzeug

sd&m AG
software design & management
Carl-Wery-Str. 42
81739 München
Telefon 089 63812-0
www.sdm.de

Member of
Cap Gemini Ernst & Young

Christian Kamm
ProSTEP iViP Symposium 2006
Köln, 26. April 2006

sd&m AG – software design & management

Geschäftsfelder

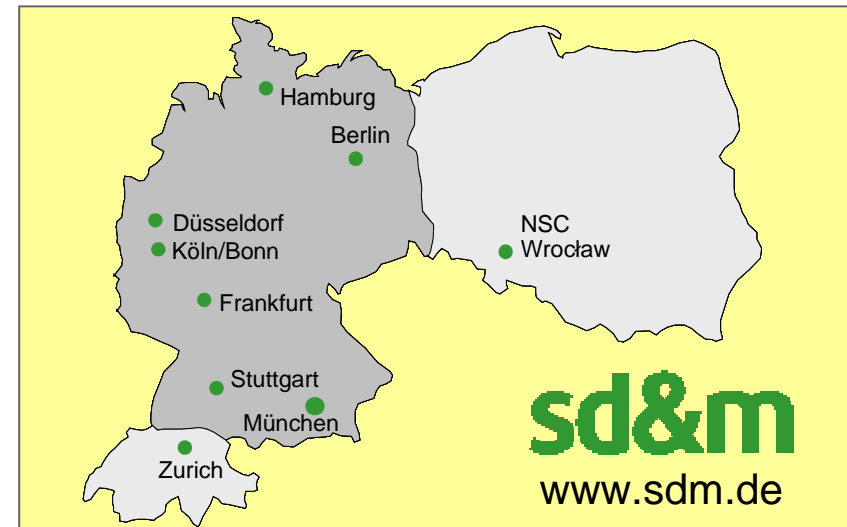
- Entwicklung und Integration maßgeschneiderter Informationssysteme für unternehmenskritische Prozesse
- IT-Beratung mit Umsetzungskompetenz

Kunden

- Namhafte Unternehmen und Organisationen, die durch Einsatz individueller Lösungen Wettbewerbsvorteile erlangen

Kernkompetenz

- Software-Engineering und Projektmanagement



Eckdaten 2005

- Mitarbeiter: 1010
- Umsatz: 141 Mio. €

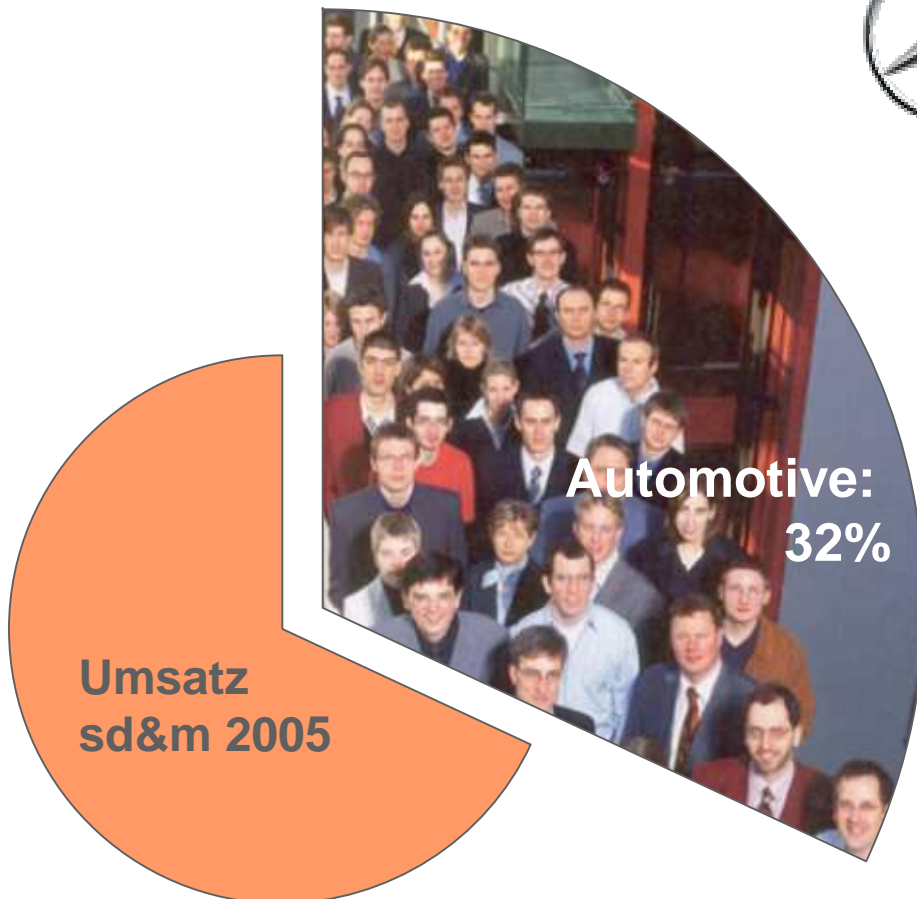
Forschung



Aktionär



280 Mitarbeiter in Automotive-Projekten: sd&m kennt die Branche



DaimlerChrysler

- seit 1996
- aktuell: 180 Mitarbeiter



BMW

- seit 1997
- aktuell: 80 Mitarbeiter



AUDI und VW

- seit 1999
- aktuell: 20 Mitarbeiter



■ Einordnung des Vortrags

Überblick über gängige Schätzverfahren und Vorschlag

Expertenklausur

COCOMO II

Zusammenfassung und Faustregeln

Resumee

Vortragsziel und Nutzen für Sie als Zuhörer

**Vortragsziel –
Diese Fragen beantworten:**

Nutzen für Sie als OEM

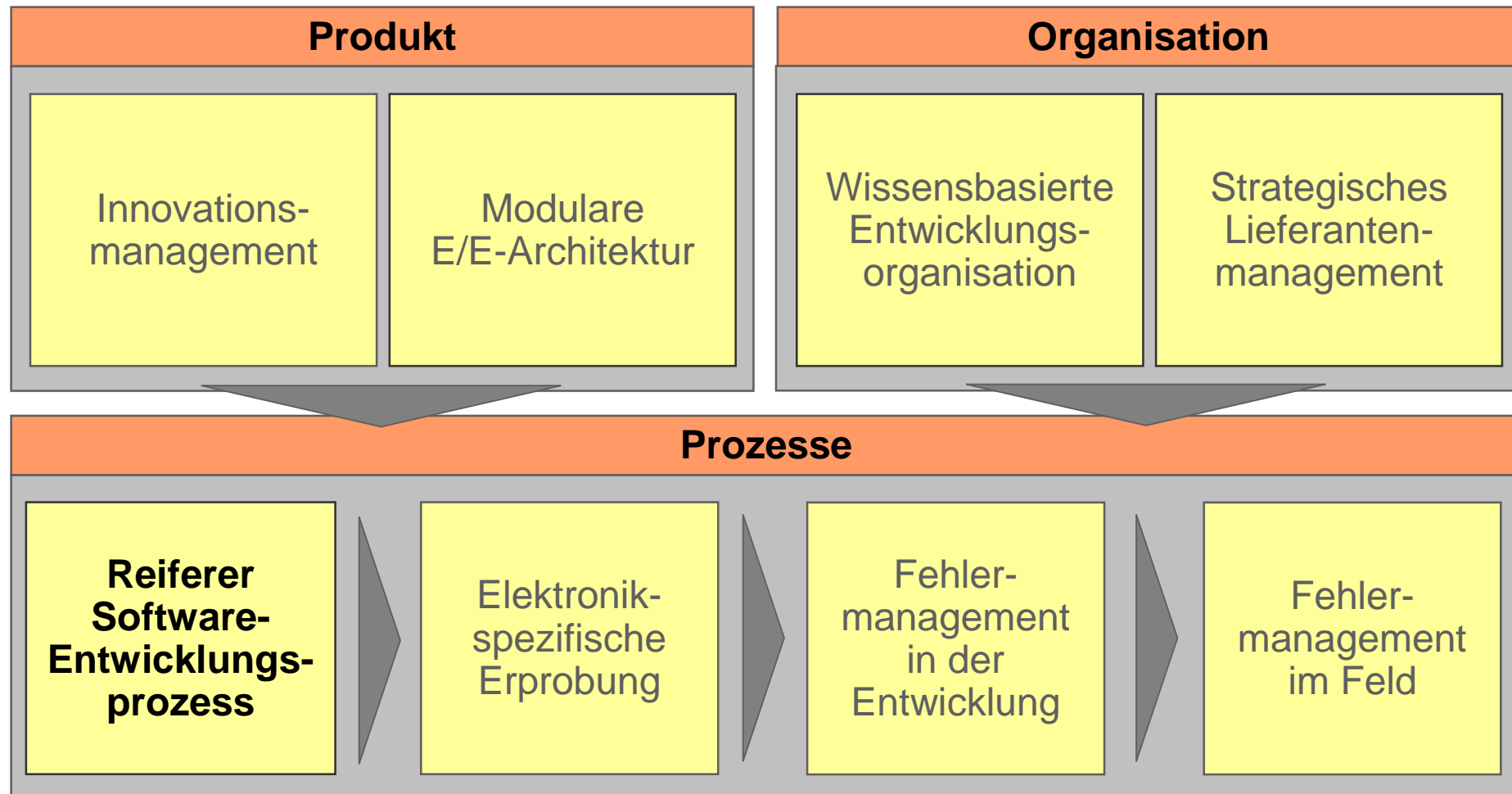
- Aufwandsplausibilisierung
- Mehr Planungssicherheit in Fahrzeugprojekten
- Rückschlüsse auf die Lastenheftgestaltung

Nutzen für Sie als Zulieferer

- Mehr Sicherheit in der Aufwandschätzung und Projektplanung
- Beitrag zum Risikomanagement

Einordnung – Systematische Aufwandsschätzung als Beitrag zu einem reiferen Software-Entwicklungsprozess

¹ Nach McKinsey/PTW-HAWK-Survey





Einordnung des Vortrags

■ **Überblick über gängige Schätzverfahren und Vorschlag**

Expertenklausur

COCOMO II

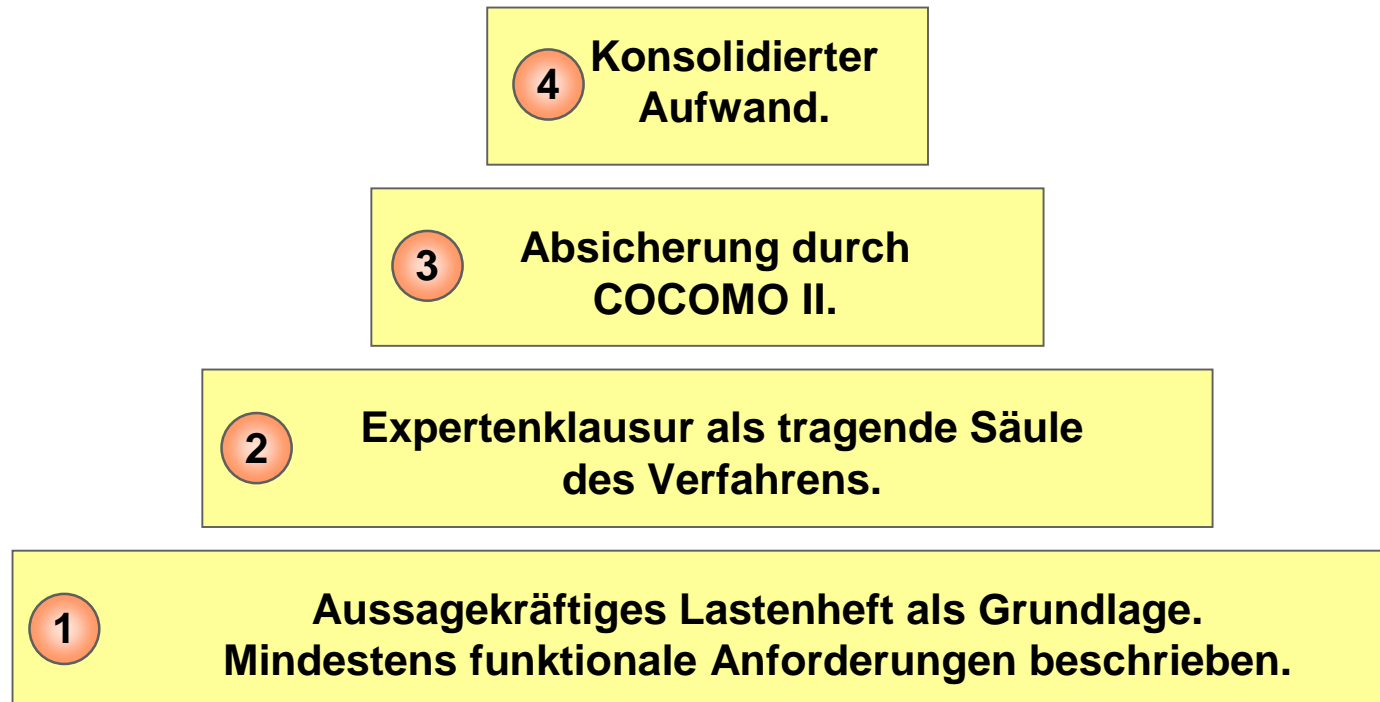
Zusammenfassung und Faustregeln

Resumee

Methoden zur Aufwandsschätzung im Überblick

Methoden	Beschreibung	Aufwand	Knackpunkt
Einzelschätzung	Experte definiert Arbeitspakete und schätzt Aufwand	gering	Experte
Mehrfachbefragung	Mehrere Experten führen unabhängig und einmalig eine Einzelschätzung durch	mittel	Experten
Delphi	Strukturierte anonyme Mehrfachbefragung in mehreren Iterationen	groß bis sehr groß	Experten
Expertenklausur	Strukturierte, kollektive Mehrfachbefragung mit Gruppendynamik auf Basis einer Stückliste	mittel	Experten, Stückliste
Prozentsatzmethode	Über Phasen-Kennzahlen wird der Gesamtaufwand bestimmt	gering	Nur zur Plausibilisierung
Vergleichsmethode	Über ähnliche Projekte wird per Analogieschluss auf den Aufwand für das aktuelle Projekt geschlossen	mittel	Ähnliche Projekte
Function Point	<ol style="list-style-type: none"> Unbewertete FP (~Größe der Software) Bewertete FP (+/- 35%) Aufwandsermittlung auf Basis einer Erfahrungsdatenbank (Aufwand dafür: sehr groß!) 	mittel bis groß	Erfahrungs-DB
COCOMO II	<ol style="list-style-type: none"> Sizing (SW-Größe in LOC) Rating der Faktoren über Tabellen Berechnung des Schätzintervalls 	Sizing:mittel Rating:gering	Sizing

Vorschlag: Zweigleisiges Verfahren aus Expertenklausur und algorithmischer Schätzung mit COCOMO II



Bevor wir in die Details abtauchen - Was schätzen wir überhaupt?

Wir schätzen den **Entwicklungsaufwand** in Bearbeitertagen.

Schätzeinheiten: 1 Bearbeitermonat [BM] = 20 Bearbeitertage [BT]
= 20*8 Stunden [h].

Nettoaufwand: Eigentlicher Aufwand für die Erstellung der Software =
Produktive Leistung des Entwicklungsteams.

Querschnittsaufwand: PM, PL, Chefdesign, Qualitätssicherung,
Konfigurationsmanagement, Meetings, Technische Infrastruktur.

Nebenaufwand: Einarbeitung, Reisezeit, Teamaufbau.

Bruttoaufwand = Nettoaufwand + Querschnittsaufwand + Nebenaufwand.



Einordnung des Vortrags

Überblick über gängige Schätzverfahren und Vorschlag

■ **Expertenklausur**

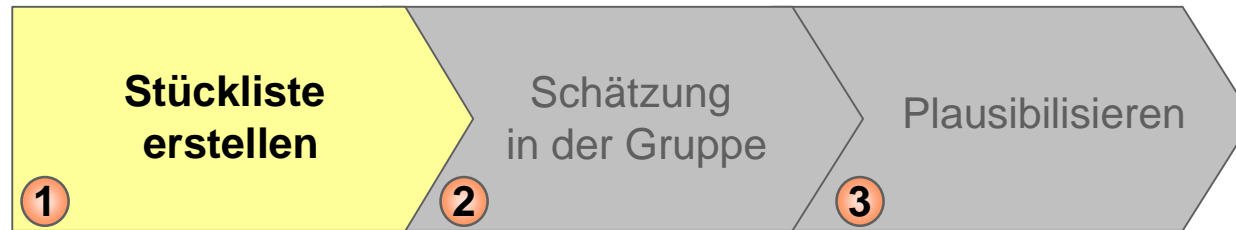
COCOMO II

Zusammenfassung und Faustregeln

Resumee

Expertenklausur (1)

Stückliste erstellen



- Systemarchitektur liefert grobe Komponentenzersetzung.
- Entwicklungsprozess liefert Phasenergebnisse.
- Experte identifiziert Aufwandsposten pro Phase und pro Komponente (*hypotetische Konstruktion*).
- Phasenübergreifende Aufwandsposten separat auführen.
- Granularität der Aufwandsposten muss passen (<20 BT).
- Alle Aufwandsposten für Netto-, Querschnitts- und Nebenaufwand explizit auführen.
- Kategorienbildung für Komponenten bei großen Systemen.
- Nur erfahrene Chefdesigner erstellen die Stückliste: Geschätzt wird nur, was in der Stückliste enthalten ist!

- Systemarchitektur
- Funktionale Anf.
- Nichtfunkt. Anf.
- Entwicklungsprozess



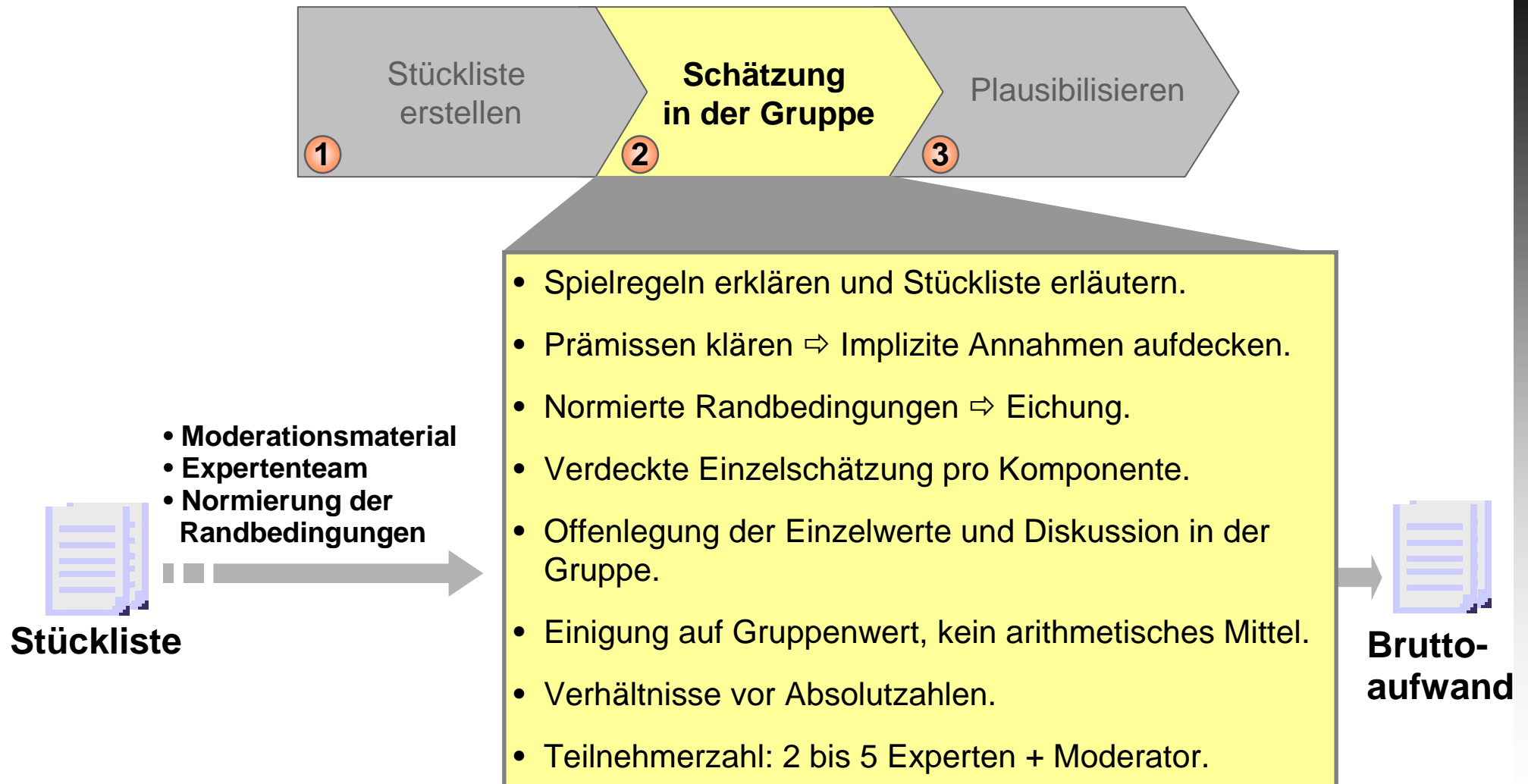
Lastenheft



Stück-
liste

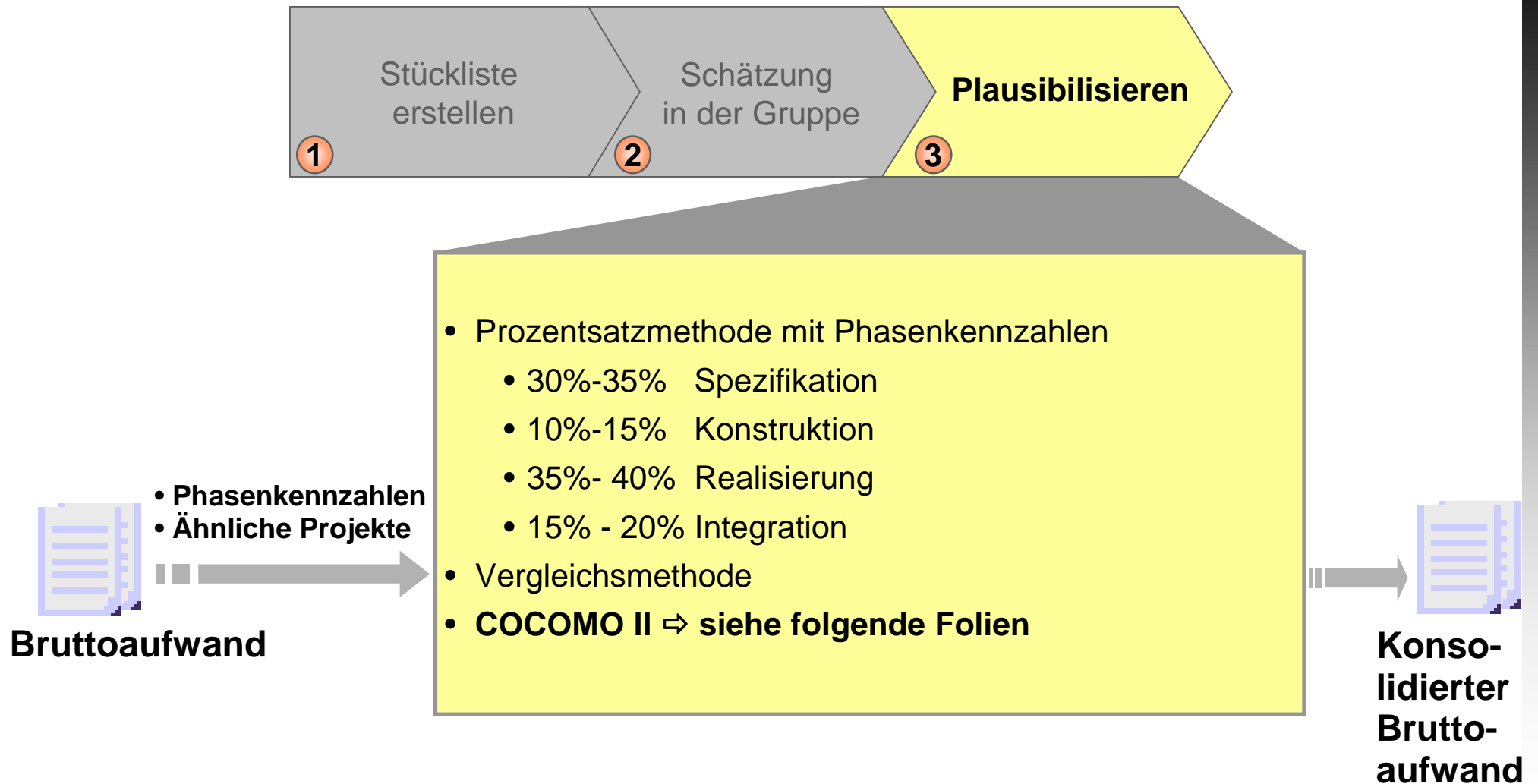
Expertenklausur (2)

Schätzung in der Gruppe

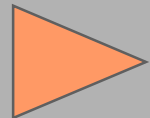


Expertenklausur (3)

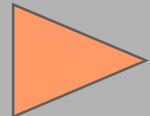
Plausibilisieren



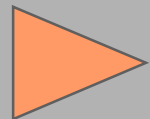
Expertenklausur - Zusammenfassung



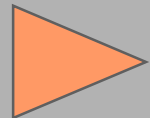
Definition der Stückliste erfordert einschlägige Projekterfahrung.



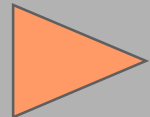
Experten mit Know How für SW-Technik und Anwendung nötig.



Einzel-Schätzwerte liegen oft weit auseinander (Faktor 2-3).

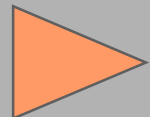


Offene Diskussion der Ausreißer in der Gruppe deckt implizite Annahmen auf und führt zu abgestimmten Gruppenwerten.



Gesamtaufwand für die Expertenklausur: 2 BT - 4 BT je nach Projekt

- Vorbereitung: 1 BT für Definition der Stückliste inkl. Formular.
- Durchführung: max. 4 h, max. 5-6 Teilnehmer.
- Nachbereitung: 0,5 BT



Aufwandschätzung ist Aufgabe des Expertenteams.
Angebotserstellung ist Aufgabe des Projektmanagers.



Einordnung des Vortrags

Überblick über gängige Schätzverfahren und Vorschlag

Expertenklausur

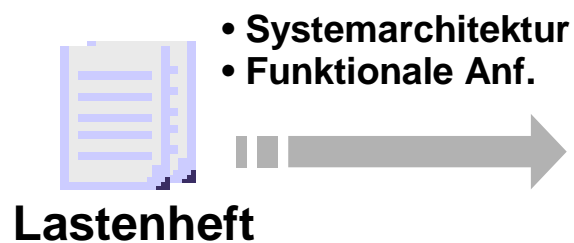
■ **COCOMO II**

Zusammenfassung und Faustregeln

Resumee

COCOMO II – Schätzung (1)

Softwaregröße abschätzen

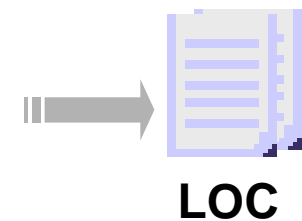


1. Möglichkeit: **Abschätzung über Prototyp**

- Prototyp ausmessen (LOC)
- Hochrechnung für 100% funktionalen Prototyp
- ~ 30% Aufschlag für Fehlerbehandlung im serienreifen Produktivcode

2. Möglichkeit: **Unbewertete Function Points**

- Grobe Heuristik mit Entscheidungstabellen ersetzt IFPUG-Zählregeln
- Funktionale Anforderungen + Datenbestände \Rightarrow Anzahl unbewertete Function Points
- Ermitteln der LOC über Backfiring-Tabellen



COCOMO II – Schätzung (2)

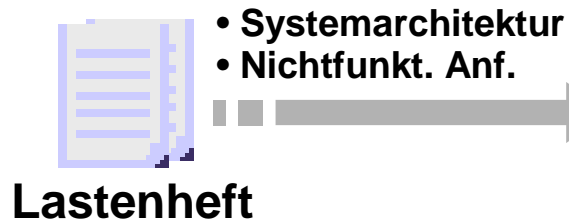
Aufwandsfaktoren bestimmen



Bestimmen der **17 COCOMO-Aufwandsfaktoren** mit Hilfe von Entscheidungstabellen:

- **Produktfaktoren:** Komplexität der Software, Zuverlässigkeit, Wiederverwendbarkeit, Ausführlichkeit der Dokumentation
- **Personalfaktoren:** Fähigkeit von Team und Key-Playern, Erfahrung mit Anwendungsdomäne und Programmiersprache, Kontinuität des Teams
- **Plattformfaktoren:** CPU- und Speicherrestriktionen, Volatilität der Plattform
- **Projektfaktoren:** Werkzeugunterstützung und verteiltes Arbeiten

Die Aufwandsfaktoren wirken **linear** zur SW-Größe



EM1
bis
EM17

Faktor	Sym- bol	Beschreibung	Very Low	Low	Nominal	High	Very High	Extra High
RELY	EM ₁	Zuverlässigkeit: Dies ist das Maß des Umfanges zu dem die Software ihre beabsichtigte Funktion über einen Zeitraum erbringt. Wenn die Auswirkung eines Softwarefehlers nur leichte Unannehmlichkeiten bedeutet, ist RELY niedrig. Wenn ein Fehler ein menschliches Leben bedrohen würde, ist RELY sehr hoch.	leichte Unannehmlichkeiten	leicht wieder herstellbare Datenverluste	wieder herstellbare Datenverluste	hohe finanzielle Verluste	Lebensgefahr	-
DATA	EM ₂	Datenbankgröße: in Bytes/Programm Lines of Code. Je größer die zu testende Datenbank, desto aufwändiger sind die Testdaten und somit erhöht sich dann auch der Gesamtaufwand.	-	D/P < 10	10 <= D/P < 100	100 <= D/P < 1000	D/P > =1000	-
CPLX	EM ₃	Komplexität der Anwendung: Gewichteter Mittelwert aus Bewertung der Kontrollfunktionen, Berechnungsfunktionen, hardwareabhängigen Funktionen, Datenmanagementfunktionen und Benutzerschnittstellenfunktionen.	einfache, elementare Funktionen: Geradliniger Code mit wenigen nicht geschachtelten strukturierten Programmelementen; Auswertung einfacher Ausdrücke; einfache Lese- und Schreiboperationen	relativ einfache Auswertungen: Übersichtliche Schachtelung strukturierter Programmelemente; relativ einfache Ausdrücke evtl. mit einfachen math. Operationen; keine Kenntnis spez. Prozessor oder I/O Eigenschaften (GET, PUT Ebene) erforderlich	einfache Schachtelungen, math. Standardroutinen, Verwendung von Oberflächenbibliotheken: Kontrolle aufgeteilt zw. Modulen, Entscheidungstabellen, Callbacks; Verwendung mathematischer und statistischer Standardroutinen, einfache Matrix- und Vektoroperationen; I/O-Verarbeitung einschl. Geräteauswahl, Statusprüfungen und Fehlerbehandlung	tiefe Schachtelung, Auswertung von Gleichungen, Erweiterung von Oberflächenbibliotheken: Tief geschachtelte strukturierte Programmelemente mit vielen zusammengesetzten Ausdrücken, Verarbeitung von Warteschlangen u. Stapeln; einfache numerische Analyse; I/O auf physischer Ebene (Seeks, Reads etc.), optimierte I/O-Überlappung; komplexe Umstrukturierung von Daten; Bau oder Erweiterung von Oberflächenelementbibliotheken, einfache Sprach-ein-/ausgabe, Multimedia	Rekursion, schwierigere numerische Gleichungen, Multimedia GUI: Rekursionen, Unterbrechungsbearbeitung mit festen Prioritäten, Prozesssynchronisation, komplexe Callbacks, heterogene verteilte Verarbeitung harte Echtzeitanforderungen auf einem Prozessor; einfache Parallelisierung; Routinen für Unterbrechungsdiagnosen - bedingung - maskierung, Umgang mit Datenleitungen performanceintensive Embedded Systems;	Scheduling, Echtzeitanwendung, komplexe Multimediakommunikation: Scheduling vieler Ressourcen mit sich dynamisch ändernden Prioritäten; Verarbeitung auf Mikrocodeebene, harte Echtzeitanwendungen; schwierige und unstrukturierte numerische Analyse, komplexe Parallelisierung; Gerätetiming abhängige Codierung, Mikroprogrammierung, performancekritische Embedded Systems; komplexes Multimedia, Virtual Reality

COCOMO II – Schätzung (3)

Prozessfaktoren bestimmen



Bestimmen der **5 COCOMO-Prozessfaktoren** mit Hilfe von Entscheidungstabellen

- Geläufigkeit der Entwicklung
- Mögliche Entwicklungsflexibilität
- Risiko- und Anforderungsmanagement (Architektur, Projekt)
- Kohäsion und Kooperation des Teams
- Reife des Entwicklungsprozesses

Prozessfaktoren wirken schwach **exponentiell** zur SW-Größe



COCOMO II – Schätzung (4)

Schätzintervall berechnen



- Frei verfügbares Tool: USC-COCOMO-2000.0
- LOC und Faktoren werden im Tool eingetragen
- Tool bietet grafische Benutzeroberfläche und implementiert folgende Formel zur Aufwandsberechnung :

$$A = C \cdot Size^{0,91 + 0,01 \cdot \sum_{i=1}^5 SF_i} \cdot \prod_{i=1}^{17} EM_i$$

- C = 2,94 (Kalibrierungskonstante)
- Size: SW-Größe in KLOC
- SF_i: Prozessfaktoren (Scaling Factors)
- EM_i: Aufwandsfaktoren (Effort Multipliers)

• LOC
• EM1..EM17
• SF1..SF5

Sizing und Rating

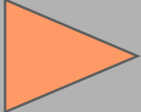
Schätzintervall [optimistic, pessimistic]

COCOMO II – Schätzung

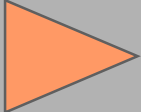
Zusammenfassung



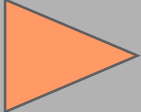
SW-Sizing ist der Knackpunkt bei COCOMO II



Prototyp ist hilfreich für die Abschätzung der Software-Größe



Heuristisches FP-Verfahren mit Vorsicht zu genießen
(empirische Datenbasis aus Business IT).
Die Heuristik sollte für Software im Fahrzeug überprüft werden
⇒ Aufbau einer Erfahrungsdatenbank für FP-Sizing



COCOMO II hervorragend geeignet für Risikoanalyse und
What-If Analysen.



Gesamtaufwand für die COCOMO-Schätzung: ~ 1,5 BT

- Sizing: ~ 1 BT
- Rating: < 0,5 BT



Einordnung des Vortrags

Überblick über gängige Schätzverfahren und Vorschlag

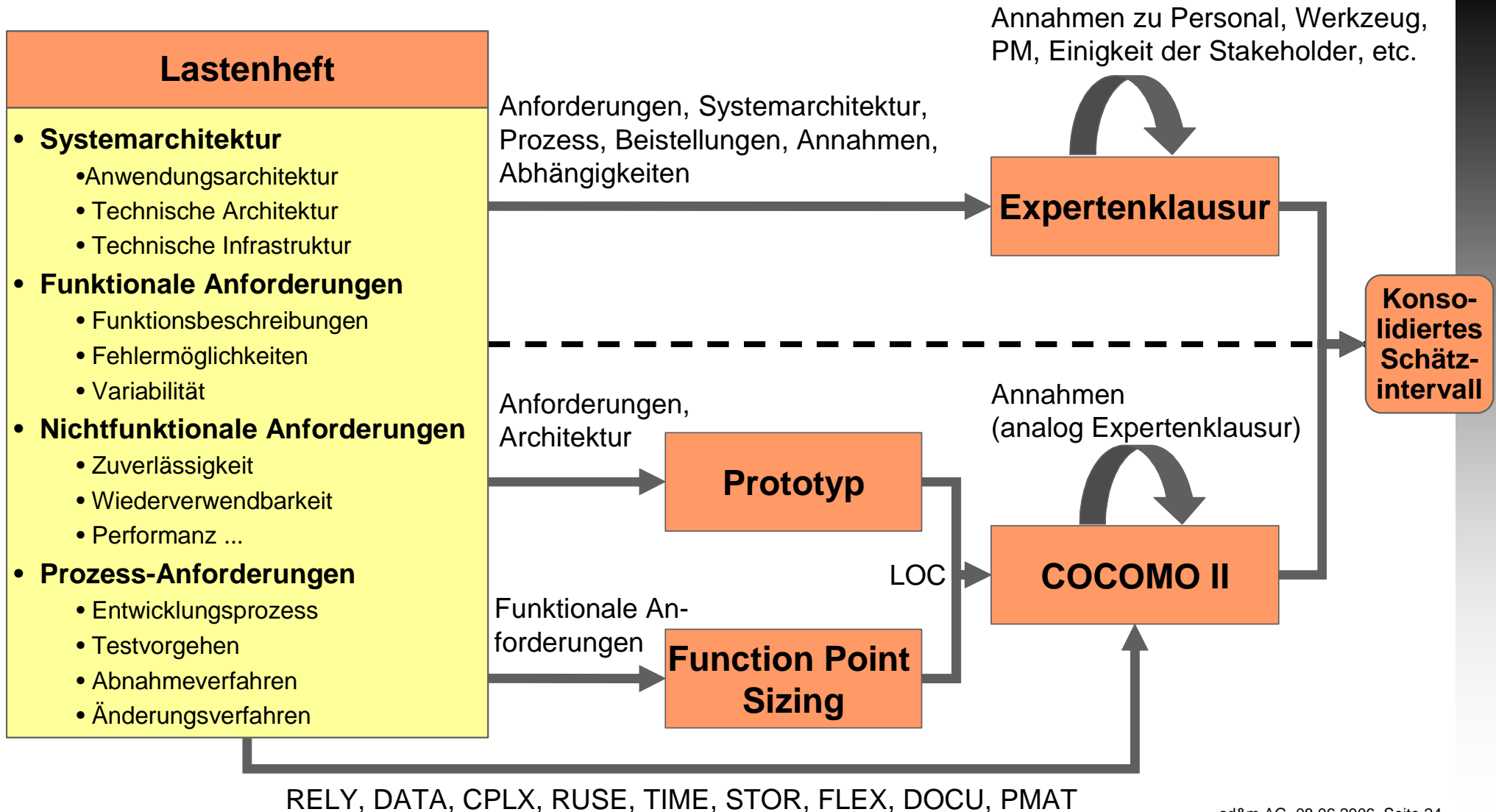
Expertenklausur

COCOMO II

■ **Zusammenfassung und Faustregeln**

Resumee

Zweigleisiges Schätzverfahren im Überblick



Faustregeln zur Aufwandsschätzung

<p>COCOMO II</p> <p>Faktor 2-3 für safety-kritische Anwendungen</p>	<p>COCOMO II</p> <p>Idealteam doppelt so produktiv wie Nominalteam</p>	<p>COCOMO II, sd&m</p> <p>Unsicherheitsfaktor 1,5 bei früher Schätzung auf Basis Lastenheft</p>								
<p>COCOMO II</p> <p>50% Einsparung durch stabile Anforderungen und reifen Prozess im Vergl. zum Worst Case</p>	<p>COCOMO II</p> <p>Faktor 1,5 für wiederverwendbare Komponenten</p>	<p>sd&m</p> <p>Kennzahlen:</p> <table border="0"> <tr> <td>Fachkonzept:</td> <td>30%-35%</td> </tr> <tr> <td>IV-Konzept:</td> <td>10%-15%</td> </tr> <tr> <td>Implementierung:</td> <td>35%-40%</td> </tr> <tr> <td>Integration:</td> <td>15%-20%</td> </tr> </table>	Fachkonzept:	30%-35%	IV-Konzept:	10%-15%	Implementierung:	35%-40%	Integration:	15%-20%
Fachkonzept:	30%-35%									
IV-Konzept:	10%-15%									
Implementierung:	35%-40%									
Integration:	15%-20%									
<p>Tom de Marco, sd&m</p> $Team_{Max} = \sqrt{Aufwand \text{ in } BM}$	<p>COCOMO II</p> $Dauer_{opt} = 2,5 \cdot Aufwand [BM]^{0,35}$	<p>Balzert, COCOMO II</p> <p>Durchschnittliche Produktivität (gesamt): ~ 350 LOC/BM</p>								



Einordnung des Vortrags

Überblick über gängige Schätzverfahren und Vorschlag

Expertenklausur

COCOMO II

Zusammenfassung und Faustregeln

■ **Resumee**

Resumee



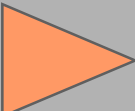
Gutes Lastenheft als solide Basis für Aufwandschätzung nötig

Funktionale Anforderungen, Nichtfunktionale Anforderungen, Prozessanforderungen und grobe Systemarchitektur sollten im Lastenheft beschrieben sein.



Systematische Aufwandschätzung für Software im Fahrzeug ist möglich

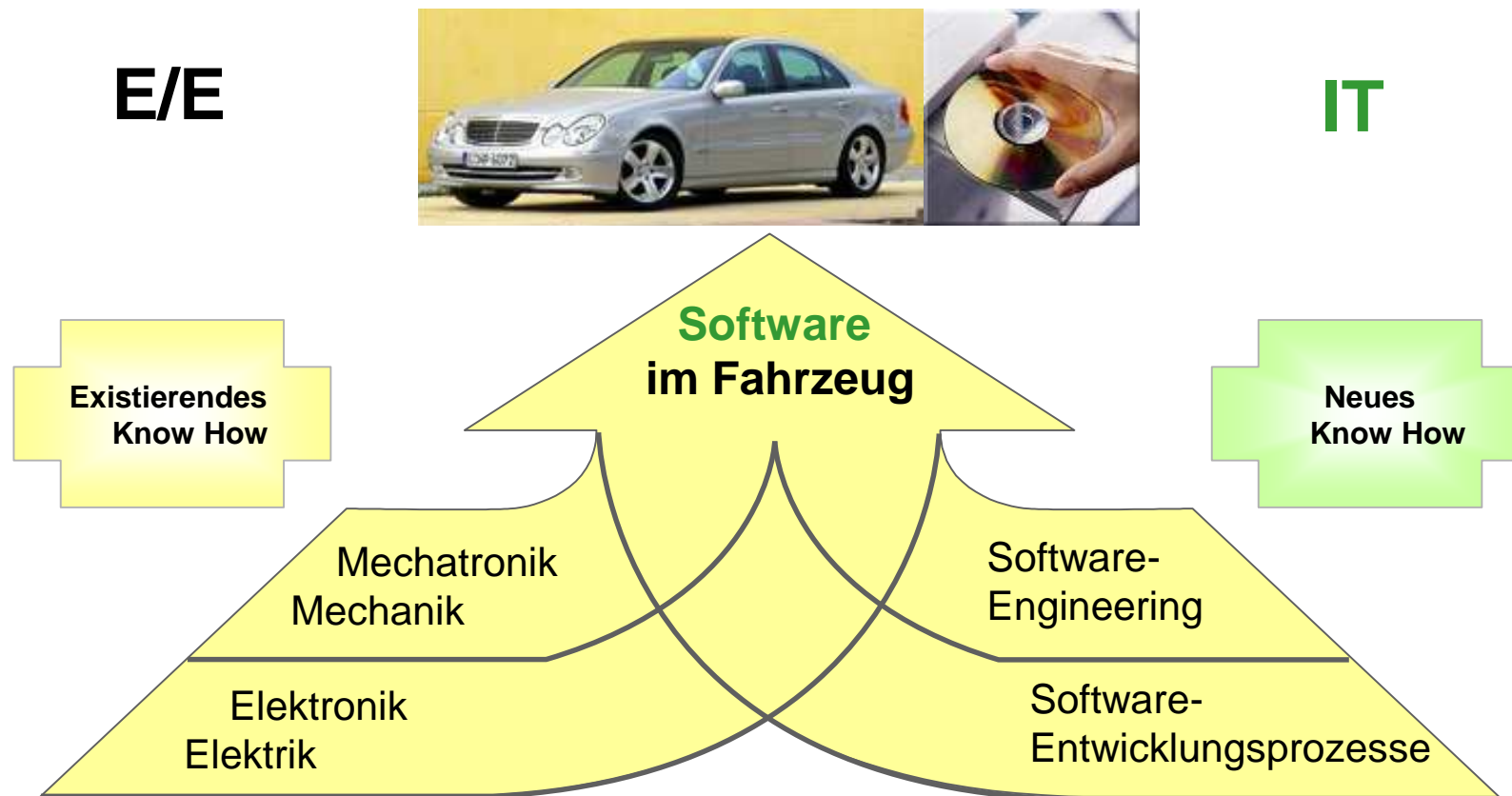
- Zwei genau definierte Verfahren bieten einen festen Rahmen dafür.
- Die Verfahren sind leicht erlernbar und übertragbar auf andere Projekte.
- Absicherung durch redundante Auslegung des Verfahrens.
- Aber: Expertenklausur nicht ersetzbar durch alleinige COCOMO II – Schätzung.
- Und: Für beide Verfahren sind Kontextexperten nötig (Stückliste, Sizing).



Aufwandsfaktoren für Software im Fahrzeug durch COCOMO II gut abgedeckt

- Die COCOMO II - Faktoren decken Business-IT und Fahrzeug-IT ab.
- Personalfaktoren und Prozessfaktoren sind in beiden Bereichen entscheidend ⇒ Lieferant, Team und Projektmanagement erfolgskritisch
- Fahrzeugspezifisch: Produktfaktoren (Zuverlässigkeit, Wiederverwendung über Produktlinien hinweg, Echtzeit), Plattformfaktoren (fehlende Standards)

Fahrzeug-IT und Business-IT wachsen zusammen – Lassen sie uns gemeinsam die neuen Herausforderungen meistern!



„Eine deutliche Steigerung des Reifegrads der Software-Entwicklungsprozesse ist dafür Voraussetzung“